Coordinated Multi-UAV Adaptive Exploration Under Recurrent Connectivity Constraints

Yaqianwen Su[1], Dianxi Shi[1,2](✉), Chao Xue[2], Jiachi Xu[3], and Xionghui He[3]

[1] Artificial Intelligence Research Center, Defense Innovation Institute,
Beijing 100166, China
`dxshi@nudt.edu.cn`
[2] Tianjin Artificial Intelligence Innovation Center, Tianjing 300457, China
[3] College of Computer, National University of Defense Technology,
Changsha 410073, China

Abstract. In the field of multi-UAV collaborative exploration, communication is one of the most fundamental capabilities for effective target deployment and collaborative exploration during mission execution. In order to increase the quality and meet the real-time requirement in various real world situation, collaborate communication strategy has already been a research hotspot both in academia and industry. Recurrent connectivity is a representative strategy with which UAVs do not need to be connected to the base station all the time unless a specific event is triggered. However, in current researches based on the recurrent connectivity strategy, the condition threshold for triggering a new connection is set to be a fixed value during all mission process. This configuration lacks adaptability in real world mission with dynamic and various situations. This paper proposes a dynamic replanning mechanism, and establishes an adaptive multi-UAV collaborative exploration strategy based on recurrent connectivity. Extensive experiments in a well constructed simulation environment were done and the results show that the proposed strategy provides good situation awareness ability at the base station, while our strategy performs an efficient explanation in both complex and simple environments, it has a stronger ability to adapt to complex environments especially.

Keywords: Multi-UAV systems · Adaptive exploration · Communication constraints · Recurrent connectivity

---

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2017YFB1001901.

T. Hara and H. Yamaguchi (Eds.): MobiQuitous 2021, LNICST 419, pp. 737–753, 2022.
https://doi.org/10.1007/978-3-030-94822-1_48

1 Introduction

Multi-UAV systems are widely used to perform missions such as exploration, map building [20], search and rescue [7,19]. Related researches show that collaborative exploration of Multi-UAV systems with communication restrictions is an important topic. Various exploration planning methods have been proposed in recent years, with some experiments presented [6,9,14,15]. UAVs not only need to complete the exploration mission efficiently but also need to exchange data with the base station at the appropriate time. Literatures propose multi-UAV exploration strategies that take into account different types of communication constraints [11,13,16,18]. Among them, recurrent connectivity is a method that can be applied to ensure situation awareness without excessively constraining the exploration mission. Recurrent connectivity means that the base station only ensures a global connection at the initial position of the UAVs, and reconnects with the UAVs every time when it needs to receive new data. As shown in Figs. 1, 2 and 3, they are schematic diagrams of recurrent connectivity.

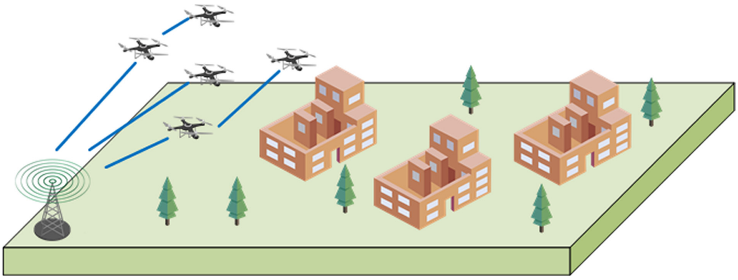


Fig. 1. The base station sends the goal's location to the UAVs.

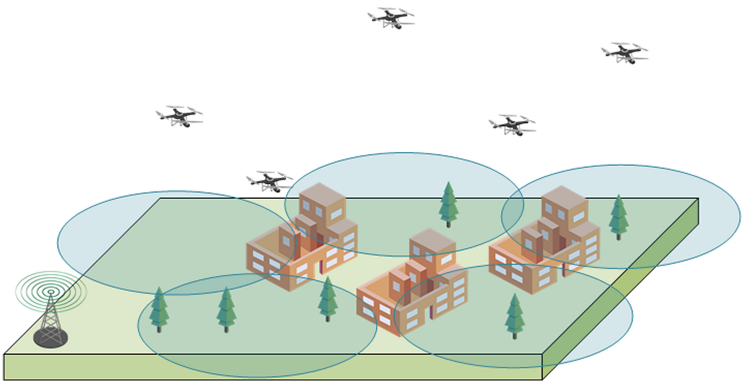


Fig. 2. The UAVs navigate to the goal's location and collects information.

