Article

Patterns

Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network

Highlights

- Individuals learn from local observations to exhibit decentralized swarm behavior
- We use reward-modulated spiking neural network for online collision avoidance learning
- Multi-drone swarm achieves self-organized, stable, and safe flight in bounded space
- Our method exhibits superior performance and stability to ANN-based methods

Authors

Feifei Zhao, Yi Zeng, Bing Han, Hongjian Fang, Zhuoya Zhao

Correspondence

yi.zeng@ia.ac.cn

In brief

The collaborative interaction mechanisms of biological swarms in nature are of great importance to inspire the study of swarm intelligence. This paper proposed a self-organizing obstacle avoidance model by drawing on the decentralized, self-organizing properties of intelligent behavior of biological swarms. Each individual independently adopts brain-inspired reinforcement learning methods to achieve online learning and makes decentralized decisions based on local observations. The proposed method enables a drone swarm to emerge with autonomous obstacle avoidance ability in bounded space.

Zhao et al., 2022, Patterns *3*, 100611
November 11, 2022
https://doi.org/10.1016/j.patter.2022.100611

CellPress

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

Patterns

CellPress
OPEN ACCESS

Article

Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network

Feifei Zhao,[1,6] Yi Zeng,[1,2,3,4,5,6,7,*] Bing Han,[1,4] Hongjian Fang,[1,3] and Zhuoya Zhao[1,3]
[1]Research Center for Brain-Inspired Intelligence, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[2]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[3]School of Future Technology, University of Chinese Academy of Sciences, Beijing, 100049 China
[4]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049 China
[5]Center for Excellence in Brain Science and Intelligence Technology, Chinese Academy of Sciences, Shanghai 200031, China
[6]These authors contributed equally
[7]Lead contact
*Correspondence: yi.zeng@ia.ac.cn
https://doi.org/10.1016/j.patter.2022.100611

THE BIGGER PICTURE Swarm behavior is widespread in nature, and the collaboration among individuals enables the group to gain more benefits. Due to the coupling influence between individual behaviors, optimization of swarm behavior is usually performed by a centralized approach, while global optimization brings a large amount of computation that is inconsistent with the mechanisms of swarm behavior in nature. In this paper, we build an obstacle avoidance model for a drone swarm inspired by the decentralized, self-organized swarm behavior mechanism in nature. Each individual adopts a reward-modulated spiking neural network to learn autonomously and makes decisions based on local observations. Eventually, the drone swarm emerges with safe flight behavior. This work shows biological plausibility in terms of learning mechanism and cognitive behavior, and it provides a basis for developing bio-inspired swarm intelligence.

Proof-of-Concept: Data science output has been formulated, implemented, and tested for one domain/problem

SUMMARY

Biological systems can exhibit intelligent swarm behavior through relatively independent individual, local interaction and decentralized decision-making. A major research challenge of self-organized swarm intelligence is the coupling influences between individual behaviors. Existing methods optimize the behavior of multiple individuals simultaneously from a global perspective. However, these methods lack in-depth inspiration from swarm behaviors in nature, so they are short of flexibly adapting to real multi-robot online decision-making tasks. To overcome such limits, this paper proposes a self-organized collision avoidance model for real drones incorporating a bio-inspired reward-modulated spiking neural network (RSNN). The local interaction and autonomous learning of a single individual leads to the emergence of swarm intelligence. We validated the proposed model on swarm collision avoidance tasks (a swarm of unmanned aerial vehicles without central control) in a bounded space, carrying out simulation and real-world experiments. Compared with artificial neural network-based online learning methods, our proposed method exhibits superior performance and better stability.

INTRODUCTION

In nature, extensively coordinated and self-organized, large-scale swarm movement behaviors exist. Group collaboration plays a vital role in the survival of organisms in nature. Honeybees collectively find good nectar sources through waggle dances.[1,2] Flocks of birds, herds of land animals, or schools of fish can spontaneously exhibit ordered patterns without collision. Their cooperative interactions also help in preying and defending against predators even in an unknown and dynamic environment.

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

Taking inspiration from the collective behaviors of biological systems in nature, swarm intelligence exhibits the characteristics of decentralization, distribution, coordination, self-organization, steady state, and the emergence of intelligence. Individual agents of swarm robotics obey with relatively simple learning ability, interacting locally with their neighbors and environment, leading to the emergence of intelligent behavior. Because of the independent decision-making without any central processing, the failure of a single robot will not affect the overall behavior, which makes the swarm system more robust and adaptive.

There has been a significant amount of work on multi-robot decision-making systems, as detailed subsequently.

Collision avoidance

Shi et al.[3] proposed a decentralized neural-swarm method for close-proximity flight of multi-robot swarms. The learning-based neural-swarm adopted a deep neural network (DNN) to predict interaction forces based on the relative positions and velocities inputs from neighboring multi-robots. DNN excels at offline training based on a large amount of training data collected from real environments. Although it can be applied to online decision-making, it often cannot effectively solve interactive few-shot online reinforcement learning tasks in real-world scenarios due to offline and online information distribution shift. Van Den Berg et al.[4,5] presented a reciprocal velocity obstacle method for multiple robots to select an action that can avoid collision with other robots. The robots need to select the preferred velocity based on the other robots' radius, current position, and current optimization velocity. This method was applied to multiple mobile robots for collision-free navigation in several challenging scenarios.[6]

Path planning

Path planning aims to reach the destination in a complex environment that may involve static and dynamic obstacles, while ensuring safe and reliable navigation. Bao et al.[7] proposed an obstacle avoidance algorithm for swarm robots based on a self-organizing migrating algorithm. The fitness function is based on the principles of attraction of targets and repulsion of obstacles to help the robot find a trajectory to move safely away from the trapped area. Based on particle swarm optimization (PSO), Biswas et al.[8] presented an obstacle avoidance and path planning method for autonomous multi-agent systems. Although these methods are relatively bio-inspired, the evolutionary optimization algorithms search the solution from amounts of randomly generated individuals, which makes them hard to be applied to real-time decision-making of actual multi-agents.

Formation maintenance

Many recent works focused on dynamically bypassing obstacles without colliding with them, while maintaining the given swarm formation. Yasin et al.[9] developed a formation maintenance algorithm with collision avoidance capability and validated it on simulated unmanned aerial vehicles (UAVs). Zhou and Schwager[10] proposed a method for a human user to teleoperate a swarm of quadrotors in an environment with obstacles. By applying multiple vector fields, the method allowed for the quadrotor swarms to maintain the desired formation, while autonomously avoiding collisions with obstacles and with each other. For maintaining a predefined formation, every agent needs to adjust its coordinates with respect to the leader or other neighboring agents. The fixed formation is so rigorous that it is very sensitive to individual failure, which may result in overall collapse.

