Proceedings of the 2nd International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/67/20240598

Research on autonomous decision-making of UAV swarm based on neural network algorithm

Xin Shen

School of Electronic Science and Engineering, Southeast University, Jiangsu, China

1067199311@qq.com

Abstract. This article focuses on typical unmanned aerial vehicle (UAV) cluster autonomous collaborative reconnaissance mission scenarios, constructs UAV classes in Matlab, and effectively constructs UAV autonomous decision-making models using two different algorithms; Using Extreme Learning Machines (ELM) and introducing neural network methods to solve autonomous decision-making problems for unmanned aerial vehicles. Test the performance of the model under multiple neural network algorithms, basically achieving unmanned aerial vehicles to complete flight targets based on multiple sensor parameters in unknown environments, using reinforcement learning algorithms to achieve autonomous decision-making of unmanned aerial vehicles, achieving multi parameter fusion (with parameters greater than 4) decision-making, with a decision-making time of less than 100ms, which can ensure timely decision-making while also considering the rationality of the model.

Keywords: Neural network algorithm, Extreme learning machine, Reinforcement learning, Drones, Autonomous decision-making.

1. Introduction

1.1. Research status

In 2014, the Hungarian team Vicsek utilized a biological swarm behavior mechanism to complete mission decisions for various drones, achieving decentralized autonomous swarm flight, formation maintenance, and target tracking for 10 quadcopters. [1]

The Office of Strategic Capabilities (SCO) of the US Department of Defense led the Perdix micro drone high-speed launch demonstration project in 2014. In January 2017, the US Navy deployed 103 Perdix drones during the flight of three F/A-18F Super Hornet fighter jets, setting a record for the largest scale flight of US military drone swarms. [2]

In January 2018, during the flight test of the CODE project, SIFT successfully completed a single person in loop monitoring test for unmanned aerial vehicle cluster tasks using a unmanned aerial vehicle cluster collaborative monitoring system with autonomous decision-making capabilities. In November 2018, during a 3-week further flight test, up to 6 real and 24 virtual drone formations were executed, verifying the ability of the drones loaded with the system to respond to sudden threats and new targets with minimal human command and control in a communication limited environment. [3]

DOI: 10.54254/2755-2721/67/20240598

Wang Keliang from National University of Defense Technology introduced that reinforcement learning based on Markov decision process model can solve the sequential decision-making problem of drones in different situations, which helps to enhance the autonomous ability of drones. [4]

Ding Wei from the People's Liberation Army of China and Zhai Yiwei from the Research Institute of China Electronics Technology Group Corporation proposed a multi unmanned aerial vehicle autonomous decision-making algorithm based on deep reinforcement learning. By using relevant knowledge of reinforcement learning, the actions of unmanned aerial vehicles are planned in real-time, enabling the formation of unmanned aerial vehicles to avoid obstacles as much as possible and ultimately reach the target area. [5]

In April 2021, the Sky Borg project team in the United States used the UTAP-22 Mackerel Shark drone developed by Kratos Corporation to complete the first flight test of the Autonomous Core System, verifying a series of capabilities including navigation and flight control, envelope adaptation, and geographic fence response. [6]

1.2. Task description

At present, decision-making in domestic and foreign drone swarms focuses on decentralization, collaborative autonomy, real-time decision-making (or decision-making ability in emergency situations), as well as task allocation and trajectory planning among drone swarms. The unmanned aerial vehicle autonomous decision-making system consists of three parts: task input, task control conditions, and task output. The drone decision-making framework is shown in Figure 1.

