drones

MDPI

Article

Onboard Distributed Trajectory Planning through Intelligent Search for Multi-UAV Cooperative Flight

Kunfeng Lu, Ruiguang Hu *, Zheng Yao and Huixia Wang

National Key Laboratory of Science and Technology on Aerospace Intelligent Control, Beijing Aerospace Automatic Control Institute, Beijing 100854, China
* Correspondence: rghu258@163.com

Abstract: Trajectory planning and obstacle avoidance play essential roles in the cooperative flight of multiple unmanned aerial vehicles (UAVs). In this paper, a unified framework for onboard distributed trajectory planning is proposed, which takes full advantage of intelligent discrete and continuous search algorithms. Firstly, the Monte Carlo tree search (MCTS) is used as the task allocation algorithm to solve the cooperative obstacle avoidance problem. Taking the task allocation decisions as the constraint, knowledge-based particle swarm optimization (Know-PSO) is used as the optimization algorithm to solve the onboard distributed cooperative trajectory planning problem. Simulation results demonstrate that the proposed intelligent MCTS-PSO search framework is effective and flexible for multiple UAVs to conduct the cooperative trajectory planning and obstacle avoidance. Further, it has been applied in practical experiments and achieved promising results.

Keywords: multiple UAVs; trajectory planning; task allocation; obstacle avoidance; intelligent search; Monte Carlo tree search; knowledge-based particle swarm optimization

Citation: Lu, K.; Hu, R.; Yao, Z.; Wang, H. Onboard Distributed Trajectory Planning through Intelligent Search for Multi-UAV Cooperative Flight. *Drones* **2023**, 7, 16. https://doi.org/10.3390/drones7010016

Academic Editors: Xiwang Dong, Mou Chen, Xiangke Wang and Fei Gao

Received: 21 November 2022
Revised: 22 December 2022
Accepted: 22 December 2022
Published: 26 December 2022

1. Introduction

Unmanned aerial vehicles (UAVs) have been extensively used in many areas, such as surveying [1–4], military surveillance [5–8], disaster rescue [9,10], etc. Although a single UAV can conduct the above-mentioned tasks, some disadvantages, including energy and others, have placed severe limitations on its applications. That is the reason why cooperative multiple UAVs have been further developed and utilized, which can overcome the disadvantages of single UAVs and accomplish tasks more robustly and intelligently with less time consumption [11].

Trajectory planning plays an essential role in the cooperative flight of multiple UAVs, which can be seen as continuous search algorithms mathematically. Commonly with some environmental constraints and UAVs' own constraints, the trajectories that minimize the cost function are seen as the best ones. Trajectory planning has been widely researched for many years, and two classes of trajectory planning methods have been developed. The first one is those traditional algorithms, including dynamic planning, the Voronoi diagram [12,13], the Dijkstra algorithm [14,15], and the A* algorithm [16]. The other class is swarm intelligence algorithms, such as the artificial bee colony (ABC) algorithm [17], the ant colony optimization (ACO) algorithm [18], and especially, the particle swarm optimization (PSO) algorithm [19–21]. In the literature [22], the trajectories of multiple UAVs were firstly given out by an offline simulation software program and saved in a text file. Then, the trajectories of all UAVs were transferred into the ground station software and displayed on a map. Finally, the trajectories were downloaded to the UAVs through the ground stations, and the UAVs completed the flight. The offline operation as in the literature [22] is now the mainstream execution style for cooperative trajectory planning.

Obstacle avoidance plays another essential role in the cooperative flight of multiple UAVs, for which the artificial potential field (APF) method is widely used [23,24]. To some extent, cooperative obstacle avoidance of multiple UAVs can be seen as a task allocation problem [25], in which multiple UAVs have to choose one side or the other to pass through the threat area. The Hungarian algorithm [22] and the auction algorithm [26,27] are two mainstream algorithms for the task allocation of multiple UAVs. In the literature [28], a two-step auction mechanism was first proposed to select the optimal action. Then, an obstacle avoidance mechanism was designed by defining several heuristic rules. Finally, a reverse auction mechanism was developed to balance the workload between multiple UAVs.

Although a large amount of effort has been devoted to addressing the trajectory planning and task allocation problems for multiple UAVs, there is still some room for further improvement. For instance, traditional task allocation algorithms often require that the input dimension must be equal to the output dimension, namely, the allocation matrix has identical row and column numbers, which severely restricts the application for more general task allocation problems. In addition, the efficiency and effectiveness of trajectory planning algorithms for multiple UAVs are not satisfactory in some circumstances. Meanwhile, the trajectory planning and task allocation for multiple UAVs are conducted offline in some literature, which cannot meet the requirements in complex and intense environments, where multiple UAVs must respond to commands rationally and quickly.

Taking the abovementioned issues into consideration, a unified framework for onboard distributed trajectory planning is proposed in this paper, which takes full advantage of intelligent discrete and continuous search algorithms. The main contributions of this paper are as follows:

(1) The Monte Carlo tree search (MCTS) is used as a task allocation algorithm to conduct obstacle avoidance, which does not require the equality of the row and column numbers of the allocation matrix. Further, the obstacle avoidance for multiple UAVs takes the energy constraint into account.
(2) Knowledge-based particle swarm optimization (Know-PSO) is used as the optimization algorithm to solve the onboard distributed cooperative trajectory planning problem, in which the motion energies of a few good particles are used to improve the velocities of those bad particles, and the information of the individual worst particles and global worst particle are also used. Furthermore, the interaction among multiple UAVs is utilized to avoid conflicts.
(3) The decisions of MCTS are taken as constraints for Know-PSO to form a unified framework for onboard distributed trajectory planning.
(4) The method proposed in this paper has been verified by actual flights and achieved good practical results.

The remainder of this paper is organized as follows: Section 2 formulates the trajectory planning and obstacle avoidance problem. Section 3 presents the proposed intelligent MCTS-PSO search framework. In Section 4, a series of simulations and actual experiments were conducted to evaluate the performance of MCTS-PSO. Finally, Section 5 concludes the paper and presents the future direction for the next work.

