*Article*

# Bioinspired Environment Exploration Algorithm in Swarm Based on Lévy Flight and Improved Artificial Potential Field

Chen Wang [1], Dongliang Wang [1], Minqiang Gu [1], Huaxing Huang [1], Zhaojun Wang [1], Yutong Yuan [2], Xiaomin Zhu [2], Wu Wei [3] and Zhun Fan [1,*]

[1] College of Engineering, Shantou University, Shantou 515063, China; 20cwang2@stu.edu.cn (C.W.); dlwang@stu.edu.cn (D.W.); mqgu@stu.edu.cn (M.G.); 20hxhuang@stu.edu.cn (H.H.); 17zjwang@stu.edu.cn (Z.W.)
[2] College of Systems Engineer, National University of Defense Technology, Changsha 410073, China; 17ytyuan@alumni.stu.edu.cn (Y.Y.); xmzhu@nudt.edu.cn (X.Z.)
[3] School of Automation Science and Engineering, South China University of Technology, Guangzhou 510006, China; weiwu@scut.edu.cn
* Correspondence: zfan@stu.edu.cn

**Abstract:** Inspired by the behaviour of animal populations in nature, we propose a novel exploration algorithm based on Lévy flight (LF) and artificial potential field (APF). The agent is extended to the swarm level using the APF method through the LF search environment. Virtual leaders generate moving steps to explore the environment through the LF mechanism. To achieve collision-free movement in an unknown constrained environment, a swarm-following mechanism is established, which requires the agents to follow the virtual leader to carry out the LF. The proposed method, combining the advantages of LF and APF which achieve the effect of flocking in an exploration environment, does not rely on complex sensors for environment labelling, memorising, or huge computing power. Agents simply perform elegant and efficient search behaviours as natural creatures adapt to the environment and change formations. The method is especially suitable for the camouflaged flocking exploration environment of bionic robots such as flapping drones. Simulation experiments and real-world experiments on E-puck2 robots were conducted to evaluate the effectiveness of the proposed LF-APF algorithm.

**Keywords:** bionic algorithm; swarm robotic; environment exploration; distributed control; Lévy flight

## 1. Introduction

The exploration problem is an important research area of robotics, which can be applied to various tasks such as military reconnaissance [1], search and rescue [2], foraging [3], and drug delivery [4]. In recent years, exploration using robots in complex environments has attracted widespread attention. Currently, environmental exploration methods often rely on recording the explored area or the marking of the environment using sensors. Among them, some use the odometer method [5,6] (recording the area that has been walked) and pheromone method [7,8] (marking the environment). However, the path information recorded by the agents is often subject to significant errors, and sometimes the agents have difficulties in labelling the environment. Therefore, the elimination of the agent's reliance on complex sensors when searching in an unknown environment, like natural creatures, is crucial.

The ability of a single agent in cognition and action may be inherently limited, and cooperation in a swarm can alleviate the impact of this limitation [9,10]. This kind of problem-solving ability is abundant in nature, for example, swarms of ants search for the shortest path [11] and honeybees choose the best food resources by dancing [12]. Therefore, it is specifically necessary to achieve effective coordination in a swarm robotic

system [13,14]. Multi-agent swarm search can search targets in the environment more effectively than is possible with single-agent exploration.

Many studies related to bionic UAVs have been reported in recent years [15–17]. Ramezani et al. used a series of virtual constraints to control an articulated, deformable wing to achieve autonomous flight of a bat robot [16]. EPFL [15] optimized the aerodynamic designs of winged drones for specific flight regimes. Large lifting surfaces provided manoeuvrability and agility like the northern goshawk. Roderick developed a biomimetic robot that can dynamically perch on complex surfaces and grasp irregular objects [17]. These latest studies show that research on bionic robots is crucial in the field of robotics, together with suitable applications. If we assume that a group of biomimetic robots are sent to perform a task of exploring the environment, then all of them as a group should look like birds, including their appearance and movement, but if they fail to behave like a natural cluster, the value of the stealth of the biomimetic robots is largely lost. Therefore, we propose a new swarm intelligence task, so that agents can achieve efficient environmental exploration of an area as much as possible like natural creatures without relying on complex sensors or huge computing power.

To overcome the difficulties mentioned above, random movement, which is a common search pattern for natural creatures, is introduced in this study. In some applications, the agent needs to search for dynamic targets in a complex environment through a random step generation mechanism [18]. Randomness plays a significant role in both swarm intelligence motion control and swarm intelligence optimisation algorithm [19]. In the swarm intelligence environmental exploration task in this study, due to the fact that the target is moving, if the explored path is not repeated properly, that is, if the agent does not go where it has gone, the dynamic target only needs to hide in the area where the agents passed through to avoid being detected. Therefore, the agents must traverse the area with some positions revisited occasionally. However, if a location is repeated many times, the detection efficiency decreases. Therefore, when exploring the environment with dynamic targets, agents must adopt a suitable random walk strategy to traverse an area appropriately.

Natural creatures exhibit two well-known random movement mechanisms: Lévy flight (LF) [18] and Brownian motion (BM) [20]. In animal foraging, when the prey in the environment is abundant, BM is sufficiently efficient [21]. Fredy et al. proposed BM as an exploration strategy for autonomous swarm robots [20]. This solution, to a certain extent, solves the problem of swarm robots realising environmental exploration tasks through bionic motion. As a more sophisticated alternative to BM, LF as a typical random walking strategy has been introduced in many studies [22–25], especially in the literature of agents' environmental exploration [26,27]. Vincenzo Fioriti et al. proved the LF's superiority to the random walk with simulations and applied the LF mechanism in the fish mass model's centre speed according to the Kuramoto equation [26]. Pang et al. pointed out that the mean and variance of steps generated by LF are important parameters effecting the searching efficiency and need to be optimised [25]. Even though these studies have achieved improved results, they are all about single-agent exploration. In biomimetic research of multi-agent exploration, Sutantyo et al. first presented the integration of LF and an artificial potential field method to achieve an efficient search algorithm for multiple agents applications [28]. However, in this study, the agent works in its own way and does not search for targets together with other agents. In some specific task scenarios, it may not be conducive to performing subsequent collaborative tasks such as entrapping after a single agent has discovered the target. It is often too late to call on other agents to collaborate when they are scattered too far away. In some situations, agents need to flock to be prepared to perform following swarm tasks [28]. Therefore, we need to discover how to make agents form flocks.