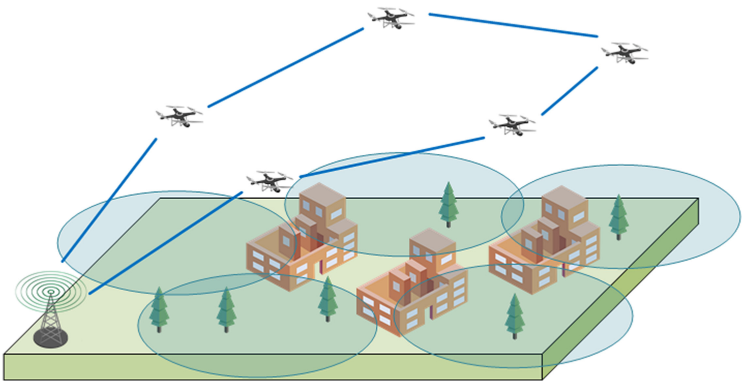


Fig. 3. UAVs transmit the collected information back to the base station and waits to receive the goal's location in the next stage.

However, in the existing exploration strategy under recurrent connectivity constraints, the specific event that triggers a new connection is usually a fixed replanning threshold. The fixed replanning threshold will bring about a series of problems, one is a UAV in the team fails, which makes it impossible to reach the preset threshold and the entire UAV system unable to replan, and the other is, when the threshold is small, frequent replanning will cause the planning to be time-consuming and affect the efficiency of exploration. In the exploration of the unknown environment, we need an exploration strategy that takes into account exploration efficiency and global situational awareness and consumes less energy.

Our work focuses on solving the problem of fixed and inflexible replanning threshold in existing exploration strategies under recurrent connectivity constraints. We propose an adaptive collaborative exploration strategy under recurrent connectivity constraints, construct an adaptive collaborative exploration framework, and designed an adaptive threshold planning algorithm.

In summary, the main contributions of this paper are as follows:

- An adaptive collaborative exploration strategy under recurrent connectivity constraints and an adaptive collaborative exploration framework, which takes into account exploration efficiency and global situational awareness.
- An adaptive threshold planning algorithm and a formal algorithm for Computing the adaptive replanning threshold, which calculates the replanning threshold adaptively according to the UAV's role and the communication path.
- Extensive simulation experiments that validate the proposed method both in complex and simple environments, and effectively reduces energy consumption.

The rest of the paper is organized as follows: Sect. 2 provides a short review of communication-constrained multi-UAV exploration. Section 3 first introduces the task scenario of this work and then proposes an adaptive collaborative exploration strategy under recurrent connectivity constraints. Section 4 presents our experimental results and Sect. 5 concludes the paper.

2 Related Work

Currently, exploration research under communication connectivity requirements is divided into three categories [1], namely exploration without any connectivity, exploration under continuous connectivity, and exploration under event-based connectivity. Since we aim to study the exploration under communication connectivity constraints, the exploration without any connection requirement will not be repeated.

Exploration under continuous connectivity is to maintain a continuous connection between all UAVs and the base station directly or in a multi-hop manner. Suitable scenarios of continuous connectivity include scenarios that require operators to have access to real-time image streams (such as search and rescue [10]) and scenarios that ensure a high degree of coordination between multi-UAV systems. When exploring under continuous connectivity, new plans are usually based on map knowledge shared by all the UAVs in the team. [13] developed a local search method to calculate the usefulness of team composition based on the distance from the nearest frontiers, configurations that did not meet continuous connectivity are severely penalized and are not selected by the algorithm. [12] proposed a distributed protocol to maintain the connectivity of the physical layer of mobile wireless networks. Obviously, ensuring continuous connectivity is associated with non-negligible costs of exploration performance.

Exploration under event-based connectivity refers to the UAV reconnecting with other UAVs and the base station based on a predetermined specific event trigger. The specific events include acquiring new information in the environment, passing a preset time interval, and so on. Exploration under event-based connectivity includes periodic connectivity and recurrent connectivity.

Periodic connectivity means that UAVs are allowed to autonomously explore unknown areas, but the perception data must be sent back to the base station regularly. Applicable scenarios of periodic connectivity include search and rescue in cities with many obstacles. Periodic connectivity is regarded as an asynchronous condition by some articles which is desirable but not enforced as a hard constraint. [21] includes a criterion that considers the probability of communication in the search strategy in order to prioritize places with a high probability of communication. [2] and [8] investigated stronger forms of asynchronous connectivity. The former focused on line-of-sight connectivity, and it proposed a behavior-based architecture and tested it in exploration scenarios of additional prior information about the environment. The latter did not explicitly consider a fixed BS, but it can fully explore the unknown environment in a decentralized way. The proposed architecture is based on actions and messages exchanged between the robot and the placed beacon.

Recurrent connectivity can be defined as ensuring a global connection only at the deployment location of the UAV and forcing the connection every time the

UAV collects new data. It not only has relatively loose requirements for communication connection but also can carry out timely global situational awareness. It is a trade-off between communication connection and the global situation and is widely used in search and rescue, reconnaissance scenarios.