Pattern formation

Some research focused on a self-organized morphogenetic approach for pattern formation in robot swarms. They used a gene regulatory network capable of implementing the reaction-diffusion Turing patterns as the basis for the pattern formation.[11–13]

Exploration environment

Lswarm[14] efficiently optimized a global coverage strategy in a complex 3D urban environment while avoiding collisions with static obstacles, dynamic obstacles, and other agents based on optimal reciprocal collision avoidance method. McGuire et al.[15] presented a minimal navigation algorithm that allowed a swarm of tiny flying robots to autonomously explore an unknown environment and subsequently come back to the departure point. Essentially, environmental exploration needs centralized control on multi-robots from a global perspective.

Flocking

Boids[16] described that the aggregate motion of the simulated flock results from the interaction of individual agents adhering to relatively simple rules. The rules applied in the simplest Boids include separation, alignment, and cohesion. Based on the Boids model, Alaliyat et al.[17] optimized the moving vector coefficients in the Boids model using a genetic algorithm and PSO algorithm. Based on the three general rules of Boids, Vásárhelyi[18] proposed a decentralized control framework for real drones flocking in a noisy, windy, delayed environment. Vásárhelyi[19] further incorporated CMA-ES evolutionary optimization algorithm to optimize the tunable parameters in the flocking model for a large group of autonomous flying robots navigating in confined spaces.

Collision avoidance is a fundamental problem in the study of swarm intelligence. The problem of collision avoidance has been well studied through neural network,[3] evolutionary algorithm,[7,8] and mathematical optimization.[4,5,10] However, DNN and evolutionary algorithms need to optimize a large number of parameters in the simulated agents before transplanting the learned model to the real drones, which shows relatively low adaptability to the flexible and complex environment. On the other hand, for mathematical optimization algorithms, collective safe behaviors are derived from linear programming or gradient descent from a global perspective, while individuals do not have the ability to learn safe strategies.

Swarm collision avoidance in nature is based on the individual, independent, autonomous learning ability with the combination of local interaction with neighbors. The decision-making process shows to be decentralized and self-organized. One potential approach to designing self-organized collision avoidance for real UAVs is to draw inspiration from biology. The self-organized collision avoidance model designed in this paper takes into account bio-inspired local interaction together with decentralized autonomous decision-making, as well as brain-inspired

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

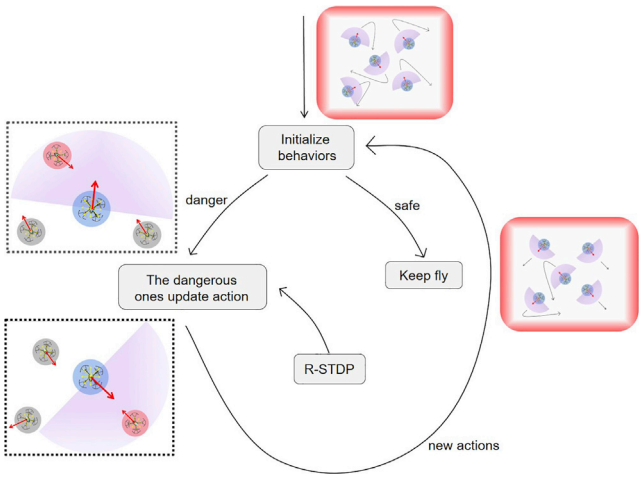

Figure 1. Decision-making process of our model

reward-modulated spiking neural network (RSNN). Different from the previous collision avoidance methods with offline pre-trained models and global mathematical calculations, our model focuses on the brain-inspired autonomous learning of a single individual for achieving decentralized and self-organized decision-making, which shows to be more biologically plausible. The main contributions of this paper can be summarized as follows:

(1) Our proposed collision avoidance model exhibits decentralized and self-organized decision-making, enabling individuals to learn from local observations efficiently and independently, so it is more suitable for real-world online drone swarm collision avoidance.
(2) We establish a brain-inspired reinforcement learning model (RSNN) with the combination of spiking neural network (SNN) and reward-modulated spike-timing-dependent plasticity (R-STDP) for online swarm collision avoidance.
(3) We conduct both simulation and real-world experiments with different numbers of agents to verify the effectiveness of our proposed method. Based on fully autonomous, decentralized, self-organized decision-making, the multi-drone swarm attains a stable and safe flight (with no collisions between agents) in bounded space and maintains it over time.

RESULTS

Decision-making process

The collision avoidance problem can be defined in the context of multiple autonomous drones flying freely in a bounded space with obstacles and other moving agents, where the drones can keep safe flights without any collision. The decision-making process of the UAVs swarm is depicted in Figure 1. The agents first randomly scattered around the scene in random flying directions. Each individual consists of an independent decision-making center (the brain-inspired reinforcement learning network, RSNN), which can autonomously learn safe flight strategies based on local observations from neighbors. When encountering danger (there are other agents that are visible, close, and approaching), the dangerous agents need to execute RSNN to optimize flight strategy (learn to choose an action from eight flight directions). RSNN learns according to the flight direction and relative position of other agents in the neighborhood that pose potential threats. In particular, we define a vertically inward direction for the boundary. Note that the speed magnitude for each agent is the same: 1/step in the x and y direction for simulated scene, and 20 cm/step in the x and y direction for real-world experiment. Thus, we will not consider updating the speed magnitude and only update the speed direction.

Essentially, multiple drones flying freely in a bounded space may result in the cooperation issue between multi-agents, since the strategy update of every signal agent may affect the response of every other agent, which makes it hard for decision-making in a decentralized way for multi-agents. Different from some global optimization methods, this paper solves the cooperation problem through completely independent individual autonomous learning based on online and real-time local observation, which shows to be more inspired by nature and biologically plausible. We perform both simulation and real-world experiments with different numbers of robots to verify the effectiveness of our proposed method.

Simulation experiments

We first implemented multi-robot collision avoidance in the simulated scene. In a simulated scene, we could perform extensive experiments to gather sufficient statistics on trends such as the relation between the performance and the number of robots. We can also observe the learning process of RSNN. As a simulation, we defined a 500 x 500 scene with 4–25 agents. The collision threshold T_{col} is set to 25, and the visible threshold T_{vis} is set to 75. Figure 2 shows four examples of the moving trajectory in the simulated bounded scene with 4, 6, 10, and 12 agents. The moving trajectories indicate that when approaching other agents and boundary, all the individuals can quickly change flight direction to avoid collision. Besides, we perform the collision avoidance experiment in the bounded space for 1 h in succession, and all agents can keep safe flight without any collision. Because RSNN is online trial-and-error learning, the decision-making process will be a little unstable and chaotic at the beginning (with no collision, just need several tries), then the multi-agent swarm gradually emerges to a steady state for long periods of safe flying.