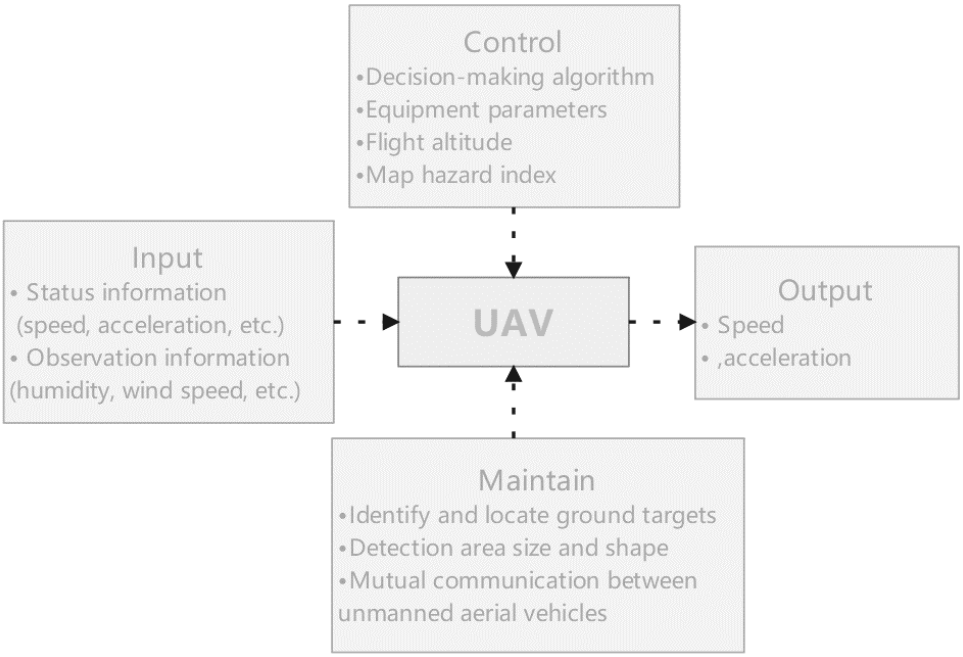

Figure 1. Drone decision framework diagram

This article aims to establish a deep reinforcement learning algorithm for achieving decentralized collaborative autonomous real-time decision-making of unmanned aerial vehicles, in order to achieve formation flight and target tracking of unmanned aerial vehicles. In this regard, we look forward to selecting the most suitable algorithm to achieve the unity of decision accuracy and decision instantaneity.

1.3. Article structure

This article constructs a framework for autonomous decision-making of unmanned aerial vehicles using two methods: extreme learning machines and reinforcement learning. In the second section, the

DOI: 10.54254/2755-2721/67/20240598

principle of extreme learning machines, the construction of unmanned aerial vehicle classes, and the implementation and results of autonomous decision-making are introduced. In the third section, the principle of reinforcement learning and the training results are introduced. In the third section, the advantages and disadvantages of the proposed model are reviewed, and future research directions are proposed.

2. Extreme learning machine

Extreme Learning Machines (ELM) algorithm is a feedforward neural network. Its network structure is similar to BP neural network, with the difference being the way parameters are adjusted. The Extreme Learning Machine consists of an input layer, a single-layer hidden layer, and an output layer. The connection weights between the input layer and the hidden layer are randomly determined and will not be changed in the future. The connection weights between the hidden layer and the output layer are obtained by solving a system of equations. Extreme learning machines can be used for fitting and classification separately. We focus on its classification function. The effectiveness of this algorithm is very good. When the number of neurons in the hidden layer is greater than the number of sample features, the simulation results after training are very close to the real situation, and the training speed and response speed are also quite high.

2.1. Operational principle

Specifically, the ELM calculation process is as follows:

$$f_L(x) = \sum_{i=1}^{L} \beta_i g_i(x) = \sum_{i=1}^{L} \beta_i g(\omega_i * x_j + b_i), j = 1, \cdots, N \quad (1)$$

Given N arbitrarily different training samples$\{(x_i, t_i)\}_{i=1}^{N}$,where$x_i = [x_{i1}, x_{i2}, \cdots, x_{in}]^T \in R^n$is the corresponding expected output vector. A standard ELM network with n input neurons, L hidden layer neurons, and m output neurons and an activation function of g (x) is represented by the following mathematical model:

$$H\beta = T$$

in which,

$$H = [h(x_1)^T, \cdots, h(x_N)^T] = \begin{bmatrix} g(\omega_1 * x_1 + b_1) & \cdots & g(\omega_L * x_1 + b_L) \\ \vdots & \ddots & \vdots \\ g(\omega_1 * x_N + b_1) & \cdots & g(\omega_L * x_N + b_L) \end{bmatrix}_{N \times L} \quad (2)$$