[18] solved two problems, one is finding a deployment of relay nodes, which ensures global connectivity between each agent and the base station, this problem was reduced to the computation of a minimum Steiner tree with the agents' locations as a terminal set, and the other is given the current deployment and new locations agents should reach, finding the redeployment that minimizes UAVs' traveling time, this problem was solved by using a dynamic programming algorithm. [5] split the general optimization problem into sub-problems: Explorers placement, relays placement, and UAV path generation. In particular, given a set of candidate locations to be connected, relays placement is achieved by solving variations of the Steiner minimum tree problem with a minimum number of Steiner points and bounded edge length. [4] proposed a single-stage strategy based on Integer Linear Programming for selecting and assigning UAVs to locations. They design a two-stage strategy to improve computational efficiency, by separating the problem of locations' selection from that of UAV-location assignments. Extensive testing both in simulation and with real UAVs shows that the proposed strategies provide good situational awareness at the base station while efficiently exploring the environment.

3 Method

3.1 Scenario Description

In this paper, we consider using multiple UAVs to explore an initially-unknown, two-dimensional, continuous, and bounded environment, as shown in the Fig. 4. There are obstacles of different shapes in this environment. The base station(BS) as a supervising control center exists in a fixed, known location in the environment. There is a team of n UAVs to perform unknown environment exploration missions, and each UAV is equipped with finite-range sensors to detect the outer boundaries of obstacles and perceive the surrounding space. Whenever the UAV sends new information, the BS will update the map of the exploration area and is responsible for the planning of the UAV's goal location. The whole UAV team collaborates to explore until the exploration mission is completed, and the BS finally obtains a complete global map. Data exchange between UAV and UAV, UAV and the BS through an *ad hoc* network.

3.2 Adaptive Collaborative Exploration Strategy Under Recurrent Connectivity Constraints

Recurrent Connectivity means that the global connection is only maintained at the initial deployment position of the UAVs, and it can be disconnected for any length of time during the journey to the goal's location. A new connection

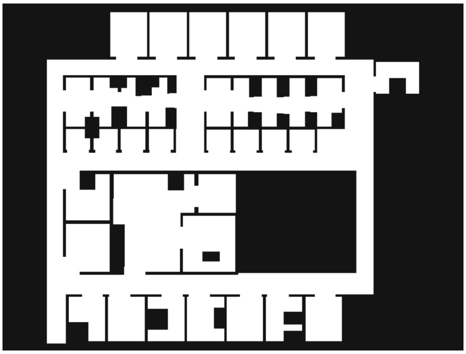


Fig. 4. Schematic figure of the scenario.

is triggered by a specific event (such as the completion of the transmission of information, a certain number of UAVs reach the ready state, etc.). In the current research work, this specific event that triggers a new connection is generally that a fixed number of UAVs reach the ready state (ready state refers to the UAV 1) reached its goal's location in this stage, 2) has transmitted its perceived data to the BS, 3) has no other UAVs still require it as a relay), that is, the time to trigger a new connection is when the replanning threshold reaches a preset fixed value.

Aiming at the initially unknown two-dimensional, continuous, and bounded environment exploration problem in Fig. 4, we propose an adaptive collaborative exploration strategy under recurrent connectivity constraints, and construct an adaptive collaborative exploration framework, as shown in Fig. 5.

This framework mainly describes the behavior of the BS and UAVs in the exploration process using the adaptive collaborative exploration strategy under recurrent connectivity constraints. The framework uses an adaptive collaborative exploration strategy under recurrent connectivity constraints to accomplish the exploration mission, which is usually divided into several stages. In each stage, the BS mainly performs global map update and preprocessing, UAV's state recognition and confirmation of whether replanning is required, and obtains the goal's position of each UAV and replanning threshold in the next stage through the adaptive threshold planning algorithm (ATP algorithm). The UAV mainly uses the sensing module to sense the communication environment, obstacles, and its own location, and exchange the perceived data with the BS. It receives the goal's location of the next stage sent by the BS and uses the navigation module to navigate to the goal's location autonomously, and updates the own map for the next stage of exploration. Figure 6 shows a snapshot of the exploration process.