For multi-agent collision avoidance, adding more robots leads to a higher collision probability. The number of collisions can be considered as the primary performance metric. We evaluate the number of collisions as the number of times that $\sum_j d_{ij} < T_{col}$ is satisfied in a given period of time. To verify the effectiveness of our method on different numbers of agents, we count the number of collisions between agents with 60, 65, and 70 collision threshold T_{col}, respectively. We test 5, 10, 15, 20, and 25 robots in each simulated environment. For each test configuration, 15 environments are generated for every 500 frames (approximately 30 seconds) of simulation.

The results shown in Figure 3A indicate that adding more agents leads to more collisions. This effect is mainly due to the limitation of the fixed bounded space. From Figure 3A, there

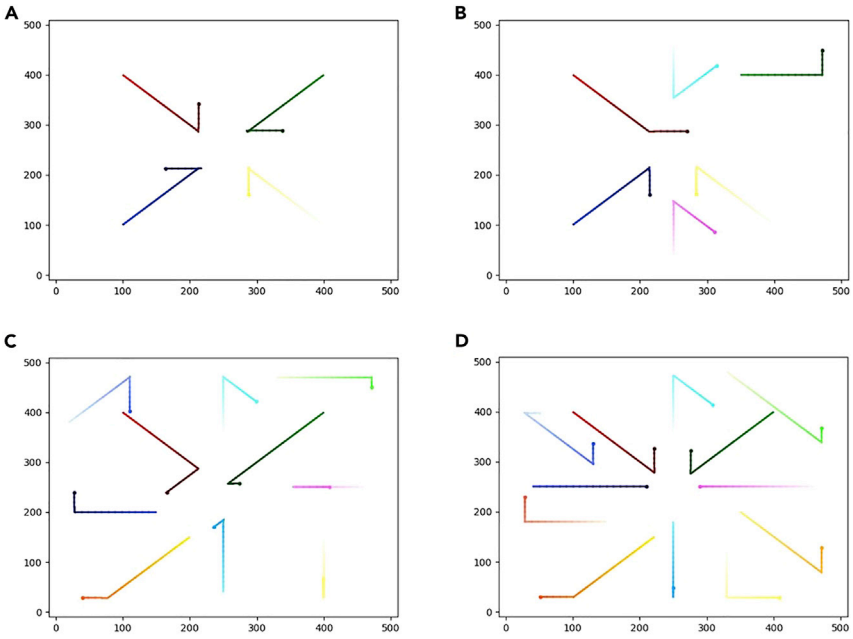

Figure 2. Results of the simulation experiments

(A–D) The moving trajectory in the simulated bounded scene with 4 (A), 6 (B), 10 (C), and 12 (D) agents.

are almost no collisions with five agents. Even for 25 agents, on average, only eight collisions occur under $T_{col} = 70$ threshold. A few collisions occur at $T_{col} = 70$, which implies that the trial-and-error learning may try the wrong action at the beginning and cause the collision at $T_{col} = 70$. For $T_{col} = 65$, there are fewer than two collisions that occur on average for different numbers of agents. For $T_{col} = 60$, there is almost no collision between the agents. The number of collisions decreases at thresholds 70, 65, and 60, suggesting that agents can gradually learn to avoid danger. In particular, since the visible threshold T_{vis} is equal to 75, five agents could learn safe strategy within 5–10 steps, and twenty-five agents could learn a safe strategy within 10–15 steps. These results demonstrate that our RSNN can quickly learn safe strategies to avoid collision.

The collision avoidance ability of collective agents is attributed to individual reinforcement learning with RSNN. We illustrate the effectiveness of RSNN by counting the change of the number of collisions between 20 agents with $T_{col} = 70$ and $T_{col} = 65$. The number of collisions is calculated on 15 experiments with each over 500 frames and fitted by linear regression, as depicted in Figure 3B. At the beginning of learning, RSNN has no preference for choosing behavior. Thus, collisions may occur. In the early stage of learning, it takes many times of trial and error to learn the correct rules. With the accumulation of experience, RSNN becomes capable of early detection of potential collisions and thus learns safe strategies for avoiding such collisions as early as possible. Therefore, with the advance of learning, the number of collisions will gradually decrease, as shown in Figure 3B. The trends of the curves demonstrate that the RSNN gradually convergences and learns the correct strategy so that the collective behavior can gradually tend to be safe at a steady state.

Real-world experiments

Subsequently, we performed real-world experiments. Particularly, the drone swarm consists of several RoboMaster Tello Talent units developed by DJI. The hardware package of the drones contains the following main modules: the vision positioning system is used for positioning. In the real-world scene, positioning is often accompanied by error, and the measured positioning error in this paper is about 10 cm; the expansion kit includes an open-source controller that supports programming with MicroPython. The open-source controller combines a 2.4/5 GHz dual-frequency Wi-Fi module for ranging to other drones and to the wireless beacon.

Considering the inevitable constraints of real drone swarms, we make some improvements to guarantee the swarm behavior. Due to the inevitable unstable transmission, the position and velocity data received by the model may be delayed and old. We solve this problem by keeping open another process to synchronously acquire the real-time position and flight direction of the drone swarm while executing the drone behavior, preventing the acquired location information from being old. In addition, we count the error of the position in the real scene (about 10 cm), and we carefully define the collision threshold between UAVs as $T_{col} = 80\,cm$ and visible threshold between UAVs as $T_{vis} = 160\,cm$ in the model, which can accommodate the

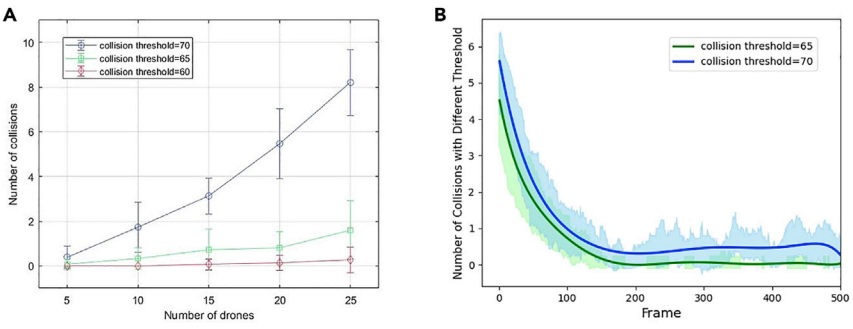

Figure 3. Simulation experiment analysis

(A) The number of collisions for different number of agents with different collision thresholds.

(B) During learning, the change of the number of collisions.