2. Mathematical Model

Cooperative trajectory planning driven by obstacle avoidance for multiple UAVs can be presented in the following mathematical models:

(1) Task allocation to obstacle avoidance: As shown in Figure 1, m UAVs are configured to go to the target area to conduct some important operations, in which the threat area must be avoided. Regarding each UAV, there are two choices for it to avoid the threat area, going through one side of the threat area or the other. Consequently, the cooperative obstacle avoidance problem can be translated into the task allocation problem, for which there are m decisions that have to be made. Generally, suppose

there are n choices for each UAV, then the mathematical model for task allocation is as below:

$$\begin{aligned} &\max \sum_{j=1}^{n} w_j[1-\prod_{i=1}^{m}(1-e_{ij})^{x_{ij}}] \\ &s.t.\begin{cases} \sum_{j=1}^{n} x_{ij}=1,\ i=1,2,\ldots,m \\ x_{ij}=0\ or\ 1,\ i=1,2,\ldots,m, j=1,2,\ldots n \end{cases} \end{aligned} \tag{1}$$

where w_j is the threat value of j choice, e_{ij} is the capacity evaluation of the i UAV to pass through the j choice and can be seen as the energy constraints, x_{ij} is the final decision, whether the i UAV passes through the j choice or not. The MCTS is used to solve the model to obtain x_{ij}. All x_{ij} compose the task allocation matrix, which is a 0–1 matrix and has m rows and n columns. Each row has only one, which means that the corresponding UAV can only make one choice.

(2) Cooperative trajectory planning driven by obstacle avoidance: The goal of cooperative trajectory planning is to minimize the total distances of m UAVs from the start area to the target area; in the meantime, m UAVs must avoid the threat area and not collide with each other. The mathematical model is as below:

$$\begin{aligned} &\min \sum_{i=1}^{m}\sum_{j=1}^{n} l_{ij}x_{ij} \\ &s.t.\begin{cases} dm_{i,k}>d_s,\ i,k=1,2,\ldots,m,i\neq k \\ x_{ij}=0\ or\ 1,\ i=1,2,\ldots,m,j=1,2,\ldots n \end{cases} \end{aligned} \tag{2}$$

where l_{ij} is the length of a trajectory that the i UAV pass through the j choice, $dm_{i,k}$ is the margin distance between trajectories of two different UAVs, and d_s is the safe distance between two adjacent UAVs. x_{ij} is the final decision, whether the i UAV passes through the j choice or not as in formula (1), which is solved by the MCTS method.

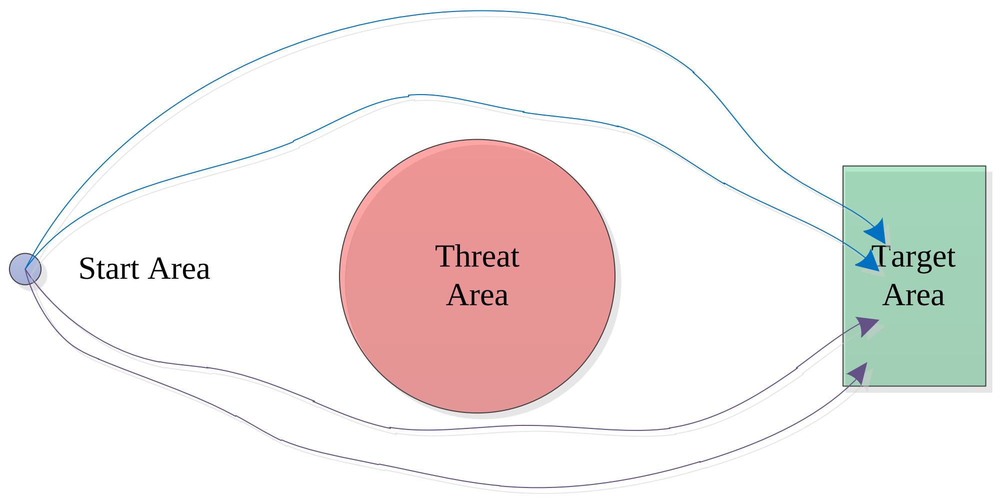

Figure 1. Representative scene of cooperative obstacle avoidance and trajectory planning.

3. MCTS-PSO Framework for Onboard Distributed Trajectory Planning

In this section, we elaborate on the cooperative trajectory planning driven by obstacle avoidance, which can be conducted in an onboard distributed mode. Firstly, MCTS is used as a task allocation algorithm to conduct obstacle avoidance, taking the energy constraint into consideration; secondly, the decisions of MCTS are taken as constraints for Know-PSO to conduct onboard distributed trajectory planning for multiple UAVs. The schematic diagram of the MCTS-PSO framework is shown in Figure 2, and the Pseudo-code of the MCTS-PSO Algorithm 1 is:

Algorithm 1: MCTS-PSO framework

Input: UAVs number m,
choices number n,
start position P_s and target position P_t,
Threat area center P_c and radius r,
the safe distance d_s,
Output: best trajectories
1: for $i \leftarrow 1$ to m do
2: for $j \leftarrow 1$ to n do
3: Evaluate the threat values w_j;
4: Evaluate the capacity values e_{ij};
5: end for
6: end for
7: Use MCTS to solve formula (1) to get decisions x_{ij};
8: Use Know-PSO to generate a trajectory for one UAV i;
9: for $k \leftarrow 1$ to m do
10: if $k == i$
11: continue;
12: else
13: Use Know-PSO to generate a trajectory for UAV k with d_s;
14: Check whether $dm_{i,k}$ is larger than d_s or not;
15: end if
16: end for
17: Return m best trajectories;
The resulting m best trajectories obtained after the above steps is the optimal solution for onboard distributed cooperative trajectory planning;
end