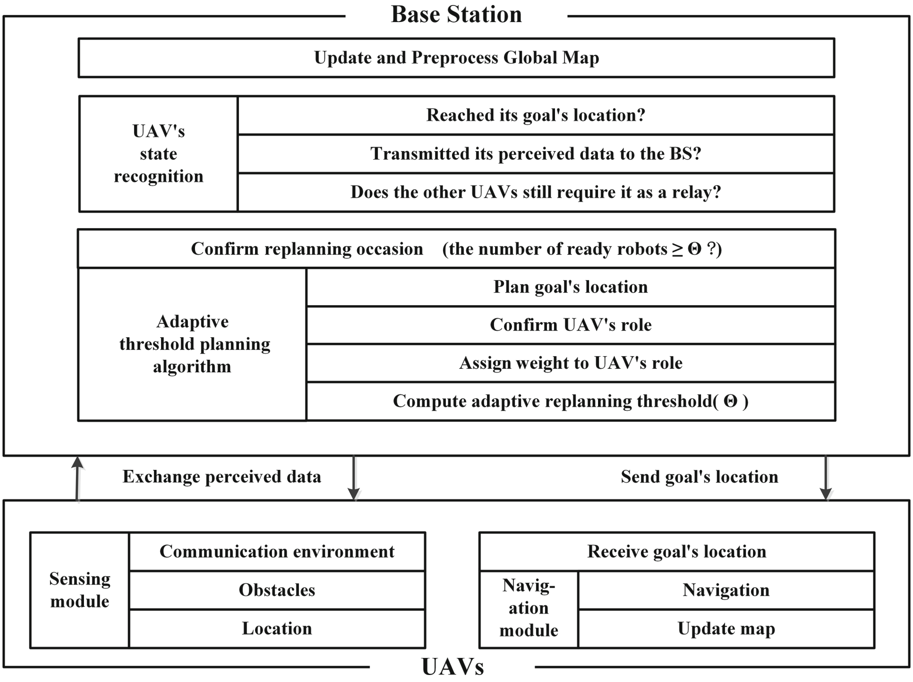


Fig. 5. Adaptive collaborative exploration framework.

Fig. 6. Exploration snapshot. Blue and red - not ready and ready UAVs. Green squares - The BS. Green lines - current communication links. (Color figure online)

3.3 Adaptive Threshold Planning Algorithm (ATP Algorithm)

The adaptive threshold planning algorithm is divided into four parts, which are planning the goal's location, confirming UAV's role, assigning weight to UAV's role, and computing the adaptive replanning threshold. The overall flow ATP algorithm is shown in Fig. 7.

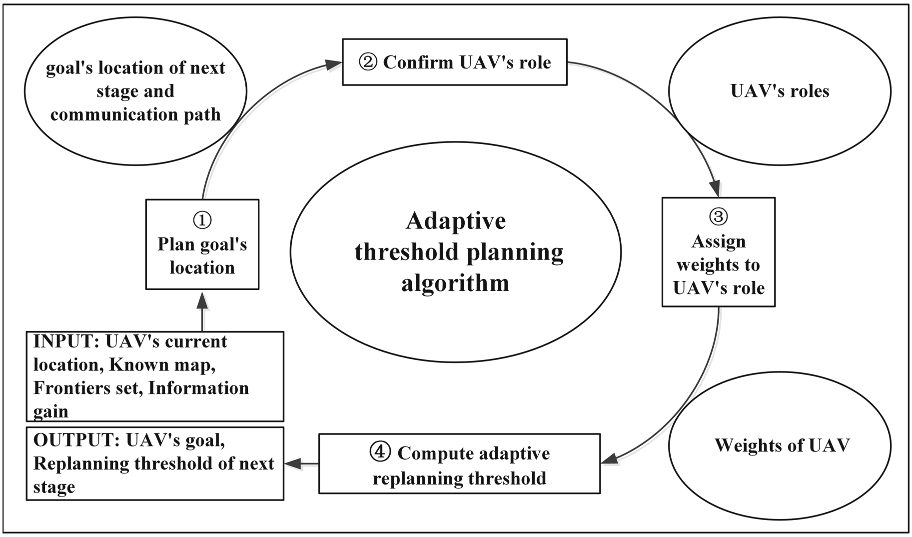


Fig. 7. The overall flow ATP algorithm.

The input of the algorithm is UAV's current location, known map, candidate frontiers set, and information gain. The candidate frontiers set and information gain are obtained by the discretization preprocessing of the map by the BS in this strategy. The candidate frontiers set is, based on the literature [22], clustered representation of the frontiers of the existing map, forming a candidate set of goal's location. Information gain is estimated by acquiring sensor data preprocessed map information (see [11,16]). The first part of the algorithm is to plan the goal's location and obtain a communication path of the next stage. In the second part, the UAV's role is confirmed according to the degree of the node of the communication path tree and whether the location of the node belongs to the candidate frontiers set. The third part assigns weights to each UAV based on its contribution to the map update and information gain. The fourth part is to compute the replanning threshold of the next stage according to the adaptive replanning threshold formula. The output of the algorithm is the UAV's goal and the replanning threshold of the next stage.

Planning the Goal's Location. This part is to plan the goal's location and calculate the communication path based on the current location of the UAVs and the known map information. We work on a graph-based representation of the environment $G = (V, C)$ where vertices in V encode some discretization. Each vertex $v \in V$ is associated with a location, the edge set C encodes the availability of communication links between pairs of vertices. We select the goal's location from the candidate frontiers set according to the utility function [3], and arrange each UAV on the goal's location. Through the calculated goal's location information, the tree of the communication path of the next stage is obtained.