In ELM, H is also known as the random feature mapping matrix [7], $\omega_i = [\omega_{i1}, \omega_{i2}, \cdots \omega_{in}]^T$ represents the input weight that connects the i-th hidden layer neuron to the input neuron, and b_i represents the bias of the i-th hidden layer neuron, $\beta = [\beta_1, \beta_2, \cdots, \beta_L]^T$ represents the weight matrix between the output layer and the hidden layer, $T = [t_1, t_2, \cdots, t_N]^T$ represents the expected output matrix of the training sample. After the hidden layer neuron parameters (w_i, b_i) are randomly generated and the training sample is given based on any continuous sampling probability, the hidden layer output matrix H is actually known and remains unchanged. In this way, equation (1) is transformed into

$$\hat{\beta} = H^{-1}T \quad (3)$$

Among them, H^{-1}represents the Moore Pen pose generalized inverse of the hidden layer output matrix H.

2.2. Unmanned Aerial Vehicle (UAV)

The flight of drones is not limited by terrain and climate, and has collision avoidance function. During the model construction process, there is no need to consider the impact of drones on the range of ground detection areas. [8]

Formal description of drones in task collaboration:

UAV=<start_point, terminal_point, current_point, trans_vector, cur_ter_vector, hazard_value, track>

DOI: 10.54254/2755-2721/67/20240598

Among them, **start_point:** represents the starting position coordinate of the drone;

Terminal_point: represents the coordinates of the endpoint position of the drone;

Current point: represents the coordinate position coordinates of the drone at the current time;

Trans_vector: represents the direction of the drone's movement at the next moment;

Cur_ter_vector: represents the unit direction vector of the coordinate position coordinates and endpoint position coordinates of the drone at the current moment;

Hazard_value: represents the environmental information received by the drone sensor at the current time. Calculate the danger index of the current coordinate position of the drone and its surrounding eight grid positions from this;

Trail: The flight trajectory of a drone from its starting position to its ending position, and the coordinate position (Xi, Yi) of each movement of the drone.

2.3. Decision training

This article developed an app in Matlab to simulate the flight of drones and sensor sensing of surrounding data. A hazard index model [9] was used to generate a two-dimensional map of drone flight, with hazard indices ranging from 1 to 9, indicating an increase in danger level from low to high. Simulate the data perceived by sensors during the flight of a drone using a cellular array, where each element represents the environmental conditions near that coordinate.

Collect information from team members to simulate drone decision-making, save it as a training set, and input it into an extreme learning machine for machine learning. After training, the model performed well on the test set.

However, during the training process, the trained model may be influenced by the subjective habits of the personnel providing the training set. To solve this problem, it is necessary to find a more objective and universal training set, and establish more detailed and quantitative evaluation criteria for the quality of the model. This is a problem that future research needs to solve. [10]

In addition, the model proposed in this section is an idealized mathematical model. In this discussion, the drone swarm is considered as a particle, and the movement of the drone's position can be completed instantly. However, in practical situations, factors such as the speed, acceleration, yaw angle, and flight attitude of the drone should be considered, which are also crucial for establishing more practical and effective models.

3. Reinforcement learning enables autonomous decision-making

The problem discussed in reinforcement learning is how an agent can maximize the rewards it can receive in a complex and uncertain environment. By perceiving the state of the environment and responding to actions, better actions can be guided to achieve maximum benefits, which is called learning in interaction. This learning method is called reinforcement learning.