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

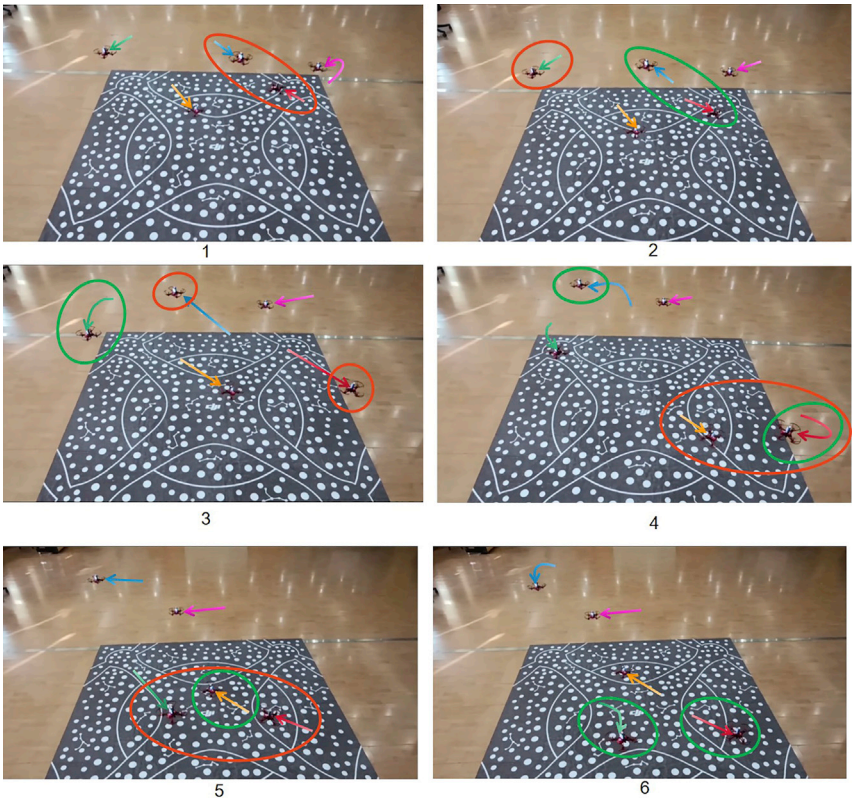

Figure 4. The flying trajectory for five UAVs collision avoidance in the real-world scene

positioning with errors and the behavior with disturbances. Considering that the real UAV swarms fly out of synchronization in time and space, the decentralized UAV cannot guarantee to reach the target position at the same time, which will affect the performance of learning. Based on the formation design of Tello Talent, we instruct the drone swarm to fly to the target position and wait for it to complete the command before executing the next command.

Real-world experiments perform collision avoidance with two, three, four, and five small UAVs (150 x 150 x 45 mm) at the same time in a bounded 3 x 3 m space. We counted the total number of collisions of about 50 tests for 5 min each time. When there are only two or three UAVs in the bounded space, online RSNN can quickly learn the correct flight strategy, and the number of collisions is less than five. However, when there are many UAVs in the bounded space (four or five UAVs), the RSNN may learn wrong strategies due to the inaccurate positioning of the real UAVs, which leads to a decline in accuracy. Moreover, due to the inaccurate movement of the UAVs swarm in real scenes, there are disturbances and errors, resulting in high requirements for the accuracy of the model. For simplicity, we adopt well-trained RSNNs that learn online in the simulation scene (with the same environment as the real-world scene), and then we transplant the well-trained RSNNs to the real scenes. They can achieve stable safe flight (no collision between UAVs) and maintain it for a long time.

Figure 4 depicts the experimental results of five UAVs' collective decision-making in bounded indoor space. The UAVs are dispersedly located in the environment at the beginning. Then they wander in the bounded space and avoid each other through the pre-trained RSNNs. Arrows with different colors in Figure 4 represent the trajectory and flight direction of the UAVs. The red circles indicate the UAVs with potential danger in the current frame. The green circles indicate the avoidance behavior of UAVs in response to the danger. The trajectories reveal that five real UAVs can avoid the boundary and each other and keep safely flying for a long time.

Attributing to the well-trained RSNNs (trained in an online manner on a simulation scene with the same environment as the real-world scene), no collision occurs when several real-world experiments are conducted, which demonstrates that the local interaction and RSNN are robust and flexible for the disturbance and variety of real scenes.

DISCUSSION

This study proposed an RSNN-based self-organized collision avoidance model for drone swarms. Our main highlight is that the swarm intelligent behaviors emerge from the decentralized individual reinforcement learning based on local interaction. The proposed model is applied to the swarm collision avoidance tasks in bounded space in both simulated and real-world scenes. Because of the behavior interaction between multiple agents, it is difficult to self-organize and achieve swarm cooperative decision-making tasks. Most existing researches deal with swarm collective decision-making through optimizing the overall behavior of the swarm from a global perspective. Neural network-based collision avoidance method[3] needs a large number of training samples to optimize the network well to apply to real drones. Mathematical optimization methods[4,5,10] considered all agents in the environment and calculated an optimal path for each agent, while the individual did not possess autonomous learning ability. Evolutionary algorithm[7,8] searched the optimal parameters for all agents through simultaneously evolving the overall swarm behaviors. To sum up, offline pre-trained and global optimization always require a large amount of time consumption and complex computation, which makes them hard to be applied to online, real-world decision-making.

Compared with the approaches used in the previous study on the multi-robot system, our collision avoidance model for drone swarms exhibits the characteristics of decentralization and self-organization, which means the emergence of collective behavior only exploiting local interactions among multi-drones, without any reference to the system as a whole. Furthermore, it shows to be more efficient because each individual leverages a light-weight RSNN to learn independently online. To ensure the self-organizing local interaction of our model, the input of RSNN is

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

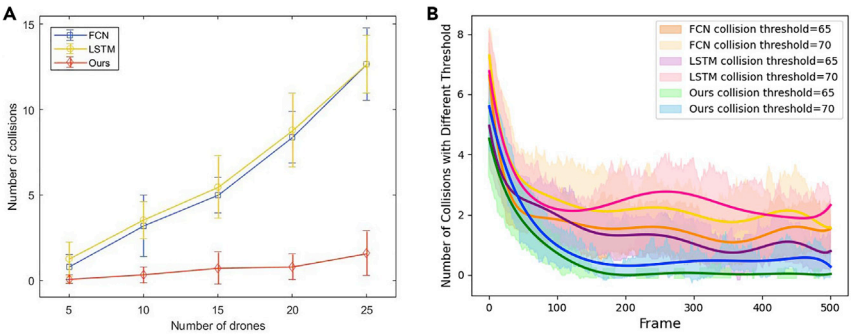

Figure 5. Comparative results on simulation experiments

(A) The number of collisions for different number of agents with different models.

(B) During learning, the change of the number of collisions with different models.

the local information from the individual's perspective. The feedback from the environment (reward function) also shows a local environmental assessment with no need to access all the individuals. Individual independent reinforcement learning also reflects the emergence of self-organized order from initial disorder. These all support our insight that only through local interaction between individuals with simple learning ability can the multi-agents emerge in collective intelligent behaviors. In summary, our proposed method shows to be more nature inspired (decentralized and self-organized decision-making), more energy efficient (small RSNNs architecture with computational efficiency to learn independently from local observation), and naturally more suitable for drone swarm collision avoidance.