Confirming UAV's Role. Through the first part, we got the planned goal's location of each UAV in the next stage, and also got the tree of their communication path. Figure 8 is a snapshot of exploration. It can be seen that the communication link between the UAV and the BS forms a tree structure, and the root node of the tree is BS. We define UAVs as three types of roles based on their location and their contribution to updating the map, namely explorer, relay, and composite relay.

Fig. 8. A snapshot of exploration shows a tree structure.

- Explorer. The explorer is at the frontier of exploration during the exploration stage, and is only responsible for exploring unknown areas and updating the

map, and does not undertake relay tasks. The formal definition is shown in Eq. (1), that is, the location of the UAV belongs to the candidate frontiers set, its information gain is greater than 0, and it has no relay function.
- Relay. The relay only undertakes the relay task and does not contribute to the map update during the exploration stage. The formal definition is shown in Eq. (2), that is, the location of the UAV does not belong to the candidate frontiers set, its information gain is 0, and it only has the relay function.
- Composite Relay. The composite relay is at the frontier of exploration, but also undertakes the relay task. The formal definition is shown in Eq. (3), that is, the location of the UAV belongs to the candidate frontiers set, its information gain is greater than 0, and it is also a relay.

According to the definition, UAVs E and D in Fig. 8 are explorers, UAVs C and F are relays, and UAVs A and B are composite relays.

$$\begin{cases} Info\_gain(Loc_{r_i}) > 0 \\ Loc_{r_i} \in Frontiers^t \\ IsRelay(r_i) = \text{FALSE} \end{cases} \tag{1}$$

$$\begin{cases} Info\_gain(Loc_{r_i}) = 0 \\ Loc_{r_i} \notin Frontiers^t \\ IsRelay(r_i) = \text{TRUE} \end{cases} \tag{2}$$

$$\begin{cases} Info\_gain(Loc_{r_i}) > 0 \\ Loc_{r_i} \in Frontiers^t \\ IsRelay(r_i) = \text{TRUE} \end{cases} \tag{3}$$

Algorithm 1 formally presents the steps of confirming UAV's role. On lines 1–3 of Algorithm 1, the algorithm shows the confirmation of the explorer, that is, when the degree of the node is equal to 0, the UAV is an explorer. On lines 4–9 of Algorithm 1, the algorithm shows the confirmation of the relay and composite relay, that is, when the degree of the node is greater than 0, and the location of the node is not in the candidate frontiers set, the UAV is a relay; when the degree of the node is greater than 0, and the location of the node is not in candidate frontiers set, the UAV is a composite relay.

Assigning Weight to UAV's Role. The steps of assigning weight to UAV's role are shown in Algorithm 2. According to the contribution of each role to the map update, we assign the weight of the three types of roles, the relay weight is 0 (Eq. (4)), the explorer weight is 1 (Eq. (5)), and the weight of the composite relay is assigned by the information gain of its location (Eq.(6)), where $r_i \in R_{cr}^{t+1}$.

$$\omega_r = 0 \tag{4}$$

$$\omega_e = 1 \tag{5}$$

Algorithm 1 Confirming UAV's role

```
Input: $T^{t+1}$, $F^t$
Output: $R_r^{t+1}, R_e^{t+1}, R_{cr}^{t+1}$
for each $r_i^{t+1} \in T^{t+1}$ do
    if $degree(r_i^{t+1}) = 0$ then $r_i^{t+1} \in R_e^{t+1}$
    end if
    if $degree(r_i^{t+1}) > 0$ then
        if $Loc_{r_i}^{t+1} \notin F^t$ then $r_i^{t+1} \in R_r^{t+1}$
        end if
        if $Loc_{r_i}^{t+1} \in F^t$ then $r_i^{t+1} \in R_{cr}^{t+1}$
        end if
    end if
end for
return $R_r^{t+1}, R_e^{t+1}, R_{cr}^{t+1}$
```

Algorithm 2 Giving weight to UAV's role

```
Input: $R_r^{t+1}, R_e^{t+1}, R_{cr}^{t+1}, Info\_gain(F^t)$
Output: $W^{t+1}$
for each $r_i^{t+1} \in R_r^{t+1}$ do
    $\omega_{r_i} = 0$
end for
for each $r_i^{t+1} \in R_e^{t+1}$ do
    $\omega_{r_i} = 1$
end for
for each $r_i^{t+1} \in R_{cr}^{t+1}$ do
    $\omega_{r_i} = Info\_gain(Loc_{r_i})$
end for
return $W^{t+1}$
```

$$\omega_{r_i} = Info\_gain(Loc_{r_i}) \tag{6}$$

Computing the Adaptive Replanning Threshold. We propose a formula (Eq. (7)) for Computing the adaptive replanning threshold, that is, the sum of the weights of all UAV roles is rounded down to get the replanning threshold for the next stage. Rounding down is due to our slack threshold strategy. According to the difference of three types of roles, our formula can be written in the form of Eq. (8). According to the weight assignment of different roles, the final form of the formula is Eq. (9).