The artificial potential field (APF) method is widely used to realise the formation control of swarm robots while achieving collision avoidance, such as UAVs [29], wheeled mobile robots [30], and underwater robots [31]. Gabor Vásárhelyi et al. proposed an

extensible motion control framework based on an improved APF method that takes into account the motion constraints in swarms, which achieved a good flocking effect in real-world experiments, demonstrating behaviour similar to that of natural creatures [32]. Motivated by [18,32], a method combining LF and an improved APF is proposed in this work for swarm robots to form flocks and explore unknown environments with constraints effectively and efficiently.

In this study, we propose a method that combines the LF mechanism and the improved APF method to make swarms of agents flock and explore the environment in a manner similar to natural organisms. In the swarm, there is an invisible virtual leader in the arena moving with the LF algorithm. The agents follow the virtual leader in groups to find targets in an unknown environment. The swarm system randomly allocates each agent to a pre-specified priority. The leader—the agent with the highest priority in the swarm—calculates the position of the virtual leader and broadcasts the information to other agents in the swarm via WiFi. When the leader is destroyed, the agent with the highest priority in the swarm becomes the leader and continues to broadcast the virtual leader's location. An improved APF method is then applied to enable the agents in the swarm to follow the virtual leader in a flocking and to explore the environment. The agents flock to explore the environment without relying on marking the environment and recording the itinerary and achieve efficient exploration of the environment, only relying on a simple random walk mechanism, just like natural creatures. This has the potential to facilitate the stealthy mission of bionic drones. Specifically, this paper contributes the following:

(1) The proposed LF-APF algorithm applies the LF search mechanism at the swarm level. Combining the advantages of LF and APF can enable agents to efficiently explore the environment through simple and natural random walking like natural creatures.
(2) The improved APF method makes agents follow the virtual leader, maintain a certain distance from each other, and move in an orderly manner in the specified task area, autonomously changing their formations to traverse complex obstacles without colliding with them.
(3) Experimental validations on E-puck2 robots are conducted. In particular, the performance of the agent's swarm movement and the fulfilment of environmental exploration tasks are evaluated in comparative studies.

The remainder of this paper is organised as follows. In Section 2, several problems for environmental exploration tasks are defined. In Section 3, we introduce the LF algorithm as the roaming strategy. In Section 4, we describe the flocking speed controller based on an improved APF method. We conduct some simulation experiments and analyse the experimental indicators in Section 5. In Section 6, we report on the real-world experiments based on E-puck2 robots and the completion time of the experiments. Finally, Section 7 concludes the paper.

## 2. Problem Definition

The central research question in environmental exploration is how to effectively traverse an unknown area. The task of exploring the environment often requires the explorer to have superior target search capabilities and environmental coverage capabilities. In the process of executing the task, the agent needs to consider avoiding collision with other individuals in the swarm and avoid collision with obstacles or boundaries. At the same time, the swarm robot needs to follow the virtual leader. The virtual leader walks randomly with a bionic roaming strategy, and agents follow the virtual leader to achieve the effect of environmental exploration.

**Definition 1** (Repulsion)**.** *The distance between agents is maintained within a certain range, and it can be adaptively and dynamically adjusted as the environment changes. When the distance is less than $r_{arep}$, a repulsion speed is generated. Similarly, when the distance between the agent and target is less than $r_{at}$, a repulsion speed of the target is generated. When they are far apart from each other, there is no mutual repulsive speed effect.*

**Definition 2** (Avoid obstacles and walls). *Agents need to perform tasks within the specified task area. Therefore, agents cannot go out of a specific area in the process of performing tasks, which is equivalent to some virtual walls. In addition, agents need to avoid obstacles. The agent decelerates smoothly when encountering obstacles or walls to avoid colliding with them. The agent needs to slow down smoothly instead of stopping abruptly near obstacles or walls. Specifically, the closer the agent is to the them, the faster it decelerates.*

**Definition 3** (Follow the virtual leader). *All agents in the swarm follow the movement of the virtual leader. The virtual leader does not actually exist in the arena, and its position is calculated by the leader. When the agent moves to the position of the virtual leader, it needs to decelerate smoothly as the distance decreases, similar to avoiding obstacles. The closer the agent to its expected stopping point, the faster its speed should decay. When the distance is very close, its speed even needs to decay at the rate of change of the exponential function.*

**Definition 4** (Roaming strategy). *The virtual leader traverses the environment with a bionic walking strategy, and the agents in the swarm follow the virtual leader. This traversal strategy should have the following functionalities, i.e., the agents find all targets in the least possible time $t \in R$. In addition, agents travel the arena with the largest possible coverage ratio $r \in (0, 1]$.*

## 3. Roaming Strategy: Lévy Flight

LF is named after the French mathematician Paul Lévy. It refers to the random walk with a heavy-tailed distribution in the probability distribution of the step length, which means that there is a relatively high probability of large strides in the process of random walking. Natural creatures with LF mechanism tend to traverse a small place by generating many small steps and then move to another area through a large step to continue traversing to obtain higher search efficiency. The Lévy probability distribution is stable with infinite second-order moments and has the following form [18]:

$$P_{\alpha,\gamma}(l) = \frac{1}{\pi} \int_{-\infty}^{\infty} e^{-\gamma q^{\alpha}} \cos(ql) dq \tag{1}$$