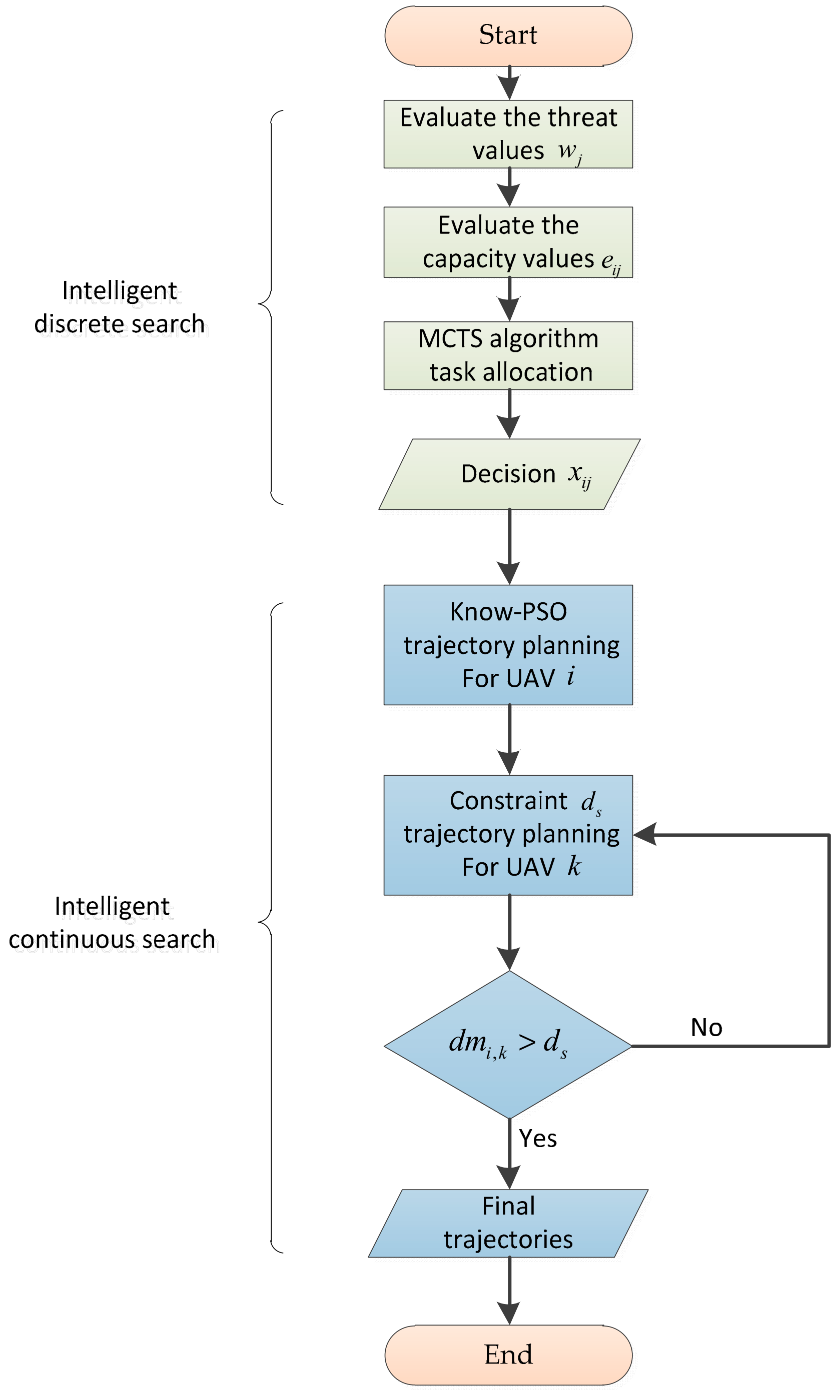

Figure 2. The schematic diagram of the MCTS-PSO framework.

3.1. MCTS Task Allocation for Multiple UAVs

Task allocation plays an essential role in the cooperative flight of multiple UAVs. In this paper, MCTS, as an intelligent discrete search algorithm, is used to conduct task allocation. MCTS does not require the equality of the row and column numbers of the allocation matrix, beyond the traditional mainstream Hungarian algorithm and auction algorithm. The flow chart of MCTS task allocation is shown in Figure 3, whose specific steps are presented as follows:

Step 1: Input UAV number, choices number, threat values, UAV capacities, and iteration number. The UAV capacities can be seen as the energy constraints:

$$e_{ij} = \frac{1}{\alpha * dist(UAV_i, choice_j)} \tag{3}$$

where α is the energy coefficient.