$$\theta^{t+1} = \lfloor \sum_{r_i \in R^{t+1}} \omega_{r_i} \rfloor \tag{7}$$

$$\theta^{t+1} = \lfloor \sum_{r_i \in R_r^{t+1}} \omega_{r_i} + \sum_{r_j \in R_e^{t+1}} \omega_{r_j} + \sum_{r_k \in R_{cr}^{t+1}} \omega_{r_k} \rfloor \tag{8}$$

$$\theta^{t+1} = \lfloor n_r * 0 + n_e * 1 + \sum_{r_i \in R_{cr}^{t+1}} Info\_gain(Loc_{r_i}) \rfloor \tag{9}$$

Through the above four parts, the final output of the adaptive threshold planning algorithm are obtained, that is, the UAV's goal and the replanning threshold of the next stage.

4 Experiments

4.1 Configuration of Experiments

The hardware and software configuration of our experiments is shown in Table 1. We choose the MRESim [17] simulator for its focus on communication and we select two environments of size 80 m × 60 m (represented by occupancy grids, whose cell edge length is 10 cm), shown in Fig. 9. Alabama and grass are from the MRESim repository (Alabama represents an environment with simple obstacles, and grass represents an environment with complex obstacles). We run simulations with teams of 6 UAVs moving at a constant speed of 0.7 m/s, and equipped with a depth camera with a maximum range of 5 m, a 60°FOV, and an angular resolution of 1°. For each experimental set, 5 runs of 90% explored area is executed for each environment, randomly varying the starting positions of the BS and UAVs. For the simulation experiments, we assume that the communication model UAVs are endowed with coinciding with the actual possibility of communicating in the simulated world while ensuring enough bandwidth.

Table 1. Configuration of experiments

Categories	Configuration	
Hardware	Processor	Intel Core i7-9700 3.0 GHz
	RAM	16G
Software	Operation System	Ubuntu 16.04 x64
	Language	Python,Java
	IDE	PyCharm,Apache NetBeans
	Simulator	MREsim v2.0

4.2 Simulation Experiments

Evaluation Metrics. For the evaluation and comparison of individual exploration algorithms, We use the following five metrics:

- Total time explored, the total time required for the UAV team to complete the exploration mission, reflects the efficiency of the exploration.

(a) Alabama (b) Grass

Fig. 9. Simulation environments, approximate size 80 m × 60 m.

- Total time not in communication, the time the UAV team is in the “offline” state during the exploration process, reflects the situational awareness of the BS. The longer this time, the weaker the BS’s overall situational awareness of the UAV team.
- Total distance traveled, the total distance traveled by the UAV team during the exploration process, reflects the energy consumption of the entire team.
- Total times replanned, the number of replanning by the UAV team during the exploration process, reflects the efficiency of the planning algorithm.
- Average time replanned, the average time for the BS to perform replanning, also reflects the efficiency of the planning algorithm.

We evaluated three methods separately, one is our adaptive threshold planning algorithm (ATP), one is the fixed threshold planning approximate algorithm(APX) [4], and the other is the Utility [16] method. Among them, the fixed threshold of the APX algorithm has six cases of 1–6.

Table 2. Comparison of APX and ATP

Environment	Algorithm	Total time explored [s]	Total times replanned	Average time replanned [s]
Alabama	**APX1**	481.6	65.4	1.303
	APX2	308.2	43.0	1.139
	APX3	387.5	34.2	1.138
	APX4	350.9	28.8	0.902
	APX5	305.7	28.8	0.861
	APX6	385.5	28.0	0.912
	ATP	**403.3**	**35.4**	1.194
Grass	**APX1**	658.2	83.8	2.185
	APX2	490.8	54.8	2.019
	APX3	341.8	42.0	1.579
	APX4	382.1	35.6	1.313
	APX5	576.1	36.8	1.639
	APX6	364.7	31.2	1.277
	ATP	**388.6**	**40.4**	1.780

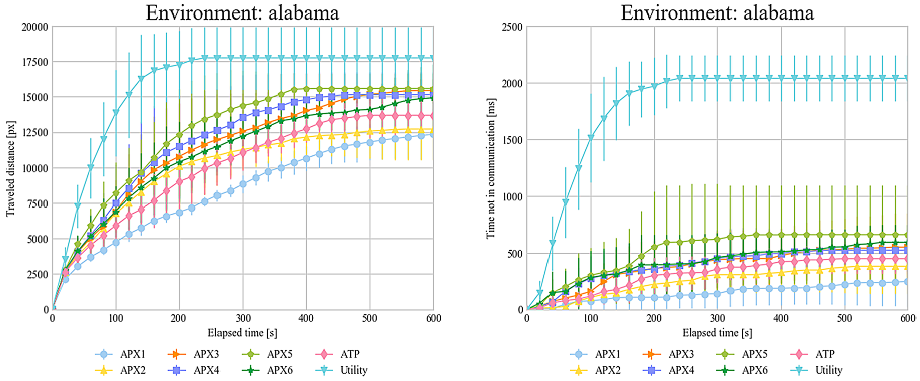
Table 3. Comparison of APX, ATP, and Utility