The distribution is symmetric with respect to $l = 0$. The parameter $\alpha$ determines the shape of the distribution. The shorter the parameter $\alpha$ ($0 < \alpha < 2$ in Lévy distribution), the bigger the tail region. When the parameter $\alpha = 2$, the distribution changes from a Lévy distribution to a Gaussian distribution. In this study, the parameter $\alpha = 1.5$, and $\gamma$ is the scaling factor. Equation (1) can be approximated by the following expression [18]:

$$P_{\alpha}(l) \approx l^{-\alpha} \tag{2}$$

Many scholars have proposed an implementation method for generating random numbers subject to Levy distribution, including a method proposed by Mantegna in 1994 [33]. This study adopted the method proposed by Mantegna to calculate the LF step size:

$$z = \frac{u}{|v|^{1/\beta}} \tag{3}$$

where $\beta \in [0.3, 1.99]$; $u$ and $v$ are two normal stochastic variables with standard deviations $\sigma_u$ and $\sigma_v$, respectively:

$$u \sim N\left(0, \sigma_u^2\right) \quad v \sim N\left(0, \sigma_v^2\right)$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta) \sin\left(\frac{\pi\beta}{2}\right)}{\beta\Gamma\left(\frac{1+\beta}{2}\right) \cdot 2^{\frac{\beta-1}{2}}} \right\}^{\frac{1}{\beta}} \quad \sigma_v = 1 \tag{4}$$

where $\Gamma(x)$ is the gamma function:

$$\Gamma(x) = \int_0^\infty e^{-t}t^{x-1}dt \tag{5}$$

However, in practical applications, the control scale factor of the LF should be adjusted as the environment changes [33]. We can multiply the stochastic process by an appropriate multiplicative factor $\gamma^{1/\alpha}$. After the linear transformation, the result can be expressed as follows:

$$Z_s = \gamma^{1/\alpha}z \tag{6}$$

From the perspective of the Lévy probability distribution, the LF algorithm produces a large number of small step lengths and a few large step lengths. The agent traverses a local area by generating multiple small steps; a few large steps may cause the agent to jump out of the local area. Based on such a step size generation mechanism, organisms in nature can efficiently traverse the unknown environment without relying on complex sensors.

## 4. Flocking Based on Improved Artificial Potential Field Method

### 4.1. Method of Following the Virtual Leader

The leader agent in the swarm continuously calculates the position of the virtual leader and broadcasts its position to other agents in the swarm. When the agents identify a known target point, they need to move to that point in the most reasonable way possible. Gabor proposed a smooth speed decay mechanism through an ideal braking curve $D(.)$ to make their expressed motion resemble natural graceful movements, with constant acceleration at high speeds and exponential approach in time at low speeds [32].

$$D(r,a,p) = \begin{cases} 0 & \text{if } r \leq 0 \\ rp & \text{if } 0 < rp < a/p \\ \sqrt{2ar - a^2/p^2} & \text{otherwise} \end{cases} \tag{7}$$

The parameter $r$ represents the distance between an agent and the expected stopping point, $p$ gain determines the crossover point between the two phases of deceleration, and $a$ is the preferred acceleration of the agent. We introduce $D(.)$ function for the agents' tracking of the virtual leader. Here, $v_{li}$ decreases smoothly as $r_{li}$ decreases. It is easy to understand that as you get closer to your target, you may slow down and stop gradually. When you are far from the target, you need to speed up your pace and catch up. The agent can smoothly approach its target position by the following equation:

$$v_{li} = C^f \cdot D\left(r_{li}, a^f, p^f\right) \cdot \overrightarrow{r_{li}} \tag{8}$$

where $r_{li} = |r_l - r_i|$ is the distance between the agent and virtual leader, and $a^f$ is the maximal allowed acceleration in the optimal braking curve used for following the virtual leader. $p^f$ represents the gain of the optimal braking curve. If this value is too large, the braking curve exhibits a constant acceleration characteristic. When this value is small, the final part of braking (at low speeds) with decreasing acceleration is elongated and accompanied by a smooth stop. $C^f$ can linearly adjust the magnitude of the speed item of the agent following the virtual leader. Higher values assume that agents can follow the virtual leader more closely. $\overrightarrow{r_{li}} = \frac{r_l - r_i}{|r_l - r_i|}$ represents the agent's moving direction toward the virtual leader.

### 4.2. Repulsion

Agents must consider executing a task without collision when they move in swarms, like flocks of birds in the sky, which flock but rarely collide. These agents need to consider the collision avoidance between each other in the process of exploring the environment. Agents do not need to worry about individuals who are relatively far away from them

in the swarm. On the contrary, every agent should try to avoid all other agents within a certain distance at the same time, and the closer a neighbouring agent is, the stronger the repulsive effect should be from its neighbour. Agents can avoid collision with each other through the following equation:

$$
v_{ij}^{rep} = \begin{cases} p_a^{rep} \cdot (r_{arep} - r_{ij}) \cdot \overrightarrow{r_{ij}} & \text{if } (r_{ij} < r_{arep}) \\ 0 & \text{otherwise} \end{cases} \tag{9}
$$

where $r_{ij} = | r_i - r_j |$ represents the distance between agent $i$ and agent $j$, and $r_{arep}$ represents the distance threshold for the speed influence at which agents start to interact and generate repulsion. $\overrightarrow{r_{ij}}$ represents the direction of the speed from agent $j$ to agent $i$. As a linear gain, $p_a^{rep}$ linearly adjusts the size of repulsion speed term. As the agent may have multiple neighbours, it is necessary to consider the repulsive effects that may be caused by all other agents in the swarm.

$$
v_{it}^{rep} = \begin{cases} p_t^{rep} \cdot (r_{at} - r_{it}) \cdot \overrightarrow{r_{it}} & \text{if } (r_{it} < r_{at}) \\ 0 & \text{otherwise} \end{cases} \tag{10}
$$

Similar repulsion occurs between the agent and the target. $p_t^{rep}$ linearly adjusts the size of repulsion speed term. When the distance between the two ($r_{it}$) is less than the desired separation distance ($r_{at}$), the agent generates a repulsion speed away from the target. The direction of the speed is $\overrightarrow{r_{it}}$, which is from the target to agent $i$. It is worth noting that this repulsion is one-way, that is, the target will not move away from the agent due to the proximity of the agent. Superimposing the repulsion speeds leads to the speed item of the agent due to repulsion.

$$
v_i^{rep} = \sum_{j \neq i} v_{ij}^{rep} + \sum_{target} v_{it}^{rep} \tag{11}
$$