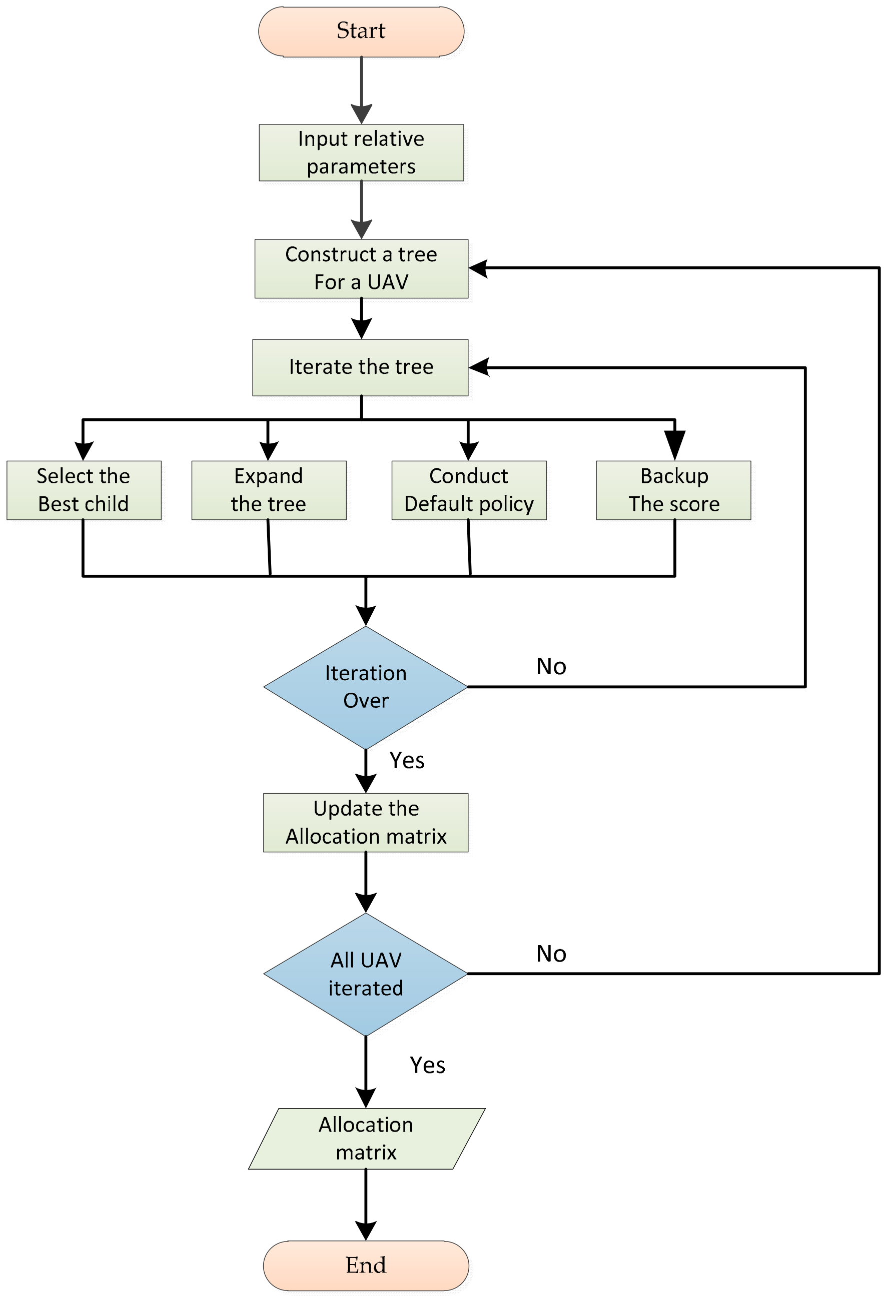

Figure 3. The flow chart of MCTS task allocation.

Step 2: Construct a new search tree and initialize the root node.
Step 3: Iterate the search tree until the iteration number:

(1) Select the best child layer by layer to find a leaf node;
(2) Expand the tree from the leaf node;
(3) Conduct the default policy;
(4) Back up the score and update nodes' attributes.

Step 4: Update the allocation matrix.
Step 5: Repeat steps 2–4 for all UAVs.

The resulting 0–1 matrix obtained after the above steps is the optimal solution for multiple UAV task allocation problems.

Pseudo-code of the MCTS task allocation Algorithm 2 is:

Algorithm 2: MCTS task allocation

Input: UAVs number m,
choices number n,
threat values w,
capacities e,
$IterNum$,
Output: Allocation matrix $AlloMx$
for $i \leftarrow 1$ to m do
Create a new tree with *root* node and initialize *root*:
$root.N \leftarrow 0$, $root.Q \leftarrow 0$;
for $j \leftarrow 1$ to $IterNum$ do
node $p \leftarrow root$;
$AlloMx\_copy \leftarrow AlloMx$;
$i\_temp \leftarrow i$;
while(True)
if p is leaf
break;
end if
find the best child of p and its index *ind*;
$p \leftarrow best\ child\ of\ p$;
$AlloMx\_copy[i\_temp][ind] \leftarrow 1$;
$i\_temp \leftarrow i\_temp + 1$;
end while
if $i\_temp \sim= m$
Expand the node p;
end if
Conduct the default policy for $AlloMx\_copy$ and get the score;
Back up the score;
end for
find the best child of *root* and its index ind_best;
$AlloMx[i][ind\_best] \leftarrow 1$;
end for
Return *AlloMx*;
The resulting 0–1 matrix obtained after the above steps is the optimal solution for task allocation;
end

3.2. Onboard Distributed Cooperative Trajectory Planning for Multiple UAVs

Trajectory planning plays another essential role in the cooperative flight of multiple UAVs. PSO, as an intelligent continuous search algorithm, has been widely used to conduct trajectory planning. Nevertheless, the standard PSO algorithm has some limitations such as premature convergence. In this paper, knowledge-based particle swarm optimization (Know-PSO) is proposed as the optimization algorithm to solve the onboard distributed cooperative trajectory planning problem, in which the motion energies of a few good particles are used to improve the velocities of those bad particles, and the information of the individual worst particles and global worst particle is also used. Furthermore, the interaction among multiple UAVs is utilized to avoid conflicts.

The well-known standard PSO algorithm is

$$\begin{cases} V_i^{k+1} = \omega V_i^k + c_1 r_1 (p_{best,i}^k - x_i^k) + c_2 r_2 (g_{best}^k - x_i^k) \\ x_i^{k+1} = x_i^k + V_i^{k+1} \end{cases} \tag{4}$$

Practically, it has been found that the information of individual worst particles and global worst particle are also beneficial, which are introduced as

$$\begin{cases} V_i^{k+1} = \omega V_i^k + c_1 r_1 (p_{best,i}^k - x_i^k) + c_2 r_2 (g_{best}^k - x_i^k) \\ \qquad\qquad -c_3 r_3 (p_{worst,i}^k - x_i^k) - c_4 r_4 (g_{worst}^k - x_i^k) \\ x_i^{k+1} = x_i^k + V_i^{k+1} \end{cases} \tag{5}$$