Figure 5 illustrates the comparison of the number of collisions for different online learning methods including long short-term memory (LSTM)[20] network, fully connected network, and our method. The experimental setup and procedure are the same as those in Figure 3, except that the results in Figure 5A compare different models with the same collision threshold ($T_{col} = 65$). To ensure the fairness of the comparison experiments, we replace the RSNN in our model with LSTM and fully connected two-layer artificial neural network (FCN), respectively, and we keep the other strategies of local interaction and online learning unchanged. From Figure 5, we observe that our method achieves significant advantages (fewer collisions) over the LSTM and FCN models for different numbers of drones and different collision thresholds. For experiments with different number of drones, the results of FCN and LSTM are very similar, while LSTM performs the worst. The reason may be that a large amount of parameters in LSTM always results in underfitting on small sample learning tasks, which makes it hard to be applied to simple online learning tasks due to the limited information available online. The architecture of the FCN is similar to that of our RSNN model, with the only difference being that RSNN adopts spiking transmission and the reward-modulated learning mechanism. The superior performance achieved by our method also illustrates that SNNs with brain-inspired learning mechanisms are better than traditional back-propagation-optimized artificial neural networks.

As shown in Figure 5B, our method (green and blue lines) converges faster, with fewer collisions, and is more stable than other methods under different collision thresholds. FCN and LSTM are more unstable than our method, as reflected by the fluctuating curves and the large fluctuating variance. These conclusions reveal that our method is not only higher performance but also more stable. Therefore, we can conclude that our proposed method achieves superior performance compared with other artificial neural network-based online learning methods.

This paper mainly focuses on collision avoidance for drone swarm. We will further expand to more complex cooperative and competitive decision-making tasks, such as flocking emergence, cooperative preying, formation of migration, and so on. To achieve more nature-like swarm intelligent behavior, we will take inspiration from the neural mechanism of higher cognitive function (such as cognitive theory of mind) for a multi-agent system.

EXPERIMENTAL PROCEDURES

Resource availability

Lead contact

Further information and requests for resources should be directed to and will be fulfilled by the lead contact, Dr. Yi Zeng (yi.zeng@ia.ac.cn).

Materials availability

This study did not generate new unique materials.

Data and code availability

All original code has been deposited at https://github.com/Brain-Cog-Lab/RSNN under https://doi.org/10.5281/zenodo.7045737 and is publicly available as of the date of publication. The data used in this paper are available from the code, and you can also request it from the lead contact.

Local interaction

During the decision-making process, each agent moves toward its flight direction at the same time. When danger is detected from surrounding neighbors, the dangerous agents update their velocity direction according to the neighbors' velocity direction and relative position. The individual brain-inspired reinforcement learning algorithm (RSNN) is totally decentralized and online (without any pre-training), since each individual independently learns safe flight strategies based on their local observations. Here, we explain the detailed self-organized collision avoidance method, starting with the local interaction between multi-agents and then expanding to the reward-modulated spiking neural network.

Unlike some existing models with central control on the whole multi-agent system simultaneously, we only focus on the local interaction. Individuals in a group can only perceive their neighbors within a certain local range and make a response to danger. Here we present the exact mathematical formulation of our local interaction mechanism.

Figure 6 depicts the local observation mechanism for a single individual, where interactions only occur between the visible, close, and approaching agents. For visibility, we define V_{ij} as the visibility of *ith* agent by an angle-dependent term with a cutoff at a visible range:

$$V_{ij} = \begin{cases} 1, & \alpha_j \in \{-\pi/2, \pi/2\} \\ 0, & otherwise \end{cases} \quad \text{(Equation 1)}$$

In the equation above, α_j is the angle between the velocity direction of agent i and agent j. For distance-based collision term, we first define visibility threshold T_{vis} and collision threshold T_{col}. Then, collision coefficient C_{ij} is calculated based on the Euclidean distance between agent i and j, d_{ij}, and the visibility between agent i and j, V_{ij}, as shown in Equation 2. Finally, we introduce two cutoffs at maximum T_{vis}/T_{col} and at minimum zero. Note that parameters T_{vis} and T_{col} are empirically defined in different scenarios. If $d_{ij} < T_{vis}$ and

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

CellPress
OPEN ACCESS

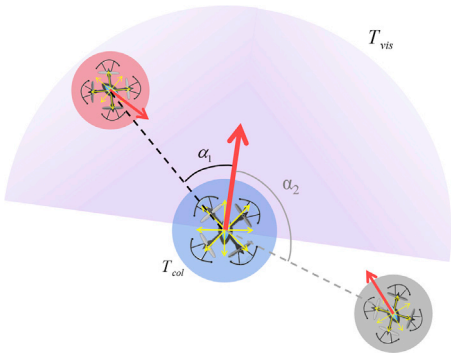

Figure 6. The local observation mechanism

$V_{ij} = 1$, agent j is visible for agent i. $d_{ij} < T_{col}$ means the collision occurred between agent i and agent j.

$$C_{ij} = V_{ij} \times \left(\frac{T_{vis} - d_{ij}(t)}{T_{col}}\right) \quad \text{(Equation 2)}$$

$$C_{ij} = \begin{cases} 0 & C_{ij} < 0 \\ C_{ij} & 0 \le C_{ij} \le T_{vis}/T_{col} \\ T_{vis}/T_{col} & C_{ij} > T_{vis}/T_{col} \end{cases} \quad \text{(Equation 3)}$$

Collision item C_{ij} clips the collision coefficient for the visible and close agents to a specific range $\{0, T_{vis}/T_{col}\}$, and the closer the distance d_{ij} is, the larger is the C_{ij}. According to the collision coefficient, danger degree D_{ij} is calculated based on the distance difference of d_{ij} between two subsequent time steps.

$$D_{ij} = C_{ij} \times (d_{ij}(t) - d_{ij}(t-1)) \quad \text{(Equation 4)}$$

For every agent i, the neighbors with $D_{ij} < 0$ will be considered as potential danger. Then agent i independently updates the strategy based on the neighbors' velocity direction and relative position. Local observations take into account visible, close, and approaching neighbors, which will directly influence a decentralized decision-making process.

Brain-inspired reward-modulated spiking neural network

In the subsection below, we explain each individual's autonomous learning component, which is inspired by the brain's decision-making mechanism. Collective intelligent behavior emerges from the individual spontaneous learning without any central control or global information. Every agent keeps random flight and monitors the local environment at the same time. When there exists potential danger ($D_{ij} < 0$), the agent in danger will select a new direction and optimize its action strategy based on its decision-making center (the reward-modulated spiking neural network).

Reinforcement learning problem

Action

We define the action space with eight velocity directions: front, back, left, right, left front, right front, left rear, and right rear. Then the output layer consists of eight clusters of neurons, each of which uses 50 neurons to represent a velocity direction. The advantages of population coding lie in leading the network to be more stable and robust for random disturbance. For the spiking neurons in the output layer, the first group with more than half of the neurons firing will be denoted as the selected behavior.