*4.3. Avoid Obstacles and Avoid Moving out of Boundaries*

In some practical tasks, we assume that the agents will explore a certain area, that is, we have defined the boundaries for the agents. They only need to explore such a specific area, and it is not necessary to explore other places. To prevent the agent from moving out of bounds, $r_{wall}$ is the safe distance between the agent and the field boundary. In other words, when the distance between the agent and the boundary is less than $r_{wall}$, the agent should produce a speed away from the boundary. We place the agents into a square-shaped arena with soft repulsive virtual walls and define virtual agents near the arena walls [32]. Virtual agents are located at the closest point of the given edge of an arbitrarily shaped convex wall polygon relative to agent $i$. When the distance between the agent and wall is less than $r_{wall}$, this speed term will take effect:

$$
v_{id}^{wall} = \begin{cases} C^s \cdot (v_{is} - D(r_{is} - r_{wall}, a^s, p^s)) \cdot \overrightarrow{v_{is}} & \text{if } (r_{is} < r_{wall}) \\ 0 & \text{otherwise} \end{cases} \tag{12}
$$

Convex obstacles inside the arena can be avoided using the same concept. When the agents are far away from the obstacle, they can ignore the influence of the obstacle on their current movement. Here, we assume that when the distance between the agent and obstacle is less than a certain value $r_{obs}$, the agent will generate the speed away from the obstacle.

$$
v_{id}^{obs} = \begin{cases} C^s \cdot (v_{is} - D(r_{is} - r_{obs}, a^s, p^s)) \cdot \overrightarrow{v_{is}} & \text{if } (r_{is} < r_{obs}) \\ 0 & \text{otherwise} \end{cases} \tag{13}
$$

In the above two equations, $r_s$ represents the position of the shill agent, located at the closest point of the given edge of an arbitrarily shaped convex wall polygon relative to agent $i$. $r_{is} = r_i - r_s$ represents the distance between agent $i$ and its closest shill agent. $v_s$ is the speed of the shill agent, pointing perpendicularly to the wall polygon edge inward of the arena ($v_{is} = |v_i - v_s|$). $\overrightarrow{v_{is}}$ is the unit vector difference between the speed of the agent and the shill agent which represents the obstacle avoidance direction of the agent after encountering obstacles. $a^s$ and $p^s$ are the same as $a^f$ and $p^f$, respectively, but for staying away from the obstacles and walls. $C^s$ adjusts the gain of the two speed terms.

*4.4. Final Equation of Desired Speed*

When the agents perform environmental exploration tasks in the unknown environment, they may encounter many complex scenarios, so the above speed influencing factors need to be considered at the same time. Agents should have all the velocities mentioned above to produce the desired motion effects. In this way, we take the vectorial sum of all the interaction terms.

$$v_i^{\text{desire}} = v_i^{rep} + v_{li} + v_{is}^{wall} + v_{is}^{obs} \tag{14}$$

Agents should meet motion constraints, that is, their speed can not be unlimited, which does not meet the needs of practical applications. When the speed generated by the above speed controller is too large, the agents should adopt a maximum speed to meet the safety requirements, but the speed direction should not be changed. In this way, after getting the speed generated by our method, we set a cut-off to cope with motion restraint. If the speed $v_i^{\text{desire}}$ is over the limit, the direction of the desired speed is maintained but its magnitude is reduced [32]:

$$\tilde{v}_i^{\text{desire}} = \frac{\tilde{v}_i^{\text{desire}}}{|\ \tilde{v}_i^{\text{desire}}\ |} \cdot \min\left\{|\ \tilde{v}_i^{\text{desire}}\ |, v_{\text{limit}}\right\} \tag{15}$$

## 5. Simulation Experiments and Analysis

*5.1. Simulation Experiments*

In this section, the performance of the proposed LF-APF method is evaluated using simulation based on MATLAB. In the simulation experiments, the agent can obtain the location information of other agents in the swarm through communication and can detect obstacles and calculate the distance from them. In addition, the agent can get the boundary position of the arena. We set the size of the arena as 250 m × 250 m. To reduce the impact of hardware computing power, we assume that the time for the agent to take a step is one second (the true time is related to the computing ability of the computer). Depending on the step size of the arena, it is necessary to adjust the size of the agent's movement. Similar to the albatross and bees in nature, although they both use the LF algorithm to search for food, the step size corresponding to the Lévy distribution should be scaled according to their different athletic abilities. To ensure fairness of comparison, we first optimise the BM step to make the agents perform as well as possible in the 250 m × 250 m arena. Then, we adjusted the parameters $\gamma$ in LF to let the median step lengths generated by the two algorithms match as closely as possible in the case of having the same size of the arena (250 m × 250 m).

As shown in Figure 1, there are four small isolating islands at sea level in the blue sea. Eight agents flocked to search for two moving targets in the complex obstacle environment. The scenario where the agents are distributed at the initial moment is shown in Figure 1a. Figure 1b shows agents in swarm having found one of the targets. Figure 1c,d shows that the agents adaptively change their formation according to the environment and pass through obstacles without any collision.

There are experimental videos for readers to watch in Appendix A. We can see that the LF-APF method has the following performance on environmental exploration tasks:

(1) The agents do not collide with each other, keep a proper distance from each other, flexibly change their formation, and shuttle in the task area, similar to a natural population.

(2) The agents can flexibly avoid isolating islands in the ocean. On some special occasions, the agents swim past obstacles in groups or pass through a limited space in a line.
(3) When the agents move near obstacles, their speed decreases smoothly, which complies more with their dynamic constraints.
(4) The agents can follow the virtual leader to achieve efficient traversal of the task area.