3.1. Operational principle

The intelligent agent model for modeling drones includes decision systems and motion systems. The decision system contains an evaluation network that can score possible actions based on the current environmental state information. The decision system selects the action with the highest score as the output action. The motion system is responsible for receiving and outputting actions, and calculating the changes in position after executing the actions. After receiving changes in location, the environmental model first transmits the new state to the agent model, and then calculates the feedback value of the new state to the agent model, and transmits the feedback value to the agent model.

The core of reinforcement learning algorithms is to obtain the basis for decision-making by solving the action evaluation scores corresponding to each state action. Different reinforcement learning algorithms can be derived based on the different ways of solving evaluation scores. This project uses the DQN algorithm and a neural network to fit the action evaluation score. The definition of action evaluation score is as follows:

DOI: 10.54254/2755-2721/67/20240598

$$q(s,a) = E\left[R_{t+1} + \gamma \max_{a \in A(S_{t+1})} q(S_{t+1}, a) \,\middle|\, S_t = s, A_t = a\right], \forall s, a \quad (4)$$

We define the error of Q value as the objective function:

$$J(\omega) = E[\left(R + \gamma \max_{a \in A(S')} q\ (S', a, \omega) - \ q\ (S, a, \omega)\right)^2]$$

In the formula, S, A, R, S' are all random variables (5)

During the training process, we have the intelligent agent collect feedback values corresponding to each state. After collecting a certain amount, we calculate the objective function using the above formula. Then, we train the neural network based on the objective function. Through multiple rounds of training, the neural network is fitted with the correct action evaluation score, thereby completing the training of the intelligent agent decision system. The pseudocode for the training process is as follows:

Pseudocode: Deep Q-learning (off-policy version)

Aim: Learn an optimal target network to approximate the optimal action values from the experience samples generated by a behavior policy π_b.

Store the experience samples generated by i in a replay buffer $B = \left\{\left(s, a, r, s'\right)\right\}$

For each iteration, do

Uniformly draw a mini-batch of samples from B

For each sample $\left(s, a, r, s'\right)$, calculate the target value as $y_T = r + \gamma \max_{a \in A(s')} \hat{q}(s, a, \omega_T)$, where ω_T is the parameter of the target network

Update the main network to minimize $\left(y_T - \hat{q}(s, a, \omega)\right)^2$ using the minibatch $\{(s, a, y_T)\}$

Set $\omega_T = \omega$ every C iterations

3.2. Training process

This project uses Matlab's deep network designer to design neural networks. The input is state and action, and the output is action evaluation score. The network structure includes input layer, long short-term network memory layer, dropout layer, fully connected network structure, and regression output layer.

Among them, the long short-term memory network is a special type of recursive neural network. This type of network is different from general feedforward neural networks, as LSTM can analyze inputs using time series; To avoid overfitting, dropout is used in layers with more parameters such as fully connected layers; In training a neural network that includes dropout layers, each batch of training data is randomly selected, essentially training multiple sub neural networks because the positions of randomly ignored weights in different sub networks are different. Finally, during the testing process, these small sub networks are combined, similar to a voting mechanism, to make predictions. When propagating forward, stopping the activation value of a neuron with a certain probability p can make the model more generalized because it does not rely too much on certain local features, thereby reducing overfitting.

Assuming the hazard coefficient at coordinates (x, y) is R, the feedback value is r, the coefficient is gamma, and the endpoint coordinates (x_m, y_m), the feedback value formula can be obtained:

$$\mathrm{R} = -R - \gamma \cdot ((x - x_m)^2 + (y - y_m)^2 \quad (6)$$

The decision system trained on Matlab takes 0.024484 seconds to make a single decision, which meets the requirements. The decision-making route can also avoid dangerous areas well, and the model has promotional significance.