State

The decision-making for an individual is influenced by the velocity directions and relative position of other individuals with the potential threats. The number of velocity directions is eight, and the relative position contains right and left. Thus, the number of neurons in the input layer is 16. For every agent i, we choose the velocity direction and relative position of agents with $D_{ij} < 0$ as the input. The input intensity I_{ij} is linearly negatively correlated with the distance d_{ij} ($I_{ij} = -d_{ij}$); that is, the smaller the distance is, the larger is the input intensity. Then, the input intensity is limited to $\{I_{min}, I_{max}\}$, which is suitable for calculating by SNN.

Reward

Reward design is composed of the difference of value function Q_i^{RL} between two adjacent times. Value function is obtained by summing over the collision coefficient C_{ij} weighted by the influence intensity I_{ij} for the agents with $D_{ij} < 0$, as given in Equations 5 and 6. For agent i, because it is locally influenced by the neighbors, the value functions evaluated at time t and $t+1$ are both based on the neighbor agents that conform to $D_{ij}(t) < 0$ at time t, and only the C_{ij} for those agents (*jth* agents conform to $D_{ij}(t) < 0$) will be considered at time $t+1$.

$$Q_i^{RL}(t) = \sum_{j \in \{D_{ij}(t) < 0\}} I_{ij}(t) \times C_{ij}(t) \quad \text{(Equation 5)}$$

$$Q_i^{RL}(t+1) = \sum_{j \in \{D_{ij}(t) < 0\}} I_{ij}(t) \times C_{ij}(t+1) \quad \text{(Equation 6)}$$

As a result, the reward function also shows a local environmental assessment with no need to access all the individuals. In Equation 7, R_i^{RL} considers the weighted difference of the Q_i between two adjacent times for the same local neighbors. γ refers to the weighted constant, and the value of γ is set according to the value of Δe in Equation 11 (here γ is equal to 5). In addition, if the distances are much closer ($Q_i^{RL}(t+1) > Q_i^{RL}(t)$), then the punishment is larger. If the distances go farther ($Q_i^{RL}(t+1) \le Q_i^{RL}(t)$), the reward will be larger.

$$R_i^{RL} = \begin{cases} \gamma \times \left(Q_i^{RL}(t) - Q_i^{RL}(t+1)\right) & Q_i^{RL}(t+1) \le Q_i^{RL}(t) \\ -1 \times \gamma \times \left(Q_i^{RL}(t+1) - Q_i^{RL}(t)\right) & otherwise \end{cases} \quad \text{(Equation 7)}$$

Network architecture

SNN is considered as the third-generation neural network.[21] The working mechanism and learning principle of SNN are more similar to that in the human brain, such as the nonlinear accumulation of membrane potential, the discrete spike transmission between spiking neurons, and the multi-scale plasticity mechanisms, etc. These characteristics make the SNN more biologically plausible and energy efficient.[22–24]

For every individual, we adopt a two-layer small SNN (as shown in Figure 7) as the basic reinforcement learning network, not only for its biological plausibility but also for the computational efficiency. The SNN consists of two layers with full connections between the input and output layer. The input layer consists of 16 neurons that correspond to the velocity direction and the relative position. The output layer consists of eight neuron populations (corresponding to eight actions), and each output action is represented by 50 neurons. The winner-takes-all mechanism is implemented through lateral inhibition among the output populations.

The building blocks of the SNN are spiking neurons, spike-time-dependent plasticity (STDP), and reward-modulated learning rules, which will be introduced as follows.

Neuron model

Leaky integrate-and-fire (LIF)[25] is a simple and commonly used neuron model to describe the dynamic neural activities of the spiking neurons, including the dynamic changes of membrane potential and the firing process of spikes, which can be formulated as Equation 8.

$$\tau_m \frac{dv}{dt} = -v(t) + RI(t), \quad \text{(Equation 8)}$$

where $v(t)$ represents the membrane potential at time t, τ_m is the membrane time constant, and R is the membrane resistance. $I(t) = \sum_j^N w_{ji}\delta_j$ denotes the total input generated by synaptic currents triggered by the arrival of spikes of presynaptic neurons. $\delta_j = 1$ or $\delta_j = 0$ indicates the presence or absence of spiking of *jth* presynaptic neurons, respectively. When the membrane potential $v(t)$ reaches a certain threshold v_{th}, the neuron will fire a spike, and the membrane potential will be reset to v_r. Here, we set $v_{th} = 0.1$, $v_r = 0$, $\tau_m = 20$, which are consistent with those in the open-source neural simulator brian2.[26]

Please cite this article in press as: Zhao et al., Nature-inspired self-organizing collision avoidance for drone swarm based on reward-modulated spiking neural network, Patterns (2022), https://doi.org/10.1016/j.patter.2022.100611

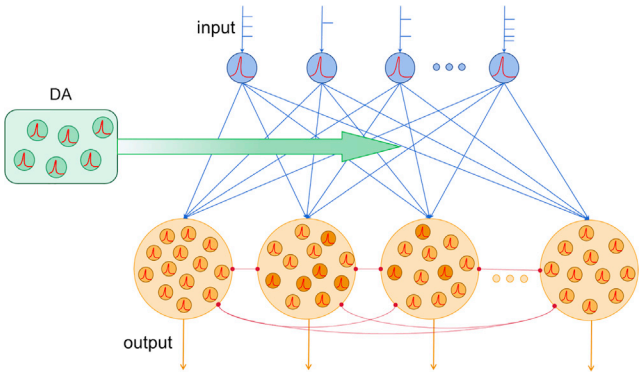

Figure 7. The architecture of reward-modulated SNN

Synaptic plasticity

STDP is a common learning principle of synaptic plasticity, which has been observed at a wide variety of excitatory and inhibitory synapses in many brain areas, both *in vitro* and *in vivo*. STDP is induced by temporal correlations between the spikes of presynaptic and postsynaptic neurons.[27–30] In STDP, repeated presynaptic action potentials arrival a few milliseconds before postsynaptic spikes induce long-term potentiation (LTP), whereas repeated spike arrival after postsynaptic spikes induces long-term depression (LTD).[31–33] When a postsynaptic spike arrives at the synapse, the weight change Δw_{STDP} is calculated as Equation 9.

$$\Delta w_{STDP} = \begin{cases} A_+ e^{(\Delta t_i/\tau_+)} & \Delta t_i < 0 \\ -A_- e^{(-\Delta t_i/\tau_-)} & \Delta t_i > 0 \end{cases} \quad \text{(Equation 9)}$$

Where, A_+ and A_- are learning rates, τ_- and τ_+ are time constant, and Δt_i is the delay time from the presynaptic spike to the postsynaptic spike. Here, we set $A_+ = 0.925$, $A_- = 0.1$, and $\tau_- = \tau_+ = 10$. The values of τ_- and τ_+ are consistent with those in Sjöström and Gerstner.[33] A_+ and A_- reflect the strength of LTP and LTD, respectively, and this paper considers the effect of LTP more.