(**a**) t = 0                            (**b**) t = 70 steps

(**c**) t = 252 steps                   (**d**) t = 712 steps
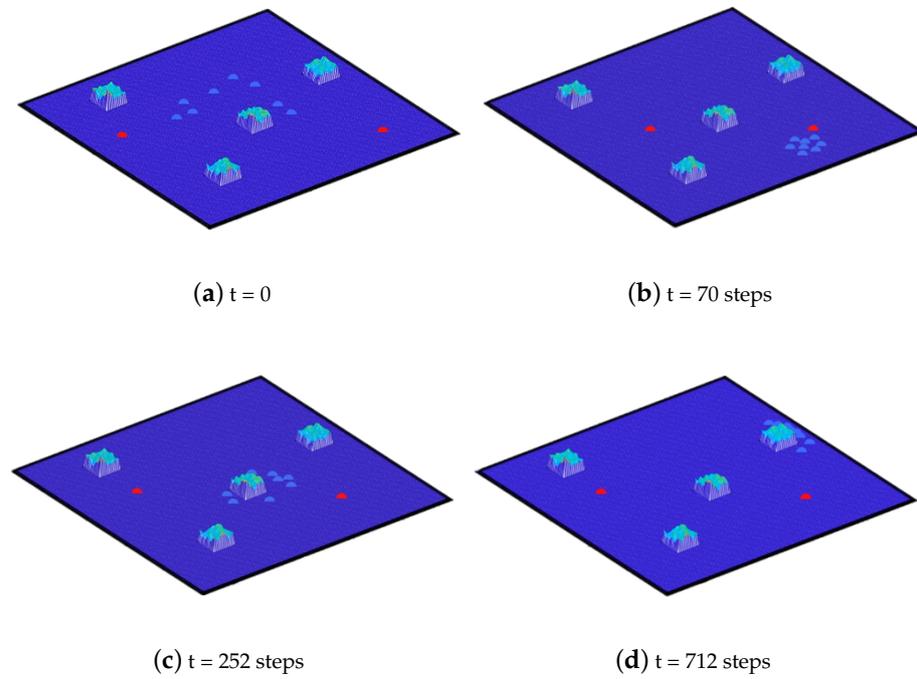
**Figure 1.** Eight agents are flocking to explore the environment with our method. There are complex unknown obstacles and two dynamic targets in the sea. Agents are blue, and targets are red. The arena is 250 m on either side. The speed of the virtual leader and targets are 1 m/s and 0.3 m/s, respectively. When the target is found (the distance between agent and target < 8 m), the target becomes stationary.

*5.2. Indicator Statistics*

In the experimental display discussed in the previous section, we can see from the figure (please also refer to the video in the Appendix A for details) that the method proposed in this study can enable a robotic swarm to achieve a good performance of environmental exploration. Since swarm robots employing BM as the exploration strategy to explore the environment scheme proposed in the previous study [20] also achieved a good environmental exploration effect, we let the swarm agents perform LF and BM for environmental exploration, respectively, and compared the results of both methods. In an unknown environment, since the target is constantly moving, it may move to any reachable place in the environment. Therefore, the evaluation index measures not only the capability of the agents to find all the targets as soon as possible [34] but also their ability to traverse the environment as much as possible. In addition, indicators of the quality of swarm movement to describe different aspects of the agents' motion are needed. The task evaluation indicators used in this study are as follows:

(1) Time for the swarm to find target;
(2) The coverage area of the swarm in a period of time;
(3) The change of agents' area coverage ratio over time;
(4) The correlation of agents' speed, the average and minimum inter-agent distances while agents are flocking.

We conducted an indicator analysis of the time to find all targets in the arena shown in Figure 2. We counted the time of identifying the target when the virtual leader runs at different speeds in 10 independent runs each. Figure 2 shows that when the agents follow

the virtual leader, with the speed of the virtual leader at 3 m/s, the agents find the targets faster. If the speed of the virtual leader is too slow, it may lead to more time for the agents to search for the targets, but if the speed is too fast, it may cause the agents to track not close enough and also lead to more time for the agents to find the target.
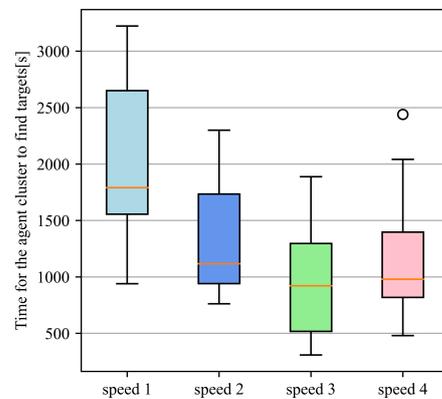


**Figure 2.** Time for the swarm to find target with LF-APF method for 10 experiments. The figure shows the virtual leader moving at different speeds (speed 1 = 1 m/s, speed 2 = 2 m/s, speed 3 = 3 m/s, speed 4 = 4 m/s) in an area of 250 m × 250 m.

To prove the advantages of LF-APF in exploring unknown regions, we compile statistics on the regions the agents walked. Let the virtual leader move at a speed of 2 m/s under two algorithms (LF and Brownian motion), and the agents follow the virtual leader to search for targets in the arena. We respectively show the coverage area in the arena with obstacles in Figure 3 and without obstacles in Figure 4. In the arena with obstacles, the obstacle areas (marked by yellow boxes) cannot be covered by the agents. Figure 5 shows how the area coverage ratio of agents varies with the time when the virtual leader moves at different speeds with two different algorithms. From Figures 3–5, it can be found that, in general, the LF has better area coverage ability than Brownian motion. At the same time, we find that when the speed of the virtual leader is too fast, the agents do not follow closely (the agents have a speed limit). If the speed is too slow, the time for the agents to traverse the environment increases. In addition, we found that when the virtual leader moves with LF at a speed of 2 m/s, the LF-APF method obtains the best area coverage ability. The above indicators proved the superiority of the LF-APF method to perform tasks in exploring unknown regions.

To make the method deploy successfully in practical applications, it is important to evaluate the effect of flocking. Considering that the speed of the agents and obstacles in the arena will affect the effect of flocking, we utilised some evaluation indicators such as the correlation of speed between agents $\phi^{\text{corr}}$ and the average and minimum of inter-agent distances ($\bar{r}_{ij}^{\min}$ and $\min(r_{ij})$) [32]. $N$ represents the number of agents in the swarm; $J_i$ represents the set of individuals in the swarm except for agent $i$. The calculation formula of $\phi^{\text{corr}}$ is as follows. We evaluated the flocking effect of LF-APF at different speeds and obstacles.

$$\phi^{\text{corr}} = \frac{1}{T} \frac{1}{N} \int_0^T \sum_{i=1}^N \frac{1}{N_i - 1} \sum_{j \in J_i} \frac{v_i \cdot v_j}{\mid v_i \mid \cdot \mid v_j \mid} dt \tag{16}$$