Environment	Algorithm	Total time not in communication [ms]	Total distance traveled [px]
Alabama	**APX1**	247.2	12349.07
	APX2	380.8	12712.15
	APX3	550.1	15447.47
	APX4	523.3	15150.60
	APX5	659.8	15587.94
	APX6	592.6	14928.98
	ATP	**447.6**	**13689.92**
	Utility	2038.2	17747.25
Grass	**APX1**	312.8	13534.23
	APX2	420.4	14251.51
	APX3	363.6	16528.31
	APX4	413.4	16966.19
	APX5	545.1	17095.05
	APX6	429.1	15818.16
	ATP	**282.8**	**14201.22**
	Utility	2253.8	22692.65

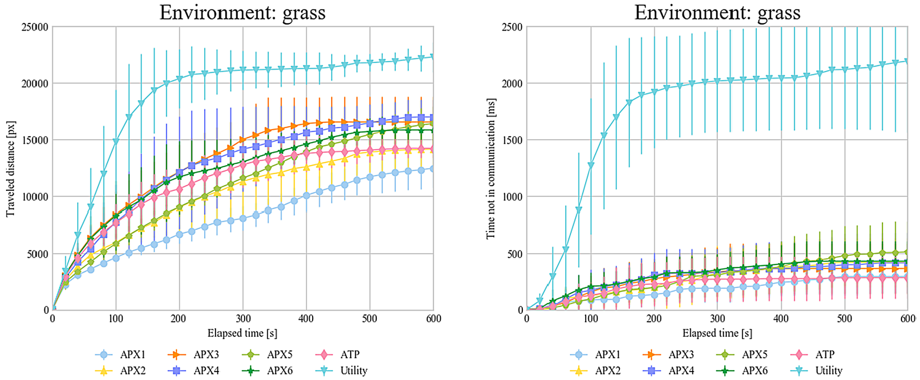
Table 2 shows the comparison between APX and ATP in total time explored, total times replanned, and average time replanned in different environments. As shown in Table 2, from the perspective of the total exploration time, the ATP algorithm has a higher exploration efficiency than most APX algorithms, regardless of whether it is a simple environment or a complex environment. Judging from the total times replanned, the total times of the ATP algorithm are significantly reduced compared with the APX1 algorithm, the total times replanned have been reduced by 45.9% and 51.7%, respectively. From the perspective of the average time replanned, the ATP algorithm is at a medium level. It can be seen that the total replanning time of the ATP algorithm is greatly reduced, allowing the UAVs to have more time for exploration and improving the efficiency of exploration.

Table 3 shows the comparison between APX, ATP, and Utility in total time not in communication and total distance traveled in different environments. As shown in Table 3, from the perspective of total time not in communication, both the ATP algorithm and the APX algorithm have greater advantages than the Utility algorithm, and the time is reduced by 87.8% to 67.6%. This shows that both the ATP algorithm and the APX algorithm can allow the BS to maintain good situational awareness. In terms of the total distance traveled, the ATP algorithm also has a significant reduction compared with the Utility algorithm. This shows that the ATP algorithm consumes less energy. Regarding the above two indicators, we plot the experimental data as shown in Fig. 10, which can more intuitively see the excellent performance of our ATP algorithm.

In addition, we can see that the APX algorithm and Utility algorithm have a certain increase in total time explored, total time not in communication, and total times replanned in a complex environment. However, the above indicators of the ATP algorithm in a complex environment perform better than in a simple



(a) Traveled distance (b) Time not in communication



(c) Traveled distance (d) Time not in communication

Fig. 10. Comparison of the planning approaches for 6 UAVs in the alabama and grass environments.

environment. This shows that our ATP algorithm has a stronger ability to adapt to complex environments.

5 Conclusion

In this work, we proposed an adaptive collaborative exploration strategy under recurrent connectivity constraints, construct an adaptive collaborative exploration framework, and designed an adaptive threshold planning algorithm. Our simulation results show that the ATP algorithm can greatly reduce the number of replanning and provide good situational awareness for the BS while ensuring a certain exploration efficiency in both complex and simple environments, and our ATP algorithm has a stronger ability to adapt to complex environments. Our future work will be as follows: One is to use deep neural networks to precategorize scene perception data for better adaptive cognition. The second is to model adaptive strategies as behaviors and use reinforcement learning methods of data-driven strategy adaptation and optimization.