Further, the motion energies of a few good particles are used to improve the velocities of those bad particles, which is defined as

$$E^{k,m_1} = \sum_{i=1}^{m_1} (V_i^k)^T (V_i^k) \tag{6}$$

The energy loss is

$$\Delta E^{k,m_1} = E^{k,m_1} - E^{k-1,m_1} \tag{7}$$

Consequently, the updated equation for bad particles is

$$\begin{cases} V_i^{k+1} = \omega V_i^k + c_1 r_1 (p_{best,i}^k - x_i^k) + c_2 r_2 (g_{best}^k - x_i^k) \\ \qquad\qquad -c_3 r_3 (p_{worst,i}^k - x_i^k) - c_4 r_4 (g_{worst}^k - x_i^k) + \Delta E^{k,m_1} / m_1 \\ x_i^{k+1} = x_i^k + V_i^{k+1} \end{cases} \tag{8}$$

The decisions of MCTS are taken as constraints for Know-PSO to form a unified framework for onboard distributed trajectory planning. Specifically,

$$cost(x_{ij}) = \begin{cases} 0, & x_{ij} = 1 \\ 1000, & x_{ij} \neq 0 \end{cases} \tag{9}$$

The flow chart of Know-PSO for onboard distributed cooperative trajectory planning is shown in Figure 4, whose specific steps are presented as follows:

Step 1: Input particle number, point number, start position and target position, threat area center and radius, task allocation matrix, iteration number, UAV number, the safe distance, and the max velocity.
Step 2: Initialize the particles and best values.
Step 3: Iterate the particles until the iteration number:

(1) Compute the cost of particles;
(2) Update the best values;
(3) Update the velocities;
(4) Update the particles.

Step 4: Generate one best trajectory.
Step 5: Repeat steps 2–4 for all UAVs considering the safe distance between them.

The resulting best trajectories obtained after the above steps is the optimal solution for onboard distributed cooperative trajectory planning.

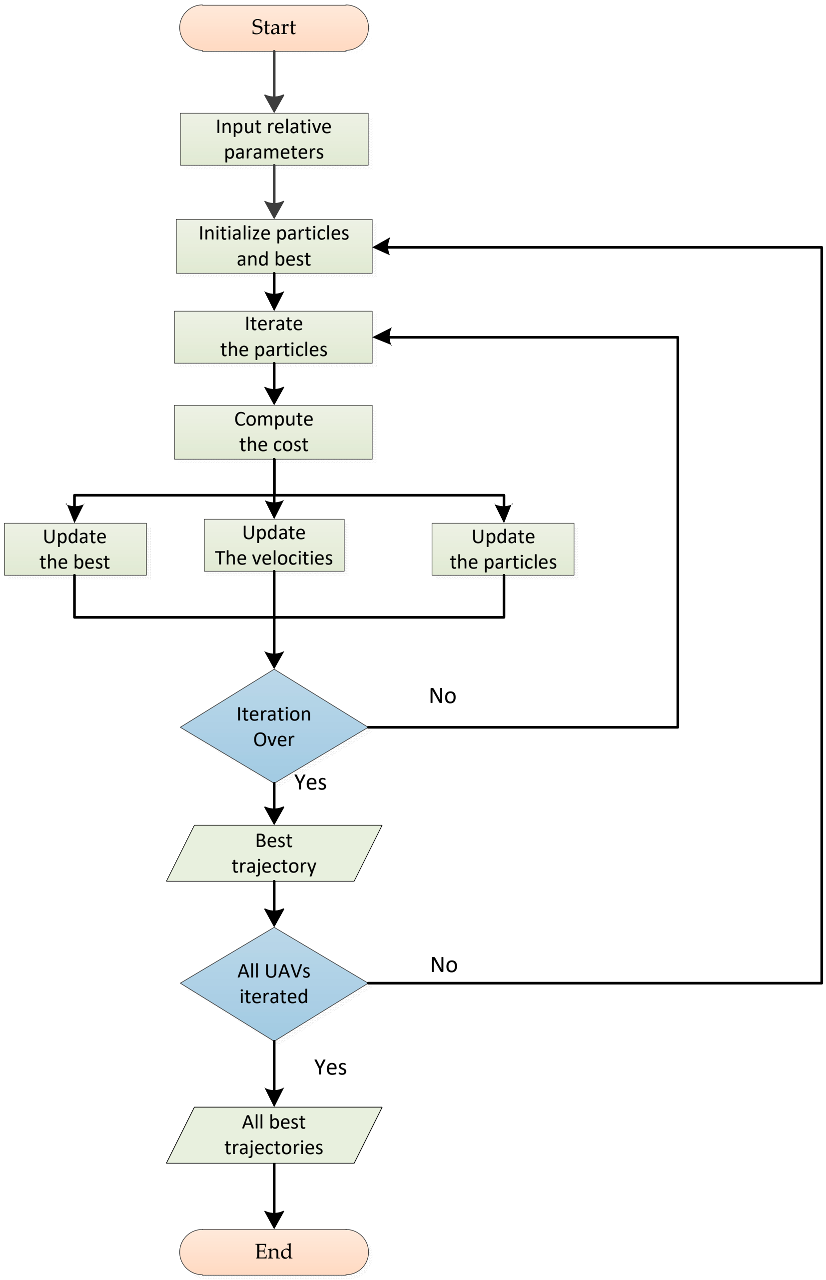

Figure 4. The flow chart of Know-PSO cooperative trajectory planning.