In Figure 6, the quality of eight agents flocking to follow the virtual leader with different speeds or obstacles are digitised. In Figure 6a,b, the results are shown when there are no obstacles in the arena. The distance between the agents is kept constant with only very minor fluctuations, and the agents' speed directions are highly correlated most of the time. At some moments, the speed correlation drops due to the virtual leader's sudden turn. When obstacles appear in the arena, the quality of the swarm motion of the agents is affected to a certain extent, as shown in Figure 6c,d. The index $\bar{r}_{ij}^{\min}$ surges in cases when

some agents are blocked by an obstacle accidentally in the arena, as shown in Figure 6c. In general, the agents achieve a relatively flocking effect safely.
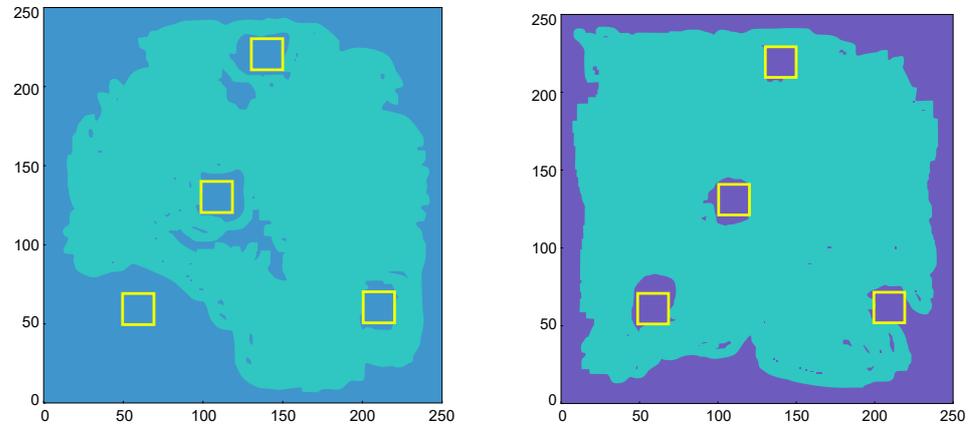


**Figure 3.** Display of the covered area with eight agents when the virtual leader performs Brownian motion (**left**) and Lévy flight (**right**) at the speed of 2 m/s after 10,000 s. There are some randomly distributed obstacles (yellow line) in the environment. Agents need to avoid obstacles automatically when covering the area.
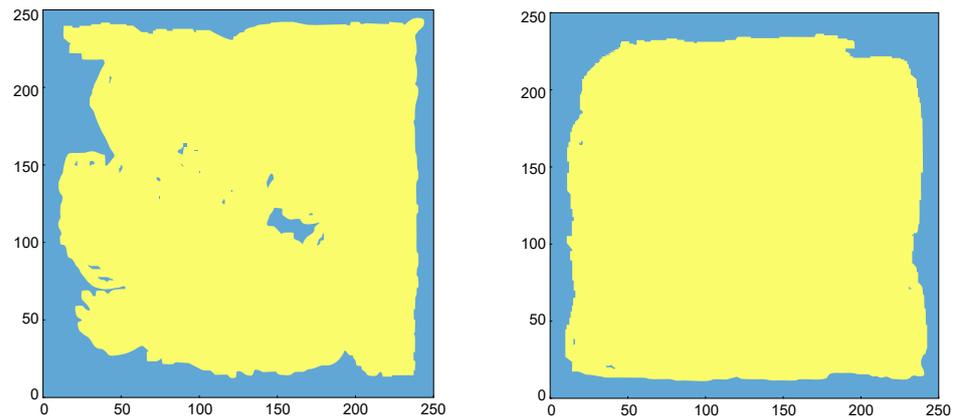


**Figure 4.** Display of the covered area with eight agents when the virtual leader performs Brownian motion (**left**) and Lévy flight (**right**) at the speed of 2 m/s after 10,000 s. There are no obstacles in the environment.



(**a**) Obstacle environment

(**b**) Obstacle-free environment
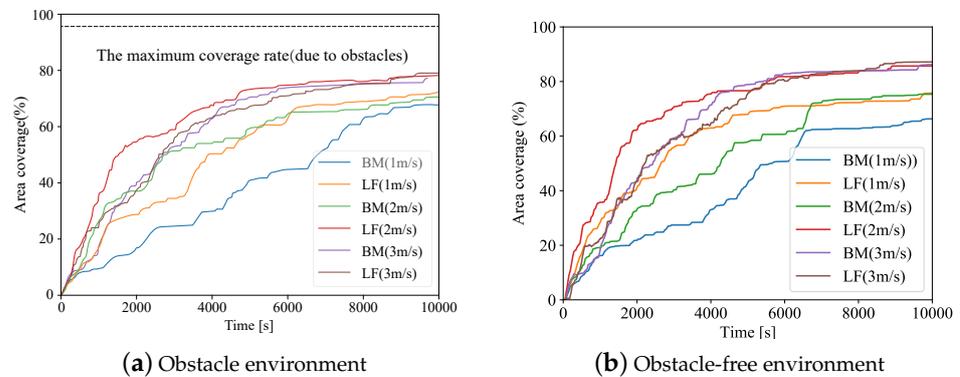
**Figure 5.** For the two different algorithms of Lévy flight and Brownian motion, the area coverage ratio of agents varies with the time.