Such a STDP rule is unsupervised, local, and instant. However, for the decision-making process, reward/punishment is delayed. Thus, the crux lies in how to build a bridge between reward/punishment neuromodulatory signals and synaptic learning rules such as STDP. To solve this problem, we adopt a reward-modulated STDP learning rule described in detail in the following subsection.

Reward-modulated learning

Biological intelligent behavior emerges spontaneously due to both local unsupervised STDP mechanism and long-term neuromodulators such as dopamine (DA) regulation. DA neurons are activated by rewarding events that are better than predicted and are depressed by events that are worse than predicted.[34–36] Due to the DA signals reward prediction error, it is considered as the regulated factor for reinforcement learning, and there has been a lot of work on combining DA regulation with STDP.[37–41] However, such DA-STDP models multiply traditional STDP by a DA term, which makes them hard to deal with credit assignment problems.

The core issue of credit assignment is which synaptic weights should be modified in order to increase the global reward for the system. Reward-modulated spike-timing-dependent plasticity has recently emerged as a candidate for a learning rule that could explain how behaviorally relevant adaptive changes in complex networks of spiking neurons could be achieved in a self-organizing manner through local synaptic plasticity.[42–48] In R-STDP, synaptic eligibility trace e is used to store a temporary memory of the STDP outcome so that it is still available when a delayed reward signal is received. The eligibility trace accumulates the change of STDP Δw_{STDP} and decays exponentially as shown in Equation 10.

$$\Delta e = -\frac{e}{\tau_e} + \Delta w_{STDP} \quad \text{(Equation 10)}$$

Where, τ_e is the time constant of the eligibility trace. Then, weight changes when the reward R_i^{RL} occurs, as shown in Equation 11.

$$\Delta w = R_i^{RL} \times \Delta e \quad \text{(Equation 11)}$$

Algorithm 1. The whole decision-making process

Initialize random action A_i^t;
  Initialize time $t = 0$;
  while *True* **do**
    for each agent i **do**
      Calculate the visibility V_{ij} from Equation 1
      Calculate the collision coefficient C_{ij} from Equations 2 and 3
      Calculate the danger degree D_{ij} from Equation 4
      if $len(D_{ij} < 0) > 0$ **then**
        %Run RSNN
        Input of RSNN: $INPUT_i = \{I_{ij}, j \in D_{ij} < 0\}$
        Output of RSNN: $A_i^t = RSNN(INPUT_i)$
      end
      Each individual executes action A_i^t
      t = t+1
      %**Update** RSNN for agent *i* who in danger
      if $len(D_{ij} < 0) > 0$ **then**
        Calculate the reward R_i^{RL} from Equations 5, 6, and 7
        Calculate the eligibility trace Δe from Equations 9 and 10
        Update RSNN from Equation 11
        Normalize the weights of RSNN;
      end
    end
  end

To sum up, this paper proposes an online collision avoidance method for drone swarm. Each individual independently uses the brain-inspired SNN to update its own strategy according to the behavior of local neighbor agents. The whole decision-making process is shown in Algorithm 1.

ACKNOWLEDGMENTS

This work is supported by the National Key Research and Development Program (No. 2020AAA0107800), the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No. XDB32070100), National Natural Science Foundation of China (Grant No. 62106261), and the Key Research Program of Frontier Sciences, Chinese Academy of Sciences (Grant No. ZDBS-LY-JSC013).

AUTHOR CONTRIBUTIONS

F.Z. and Y.Z. designed the study and experiments. F.Z., Y.Z., and H.F. contributed to the implementation of our method. F.Z., Y.Z., B.H., and Z.Z. conducted the experiments and the analyses. F.Z., Y.Z., B.H., and H.F. wrote the paper.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: May 25, 2022
Revised: August 18, 2022
Accepted: September 22, 2022
Published: October 28, 2022