Pseudo-code of the cooperative trajectory planning Algorithm 3 is:

Algorithm 3: Onboard distributed cooperative trajectory planning

Input: particle number mp,
point number n,
start position P_S and target position P_t,
Threat area center P_c and radius r,
$AlloMx$,
$IterNum$,
UAVs number m,
the safe distance d_s,
V_{max},
Output: best trajectories
1: $particles \leftarrow rand(mp, n)$;
2: $particlesBest \leftarrow zeros(mp, n)$;
3: $globalBest \leftarrow zeros(1, n)$;
4: $particlesV \leftarrow rand(mp, n) * 2V_{max} - V_{max}$
5: for $iter \leftarrow 1$ to $IterNum$ do
6: for $i \leftarrow 1$ to mp do
7: Compute the $cost$ of $particles[i, :]$ with decision $AlloMx$ and P_c, r;
8: if $cost$ is descending
9: $particlesBest[i, :] \leftarrow particles[i, :]$;
10: $globalBest \leftarrow particles[i, :]$;
11: end if
12: for $j \leftarrow 1$ to n do
13: Update $particlesV[i, j]$ according Formulas (5) and (8);
14: Adjust $particlesV[i, j]$ into $[-V_{max}, V_{max}]$;
15: $particles[i, j] \leftarrow particles[i, j] + particlesV[i, j]$;
16: end for
17: end for
18: end for
19: Here, we got the best trajectory for one UAV.
20: for $i \leftarrow 2$ to m do
21: Repeat 1~21 considering the safe distance d_s;
22: end for
23: Return m best trajectories;
The resulting m best trajectories obtained after the above steps is the optimal solution for onboard distributed cooperative trajectory planning;
end

4. Experiments and Analysis

This section demonstrates the performance of the MCTS-PSO framework by conducting a series of experiments.

A cooperative processor was deployed for each UAV. The processor had a four-core CPU, whose main frequency was 1.5G Hz and `AI computational power was 20 TOPS(int8)`.

The experiments were all conducted in open environments, including plains and mountainous areas. The environmental temperature was generally higher than −20 °C.

Wind has a very important influence on flight in open environments. According to all experiments, UAVs could be controlled stably if the wind velocity was smaller than 15 m/s; otherwise, there would be some accidents.

Firstly, a flight of two drones was conducted, whose trajectories were shown in Figure 5, of which the red circle was a threat area. It can be seen that the two drones avoided the threat area successfully and passed it by one side, respectively. The task allocation matrix of them was shown in Table 1.

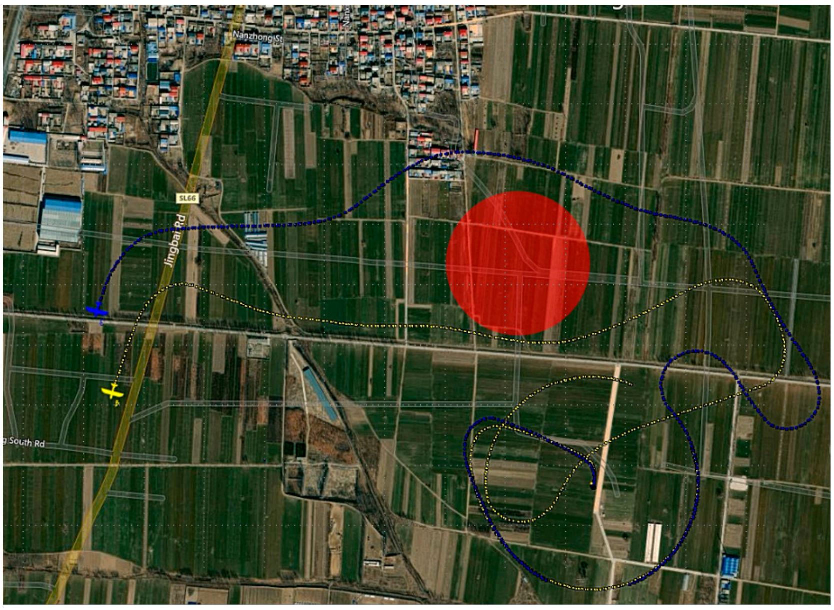

Figure 5. MCTS-PSO framework for two drones.

Table 1. Task allocation matrix for two drones in Figure 5.

	Top Channel	Bottom Channel
UAV1	1	0
UAV2	0	1

Secondly, a flight of three drones was conducted, whose trajectories were shown in Figure 6, of which the red circle was a threat area. It can be seen that the three drones avoided the threat area successfully. The task allocation matrix of them was shown in Table 2.

Thirdly, a flight of four drones was conducted in another environment, whose trajectories were shown in Figure 7, of which the red circle was a threat area. It can be seen that the four drones avoided the threat area successfully. The task allocation matrix of them was shown in Table 3.

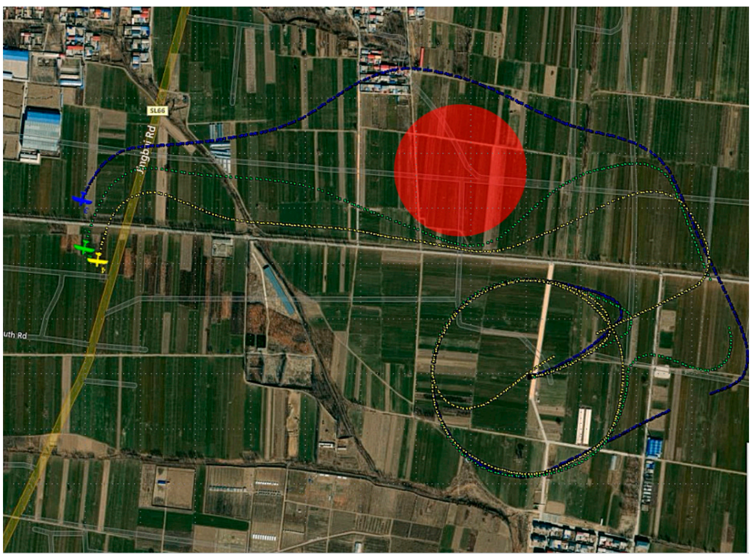

Figure 6. MCTS-PSO framework for three drones.

Table 2. Task allocation matrix for three drones in Figure 6.