(a)



(b)



(c)



(d)
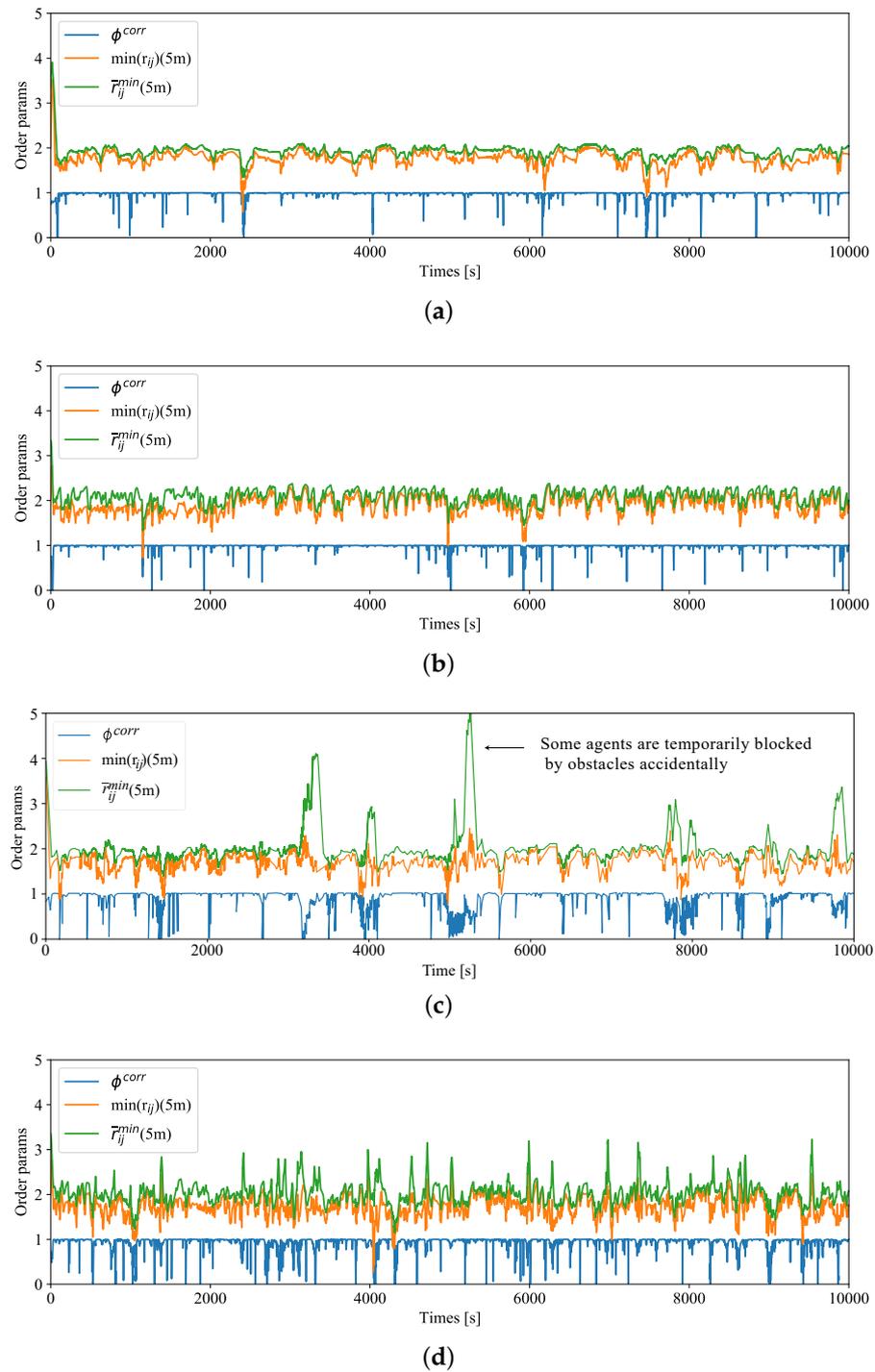
**Figure 6.** Order parameters as a function of time during experiments with eight agents in 10,000 s. $\phi^{corr}$ represents the correlation of agent's speed in swarm. $\overline{r_{ij}}^{min}$ is the average distance of the closest neighbours, whereas $min\,(r_{ij})$ is the minimum distance of the closest neighbours. The size of the arena is 250 m $\times$ 250 m. (**a**) Agents flocking to explore environment in an obstacle-free arena (1 m/s). (**b**) Agents flocking to explore environment in an obstacle-free arena (2 m/s). (**c**) Agents flocking to explore environment with obstacles in the arena (1 m/s). (**d**) Agents flocking to explore environment with obstacles in the arena (2 m/s).

## 6. Real-World Experiments

To evaluate the effectiveness of our method in real-world applications, we performed experiments with E-puck2 robots. We added an expansion board with Raspberry Pi to

the E-puck2 robots to increase their computing power. The E-puck2 robots communicate with each other via WiFi. In the arena with random obstacles and targets, eight E-puck2 robots searched for two targets with the LF-APF method. The E-puck2 robots obtained global information from the motion capture device above the arena, including the position information of the robots and obstacles of the arena. The E-puck2 robot and the initial scene of the task are shown in Figures 7 and 8, respectively.

We counted the time (Table 1) taken by the E-puck2 robot to search for the targets. From the table, we can see that the E-puck2 robots can always identify all targets and complete the task with the LF-APF method. We selected one representative experiment, as shown in Figure 9. When the E-puck2 robots encounter obstacles, they adjusted formations to bypass the obstacles without any collision. In addition, they can gather in the obstacle-free area and automatically change the formation when it is necessary to disperse. The E-puck2 robots kept a certain distance between each other during the entire flocking without colliding while forming a tight whole and moved orderly in the task area without running out of the boundary of the arena. From the above results and analysis, it can be suggested that the E-puck2 robots can adaptively deal with the environment to perform environmental exploration tasks in the arena using the LF-APF method.



**Figure 7.** E-puck2 robot platform. E-puck2 is the latest mini mobile robot developed by GCtronic and EPFL, which is an evolution of the successful E-puck robot used in many research and educational institutes.