References

1. Amigoni, F., Banfi, J., Basilico, N.: Multirobot exploration of communication-restricted environments: a survey. IEEE Intell. Syst. **32**(6), 48–57 (2017)
2. Arkin, R.C., Diaz, J.: Line-of-sight constrained exploration for reactive multiagent robotic teams. In: 7th International Workshop on Advanced Motion Control. Proceedings (Cat. No. 02TH8623), pp. 455–461. IEEE (2002)
3. Banfi, J., Li, A.Q., Basilico, N., Rekleitis, I., Amigoni, F.: Asynchronous multirobot exploration under recurrent connectivity constraints. In: 2016 IEEE International Conference on Robotics and Automation (ICRA), pp. 5491–5498. IEEE (2016)
4. Banfi, J., Quattrini Li, A., Rekleitis, I., Amigoni, F., Basilico, N.: Strategies for coordinated multirobot exploration with recurrent connectivity constraints. Auton. Robot. **42**(4), 875–894 (2017). https://doi.org/10.1007/s10514-017-9652-y
5. Cheng, X., Du, D.Z., Wang, L., Xu, B.: Relay sensor placement in wireless sensor networks. Wireless Netw. **14**(3), 347–355 (2008)
6. Cieslewski, T., Kaufmann, E., Scaramuzza, D.: Rapid exploration with multi-rotors: a frontier selection method for high speed flight. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2135–2142. IEEE (2017)
7. Delmerico, J., et al.: The current state and future outlook of rescue robotics. J. Field Robot. **36**(7), 1171–1191 (2019)
8. Jensen, E.A., Lowmanstone, L., Gini, M.: Communication-restricted exploration for search teams. In: Groß, R., et al. (eds.) Distributed Autonomous Robotic Systems. SPAR, vol. 6, pp. 17–30. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-73008-0_2
9. Meng, Z., et al.: A two-stage optimized next-view planning framework for 3-d unknown environment exploration, and structural reconstruction. IEEE Robot. Autom. Lett. **2**(3), 1680–1687 (2017)
10. Ochoa, S.F., Santos, R.: Human-centric wireless sensor networks to improve information availability during urban search and rescue activities. Inf. Fusion **22**, 71–84 (2015)
11. Pei, Y., Mutka, M.W., Xi, N.: Connectivity and bandwidth-aware real-time exploration in mobile robot networks. Wirel. Commun. Mob. Comput. **13**(9), 847–863 (2013)
12. Reich, J., Misra, V., Rubenstein, D., Zussman, G.: Connectivity maintenance in mobile wireless networks via constrained mobility. IEEE J. Sel. Areas Commun. **30**(5), 935–950 (2012)
13. Rooker, M.N., Birk, A.: Multi-robot exploration under the constraints of wireless networking. Control. Eng. Pract. **15**(4), 435–445 (2007)
14. Schmid, L., Pantic, M., Khanna, R., Ott, L., Siegwart, R., Nieto, J.: An efficient sampling-based method for online informative path planning in unknown environments. IEEE Robot. Autom. Lett. **5**(2), 1500–1507 (2020)
15. Selin, M., Tiger, M., Duberg, D., Heintz, F., Jensfelt, P.: Efficient autonomous exploration planning of large-scale 3-d environments. IEEE Robot. Autom. Lett. **4**(2), 1699–1706 (2019)

16. Spirin, V., Cameron, S., de Hoog, J.: Time preference for information in multi-agent exploration with limited communication. In: Natraj, A., Cameron, S., Melhuish, C., Witkowski, M. (eds.) TAROS 2013. LNCS (LNAI), vol. 8069, pp. 34–45. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-43645-5_5
17. Spirin, V., de Hoog, J., Visser, A., Cameron, S.: MRESim, a multi-robot exploration simulator for the rescue simulation league. In: Bianchi, R.A.C., Akin, H.L., Ramamoorthy, S., Sugiura, K. (eds.) RoboCup 2014. LNCS (LNAI), vol. 8992, pp. 106–117. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-18615-3_9
18. Stump, E., Michael, N., Kumar, V., Isler, V.: Visibility-based deployment of robot formations for communication maintenance. In: 2011 IEEE International Conference on Robotics and Automation, pp. 4498–4505. IEEE (2011)
19. Tadokoro, S.: Rescue Robotics: DDT Project on Robots and Systems for Urban Search and Rescue. Springer, Heidelberg (2009). https://doi.org/10.1007/978-1-84882-474-4
20. Thrun, S., et al.: Robotic mapping: a survey (2002)
21. Visser, A., Slamet, B.A.: Including communication success in the estimation of information gain for multi-robot exploration. In: 2008 6th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops, pp. 680–687. IEEE (2008)
22. Yamauchi, B.: Frontier-based exploration using multiple robots. In: Proceedings of the Second International Conference on Autonomous Agents, pp. 47–53 (1998)