4. Conclusion

This article takes autonomous decision-making of drone swarms as the research background, and constructs drone classes in Matlab for typical scenarios of autonomous collaborative reconnaissance tasks in drone swarms. We have tried extreme learning machines and reinforcement learning algorithms to basically achieve unmanned aerial vehicles to complete flight targets based on multiple

DOI: 10.54254/2755-2721/67/20240598

sensor parameters in unknown environments. We use reinforcement learning algorithms to achieve autonomous decision-making of unmanned aerial vehicles and achieve multi parameter fusion (parameters greater than 4) decision-making. A single decision takes 0.024484 seconds, and the decision time is less than 100ms, which can ensure the timeliness of decision-making and also consider the rationality of the model. The specific tasks are as follows:

(1) This article takes autonomous decision-making of drone swarms as the research background, constructs a multi drone autonomous decision-making environment, and uses ELM simulation to determine the flight route of drone swarms under limited environmental data, enabling them to safely complete flight goals. After training, they can perform well in flight on the test set.

(2) Based on the results of EML, this article improves the algorithm and optimizes the model, proposes an intelligent agent model based on reinforcement learning, builds it based on the current advanced multi-agent self synchronization cooperation method, analyzes the typical unmanned aerial vehicle autonomous collaborative reconnaissance process according to the characteristics of task factors, and proposes a unmanned aerial vehicle intelligent model based on reinforcement learning, combined with the autonomous collaborative framework of unmanned aerial vehicles. The test results show that the model can achieve multi parameter fusion (with parameters greater than 4) decision-making, and the single decision time is less than 100ms.

The paper focuses on the autonomous collaborative reconnaissance task of unmanned aerial vehicle clusters, studying the problem of collaborative interaction between unmanned aerial vehicles and achieving certain results. However, due to limited time, there are still many aspects that need to be further deepened and expanded, mainly in the following aspects:

(1) The current achievements are only in the theoretical stage and have not been simulated with programs loaded on drones. For more complex and ever-changing real-world situations, the current decision-making system may not always perform sufficiently well due to inadequate consideration.

(2) At present, drones are able to avoid dangerous areas and travel the shortest distance possible to the destination area, but they have not taken into account the possibility of obstacles in the path, which will also be a major issue when the system is put into use.

References

[1] HFD. Autonomous drones flock like birds[J]. Nature, 2014.

[2] Jia Gaowei, Hou Zhongxi Development of US drone cluster project [J]. Defense Technology, 2017 (4)

[3] Wang Tong, Ye Lei, Li Lei. Analysis of UAV Cluster Collaborative Monitoring System for the CODE Project in the United States [J]. Aviation Missile, 2019 (5): 25-29

[4] Wang Keliang. Research on Autonomous Decision Making Method for Unmanned Aerial Vehicle Cluster Adversarial Based on Reinforcement Learning [D]. National University of Defense Technology, 2020

[5] Ding Wei, Zhai Yiwei. Multi UAV Autonomous Decision Algorithm Based on Deep Reinforcement Learning [J]. Electronic Design Engineering, 2022, 30 (23)

[6] Zhu Chaolei; Yuan Cheng; Yang Jiahui; Kang Guowei. Overview of the Development of Military Unmanned Aerial Vehicle Equipment Technology in Foreign Countries in 2021 [J] Tactical Missile Technology, 2022 (1): 38-45

[7] Elaimy, Ameer; Alexander R Mackay, Wayne T Lamoreaux, Robert K Fairbanks, John J Demakas, Barton S Cooke, Benjamin J Peressini, John T Holbrook, Christopher M Lee. Multimodality treatment of brain metastases: an institutional survival analysis of 275 patients. World Journal of Surgical Oncology. 5 July 2011, 9 (69).

[8] Tonghua Xu. Research on Deep Reinforcement Learning Method for Autonomous Collaborative Reconnaissance of Drone Clusters. 2022

[9] Chen Gao. Research on Autonomous Mission Planning Methods for Multiple Unmanned Aerial Vehicles [D]. Nanjing University of Aeronautics and Astronautics, 2016

DOI: 10.54254/2755-2721/67/20240598

[10] Xudong Zhang, Shaobo Li, Chuanjiang LI, Ansi Zhang, Lei Yang. Overview of Drone Clusters: Technology, Challenges, and the Future [J/OL]. Radio Engineering, 2023