	Top Channel	Bottom Channel
UAV1	1	0
UAV2	0	1
UAV3	0	1

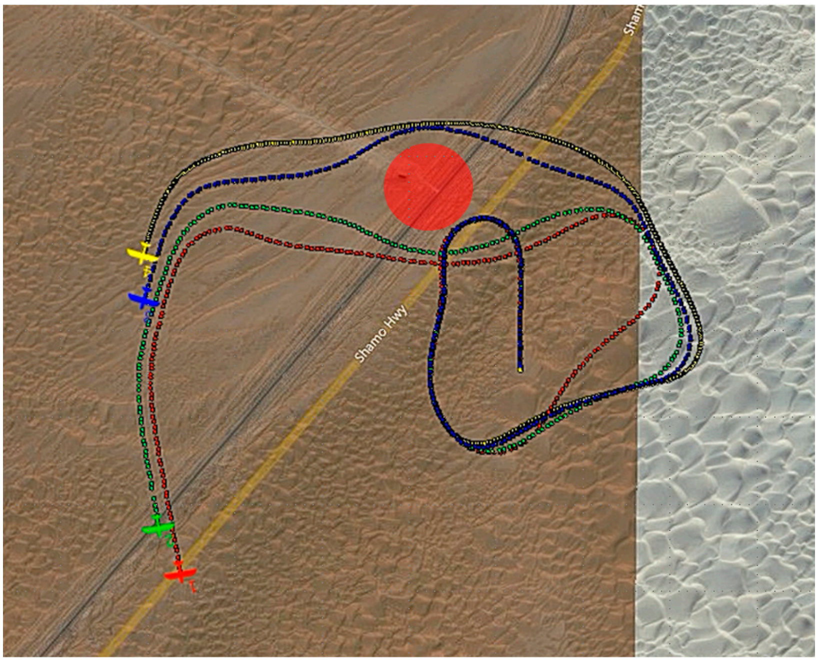

Figure 7. MCTS-PSO framework for four drones.

Table 3. Task allocation matrix for four drones in Figure 7.

	Top Channel	Bottom Channel
UAV1	1	0
UAV2	1	0
UAV3	0	1
UAV4	0	1

It was shown in these experiments, when the number of UAVs changed, the trajectories were slightly adjusted automatically. For example, comparing Figures 5 and 6, when the green UAV joined in, the trajectory of the yellow UAV was automatically adjusted to leave some space for the green one. This reflected the intelligent onboard adjustment ability of the algorithm.

Cooperative flight in an open environment was shown in Figure 8, in which five quadcopters were used. It must be noted that the MCTS-PSO framework can also be applied to other kinds of UAVs, such as fixed-wing UAVs.

Figure 8. Cooperative flying drones in open environments.

The further computational time of the proposed distributed framework and common centralized framework were shown in Table 4.

Table 4. Computational time.

	Distributed Framework	Centralized Framework
2 UAVs	0.72s	2.2s
3 UAVs	0.73s	3.1s
4 UAVs	0.73s	4.2s

It is quite clear that the proposed distributed framework was more efficient than a common centralized framework, especially when there were more UAVs, because each UAV planned its trajectory using its own processor.

The distance between the start area and the target area was about 1.4 Km. Note that the start area was where the proposed distributed framework conducted cooperative trajectory planning, not the location where the UAVs were launched. The total distances were shown in Table 5.

Table 5. Total distances.

	Total Distances
2 UAVs	3.2 Km
3 UAVs	4.9 Km
4 UAVs	6.6 Km

In open environments, other factors could also influence the real flight trajectories, such as maneuverability, wind, etc. Moreover, PSO variants could not always generate the best trajectories. These were the reasons why the trajectories in Figures 5–7 were not very close to the threat area.

Generally, the effectiveness, scalability, and adaptability of our framework were verified through the quantitative experiments with different UAV numbers. Moreover, it realized the avoidance of conflicts between multiple UAVs. All experiments demonstrated that the MCTS-PSO framework could be applied in dynamic and complex environments.

5. Conclusions

In this paper, the unified MCTS-PSO framework for onboard distributed trajectory planning is proposed, which takes full advantage of intelligent discrete and continuous search algorithms. The effectiveness, scalability, and adaptability of our framework have been verified through a series of experiments with different UAV numbers. Moreover, the proposed framework can also be applied in other similar swarm systems.

In future work, large-scale UAVs will be tested with the MCTS-PSO framework. Further, other intelligent algorithms such as multi-agent reinforcement learning will be introduced into the framework. In addition, we would like to evaluate our framework in intense confrontation applications.