REFERENCES

1. Rohrseitz, K., and Tautz, J. (1999). Honey bee dance communication: waggle run direction coded in antennal contacts? J. Comp. Physiol. Sensory Neural Behav. Physiol. *184*, 463–470.
2. Menzel, R., Fuchs, J., Kirbach, A., Lehmann, K., and Greggers, U. (2012). Navigation and communication in honey bees. In Honeybee Neurobiology and Behavior (Springer), pp. 103–116.
3. Shi, G., Hönig, W., Yue, Y., and Chung, S.J. (2020). Neural-swarm: decentralized close-proximity multirotor control using learned interactions. In 2020 IEEE International Conference on Robotics and Automation (ICRA) (IEEE), pp. 3241–3247.
4. van Den Berg, J., Guy, S.J., Lin, M., and Manocha, D. (2011). Reciprocal n-body collision avoidance. In Robotics research (Springer), pp. 3–19.
5. van den Berg, J., Lin, M., and Manocha, D. (2008). Reciprocal velocity obstacles for real-time multi-agent navigation. In 2008 IEEE International Conference on Robotics and Automation (IEEE), pp. 1928–1935.
6. Snape, J., van Den Berg, J., Guy, S.J., and Manocha, D. (2009). Independent navigation of multiple mobile robots with hybrid reciprocal velocity obstacles. In 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE), pp. 5917–5922.
7. Bao, D.Q., and Zelinka, I. (2019). Obstacle avoidance for swarm robot based on self-organizing migrating algorithm. Procedia Comput. Sci. *150*, 425–432.
8. Biswas, S., Anavatti, S.G., and Garratt, M.A. (2017). Obstacle avoidance for multi-agent path planning based on vectorized particle swarm optimization. In Intelligent and Evolutionary Systems (Springer), pp. 61–74.
9. Yasin, J.N., Haghbayan, M.H., Heikkonen, J., Tenhunen, H., and Plosila, J. (2019). Formation maintenance and collision avoidance in a swarm of drones. In Proceedings of the 2019 3rd International Symposium on Computer Science and Intelligent Control, pp. 1–6.
10. Zhou, D., and Schwager, M. (2016). Assistive collision avoidance for quadrotor swarm teleoperation. In 2016 IEEE International Conference on Robotics and Automation (ICRA) (IEEE), pp. 1249–1254.
11. Meng, Y., Guo, H., and Jin, Y. (2013). A morphogenetic approach to flexible and robust shape formation for swarm robotic systems. Robot. Autonom. Syst. *61*, 25–38.
12. Slavkov, I., Carrillo-Zapata, D., Carranza, N., Diego, X., Jansson, F., Kaandorp, J., Hauert, S., and Sharpe, J. (2018). Morphogenesis in robot swarms. Sci. Robot. *3*, eaau9178.
13. Taylor, T., Ottery, P., Hallam, J.. Pattern Formation for Multi-Robot Applications: Robust, Self-Repairing Systems Inspired by Genetic Regulatory Networks and Cellular Self-Organisation. University of Edinburgh, Tech Rep EDI-INF-RR-0971 2007;.
14. Arul, S.H., Sathyamoorthy, A.J., Patel, S., Otte, M., Xu, H., Lin, M.C., and Manocha, D. (2019). Lswarm: efficient collision avoidance for large swarms with coverage constraints in complex urban scenes. IEEE Robot. Autom. Lett. *4*, 3940–3947.
15. McGuire, K.N., De Wagter, C., Tuyls, K., Kappen, H.J., and de Croon, G.C.H.E. (2019). Minimal navigation solution for a swarm of tiny flying robots to explore an unknown environment. Sci. Robot. *4*, eaaw9710.
16. Reynolds, C.W. (1987). Flocks, herds and schools: a distributed behavioral model. In Proceedings of the 14th annual conference on Computer graphics and interactive techniques, pp. 25–34.
17. Alaliyat, S., Yndestad, H., and Sanfilippo, F. (2014). Optimisation of Boids Swarm Model Based on Genetic Algorithm and Particle Swarm Optimisation Algorithm (Comparative Study) (ECMS. Citeseer), pp. 643–650.
18. Vásárhelyi, G., Virágh, C., Somorjai, G., Tarcai, N., Szörényi, T., Nepusz, T., et al. (2014). Outdoor flocking and formation flight with autonomous aerial robots. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IEEE), pp. 3866–3873.
19. Vásárhelyi, G., Virágh, C., Somorjai, G., Nepusz, T., Eiben, A.E., and Vicsek, T. (2018). Optimized flocking of autonomous drones in confined environments. Sci. Robot. *3*, eaat3536.
20. Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. Neural Comput. *9*, 1735–1780.
21. Maass, W. (1997). Networks of spiking neurons: the third generation of neural network models. Neural Network. *10*, 1659–1671.
22. Paugam-Moisy, H., and Bohte, S. (2012). Computing with Spiking Neuron Networks (Springer Berlin Heidelberg), pp. 335–376. https://doi.org/10.1007/978-3-540-92910-9_10.
23. Maass, W. (1999). Noisy spiking neurons with temporal coding have more computational power than sigmoidal neurons. Adv. Neural Inf. Process. Syst. *9*, 211–217.
24. Bohte, S.M. (2004). The evidence for neural information processing with precise spike-times: a survey. Nat. Comput. *3*, 195–206.
25. Abbott, L.F. (1907). Lapicque's introduction of the integrate-and-fire model neuron. Brain Res. Bull. *50*, 303–304.
26. Goodman, D.F.M., and Brette, R. (2009). The brian simulator. Front. Neurosci. *3*, 192–197.
27. Bi, G.Q., and Poo, M.M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J. Neurosci. *18*, 10464–10472.
28. Bell, C.C., Han, V.Z., Sugawara, Y., and Grant, K. (1997). Synaptic plasticity in a cerebellum-like structure depends on temporal order. Nature *387*, 278–281.
29. Gerstner, W., Kempter, R., van Hemmen, J.L., and Wagner, H. (1996). A neuronal learning rule for sub-millisecond temporal coding. Nature *383*, 76–81.
30. Poo, M. (2008). Spike timing-dependent plasticity: Hebb's postulate revisited. Int. J. Dev. Neurosci. *26*, 827–828.
31. Nishiyama, M., Hong, K., Mikoshiba, K., Poo, M.M., and Kato, K. (2000). Calcium stores regulate the polarity and input specificity of synaptic modification. Nature *408*, 584–588. https://doi.org/10.1038/35046067.
32. Wittenberg, G.M., and Wang, S.S.H. (2006). Malleability of spike-timing-dependent plasticity at the ca3-ca1 synapse. J. Neurosci. *26*, 6610–6617. https://doi.org/10.1523/JNEUROSCI.5388-05.2006.
33. Sjöström, J., and Gerstner, W. (2010). Spike-timing dependent plasticity. Scholarpedia *5*, 1362.

34. Schultz, W. (1998). Predictive reward signal of dopamine neurons. J. Neurophysiol. *80*, 27.

35. Schultz, W., Dayan, P., and Montague, P.R. (1997). A neural substrate of prediction and reward. Science *275*, 1593–1599.

36. Schultz, W., Apicella, P., and Ljungberg, T. (1993). Responses of monkey dopamine neurons to reward and conditioned stimuli during successive steps of learning a delayed response task. J. Neurosci. *13*, 900–913.

37. Frank, M.J. (2005). Dynamic dopamine modulation in the basal ganglia: a neurocomputational account of cognitive deficits in medicated and non-medicated parkinsonism. J. Cogn. Neurosci. *17*, 51–72.

38. Gurney, K.N., Humphries, M.D., and Redgrave, P. (2015). A new framework for cortico-striatal plasticity: Behavioural theory meets in vitro data at the reinforcement-action interface. PLoS Biol. *13*, e1002034.

39. Doya, K. (2007). Reinforcement learning: computational theory and biological mechanisms. HFSP J. *1*, 30–40.

40. Zhao, F., Zeng, Y., and Xu, B. (2018). A brain-inspired decision-making spiking neural network and its application in unmanned aerial vehicle. Front. Neurorobot. *12*, 56.

41. Zhao, F., Zeng, Y., Guo, A., Su, H., and Xu, B. (2020). A neural algorithm for drosophila linear and nonlinear decision-makinge. Sci. Rep. *10*, 18660.

42. Zhao, Z., Lu, E., Zhao, F., Zeng, Y., and Zhao, Y. (2022). A brain-inspired theory of mind spiking neural network for reducing safety risks of other agents. Front. Neurosci. *16*, 753900.

43. Fang, H., Zeng, Y., and Zhao, F. (2021). Brain inspired sequences production by spiking neural networks with reward-modulated stdp. Front. Comput. Neurosci. *15*, 612041.

44. Frémaux, N., and Gerstner, W. (2016). Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules. Front. Neural Circuits *9*, 85.

45. Sanda, P., Skorheim, S., and Bazhenov, M. (2017). Multi-layer network utilizing rewarded spike time dependent plasticity to learn a foraging task. PLoS Comput. Biol. *13*, e1005705. https://doi.org/10.1371/journal.pcbi.1005705.

46. Izhikevich, E.M. (2007). Solving the distal reward problem through linkage of stdp and dopamine signaling. Cereb. Cortex *17*, 2443–2452.

47. Legenstein, R., Pecevski, D., and Maass, W. (2008). A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. PLoS Comput. Biol. *4*, e1000180. https://doi.org/10.1371/journal.pcbi.1000180.

48. Yan, H., Liu, X., Huo, H., and Fang, T. (2019). Mechanisms of reward-modulated stdp and winner-take-all in bayesian spiking decision-making circuit. In Proceedings of the 26th International Conference on Neural Information Processing (ICONIP), pp. 162–172.