Author Contributions: Conceptualization, R.H.; methodology, R.H. and K.L.; software, R.H. and Z.Y.; validation, K.L.; formal analysis, H.W.; investigation, H.W.; resources, R.H.; data curation, K.L.; writing—original draft preparation, K.L. and R.H.; writing—review and editing, Z.Y. and K.L.; visualization, K.L.; supervision, R.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank the National Key Laboratory of Science and Technology on Aerospace Intelligent Control, Beijing, China, for the resources provided by them.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Huang, R.; Zhou, H.; Liu, T.; Sheng, H. Multi-UAV collaboration to survey Tibetan antelopes in Hoh Xil. *Drones* **2022**, *6*, 196. [CrossRef]
2. Bollard, B.; Doshi, A.; Gilbert, N.; Poirot, C.; Gillman, L. Drone technology for monitoring protected areas in remote and fragile environments. *Drones* **2022**, *6*, 42. [CrossRef]
3. Ball, Z.; Odonkor, P.; Chowdhury, S. A swarm-intelligence approach to oil spill mapping using unmanned aerial vehicles. In Proceedings of the AIAA Information Systems—AIAA Infotech@ Aerospace, Grapevine, TX, USA, 9–13 January 2017.
4. Alevizos, E.; Oikonomou, D.; Argyriou, A.V.; Alexakis, D.D. Fusion of drone-Based RGB and multi-spectral imagery for shallow water bathymetry inversion. *Remote Sens.* **2022**, *14*, 1127. [CrossRef]
5. Tong, H.; Andi, T.; Huan, Z.; Dengwu, X.; Lei, X. Multiple UAV cooperative path planning based on LASSA method. *Syst. Eng. Electron.* **2022**, *44*, 233–241.
6. Pannozzi, P.; Valavanis, K.P.; Rutherford, M.J.; Guglieri, G.; Scanavino, M.; Quagliotti, F. Urban monitoring of smart communities using UAS. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems, Atlanta, GA, USA, 11–14 June 2019.
7. Ni, J.; Tang, G.; Mo, Z.; Cao, W.; Yang, S.X. An improved potential game theory based method for multi-UAV cooperative search. *IEEE Access* **2020**, *8*, 47787–47796. [CrossRef]
8. Zhang, H.; Sun, M.; Li, Q.; Liu, L.; Liu, M.; Ji, Y. An empirical study of multi-scale object detection in high resolution UAV images. *Neurocomputing* **2021**, *421*, 173–182. [CrossRef]
9. Lin, L.; Goodrich, M.A. UAV intelligent path planning for wilderness search and rescue. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009.
10. Kohlbrecher, S.; Kunz, F.; Koert, D.; Rose, C.; Manns, P.; Daun, K.; Schubert, J.; Stumpf, A.; von Stryk, O. Towards highly reliable autonomy for urban search and rescue robots. In Proceedings of the Robot Soccer World Cup, Berlin/Heidelberg, Germany, 12 May 2015.
11. McGuire, K.N.; DeWagter, C.; Tuyls, K.; Kappen, H.J.; de Croon, G.C. Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. *Sci. Robot.* **2019**, *4*, 971. [CrossRef]
12. Junlan, N.; Qingjie, Z.; Yanfen, W. UAV path planning based on weighted-Voronoi diagram. *Flight Dyn.* **2015**, *33*, 339–343.
13. Chen, X.; Chen, X. The UAV dynamic path planning algorithm research based on Voronoi diagram. In Proceedings of the 26th Chinese Control and Decision Conference (2014 CCDC), Changsha, China, 31 May–2 June 2014.
14. Singh, Y.; Sharma, S.; Sutton, R.; Hatton, D.C. Towards use of Dijkstra algorithm for optimal navigation of an unmanned surface vehicle in a real-time marine environment with results from artificial potential field. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2018**, *12*, 125–131. [CrossRef]
15. Cheng, N.; Liu, Z.; Li, Y. Path planning algorithm of Dijkstra-based intelligent aircraft under multiple constraints. *Xibei Gongye Daxue Xuebao/J. Northwest. Polytech. Univ.* **2020**, *38*, 1284–1290. [CrossRef]
16. Mandloi, D.; Arya, R.; Verma, A.K. Unmanned aerial vehicle path planning based on A * algorithm and its variants in 3d environment. *Int. J. Syst. Assur. Eng. Manag.* **2021**, *12*, 990–1000. [CrossRef]
17. Zhou, J.; Yao, X. Multi-objective hybrid artificial bee colony algorithm enhanced with Lévy flight and self-adaption for cloud manufacturing service composition. *Appl. Intell.* **2017**, *47*, 721–742. [CrossRef]

18. Xianqiang, L.I.; Rong, M.A.; Zhang, S. Improved design of ant colony algorithm and its application in path planning. *Acta Aeronaut. Astronaut. Sin.* **2020**, *41*, 724381.
19. Jiang, T.; Li, J.; Huang, K. Longitudinal parameter identification of a small unmanned aerial vehicle based on modified particle swarm optimization. *Chin. J. Aeronaut.* **2015**, *28*, 865–873. [CrossRef]
20. Liu, Y.; Zhang, X.; Guan, X.; Delahaye, D. Adaptive sensitivity decision based path planning algorithm for unmanned aerial vehicle with improved particle swarm optimization. *Aerosp. Sci. Technol.* **2016**, *58*, 92–102. [CrossRef]
21. Wenzhao, S.; Naigang, C.; Bei, H.; Xiaogang, W.; Yuliang, B. Multiple UAV cooperative path planning based on PSO-HJ method. *J. Chin. Inert. Technol.* **2020**, *28*, 122–128.
22. Zhang, J.; Sheng, H.; Chen, Q.; Zhou, H.; Yin, B.; Li, J.; Li, M. A Four-dimensional space-time automatic obstacle avoidance trajectory planning method for multi-UAV cooperative formation flight. *Drones* **2022**, *6*, 192. [CrossRef]
23. Chen, Y.B.; Luo, G.C.; Mei, Y.S.; Yu, J.Q.; Su, X.L. UAV path planning using artificial potential field method updated by optimal control theory. *Int. J. Syst. Sci.* **2016**, *47*, 1407–1420. [CrossRef]
24. Sudhakara, P.; Ganapathy, V.; Priyadharshini, B.; Sundaran, K. Obstacle avoidance and navigation planning of a wheeled mobile robot using amended artificial potential field method. *Procedia Comput. Sci.* **2018**, *133*, 998–1004. [CrossRef]
25. Khamis, A.; Hussein, A.; Elmogy, A. Multi-robot task allocation: A review of the state-of-the-art. *Coop. Robot. Sens. Netw.* **2015**, *8*, 31–51.
26. Gao, G.Q.; Xin, B. A-STC: Auction-based spanning tree coverage algorithm for motion planning of cooperative robots. *Front. Inf.Technol. Electron. Eng.* **2019**, *20*, 18–31. [CrossRef]
27. Khan, A.S.; Chen, G.; Rahulamathavan, Y.; Zheng, G.; Assadhan, B.; Lambotharan, S. Trusted UAV network coverage using blockchain, machine learning, and auction mechanisms. *IEEE Access* **2020**, *8*, 118219–118234. [CrossRef]
28. Sun, Y.; Tan, Q.; Yan, C.; Chang, Y.; Xiang, X.; Zhou, H. Multi-UAV Coverage through Two-Step Auction in Dynamic Environments. *Drones* **2022**, *6*, 153. [CrossRef]