**Table 1.** Statistics of the time taken by the agents to find the two targets, respectively, in 6 real-world experimental runs.

| Experiments | 1st | 2nd | 3rd | 4th | 5th | 6th |
|---|---|---|---|---|---|---|
| The time of finding one target | 174 s | 62 s | 235 s | 419 s | 311 s | 283 s |
| The time of finding all targets | 432 s | 211 s | 619 s | 847 s | 346 s | 438 s |
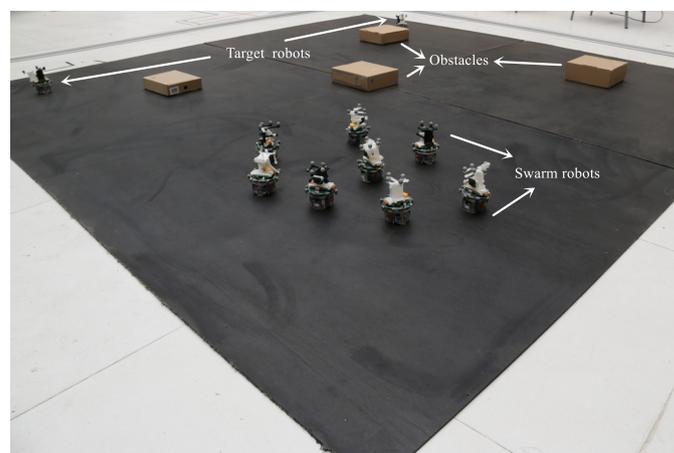


**Figure 8.** Initial scene of E-puck2 robots searching for the target in the arena.
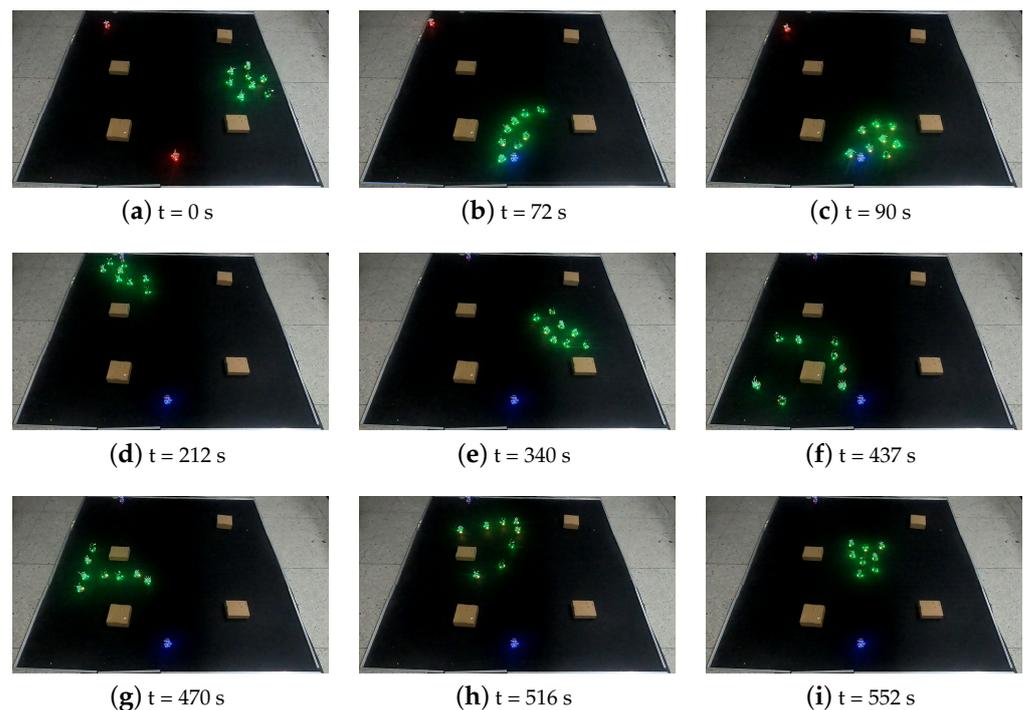
**Figure 9.** E-puck2 robots flocking to explore the environment. There are complex unknown obstacles and two dynamic targets in the arena (3 m × 3 m). The agents are green, and the targets are red. When the E-puck2 robots find the target (the distance between the agent and the target < 10 cm), the target becomes stationary and its colour turns blue.

## 7. Conclusions

In this study, we proposed the LF-APF method combining APF and LF mechanisms to achieve environmental exploration in swarms. The proposed method makes agents flock to explore unknown environments, relying on little environmental information. Agents in the swarm determine the position of the virtual leader, who performs LF, and form a flocking to follow the virtual leader to traverse the area searching for the target. In the process, the agents can adjust formations to adapt to the environment and incur no collisions between the agents or between the agents and obstacles. The resulting movements of the swarm robots are similar to those of the natural population, which suggests that the proposed method can be well applied to the exploration task of the bionic robot environment. Several simulations and real-world experiments have validated that the method can achieve effective and efficient environmental exploration.

**Author Contributions:** Conceptualization, Z.F. and C.W.; methodology, C.W.; software, C.W.; validation, Z.F., C.W., M.G. and H.H.; formal analysis, Z.F., M.G. and C.W.; investigation, Z.F.; resources, Z.F.; data curation, C.W.; writing—original draft preparation, C.W. and D.W.; writing—review and editing, Z.W. and Y.Y.; visualization, C.W. and H.H.; supervision, X.Z., W.W. and Z.F.; project administration, Z.F.; funding acquisition, Z.F. All authors have read and agreed to the published version of the manuscript.

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

**Abbreviations**

The following abbreviations are used in this manuscript:

LF      Lévy flight
BM      Brownian motion
APF     Artificial potential field

**Appendix A**

The video of experiments: Search for two targets with eight agents.
Simulation experiments: https://www.bilibili.com/video/BV1Sr4y1S7Xi?spm_id_from=333.999.0.0.
Real-world experiments: https://www.bilibili.com/video/BV1RT4y1h7vU?spm_id_from=333.999.0.0.

**References**

1. Chen, J.Y. UAV-guided navigation for ground robot tele-operation in a military reconnaissance environment. *Ergonomics* **2010**, *53*, 940–950. [CrossRef] [PubMed]
2. Waharte, S.; Trigoni, N.; Julier, S. Coordinated search with a swarm of UAVs. In Proceedings of the 2009 6th IEEE Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks Workshops, Rome, Italy, 22–26 July 2009; pp. 1–3.
3. Pitonakova, L.; Crowder, R.; Bullock, S. Information flow principles for plasticity in foraging robot swarms. *Swarm Intell.* **2016**, *10*, 33–63. [CrossRef]
4. Kei Cheang, U.; Lee, K.; Julius, A.A.; Kim, M.J. Multiple-robot drug delivery strategy through coordinated teams of microswimmers. *Appl. Phys. Lett.* **2014**, *105*, 083705. [CrossRef]
5. Wei, C.; Xu, J.; Wang, C.; Wiggers, P.; Hindriks, K. An approach to navigation for the humanoid robot nao in domestic environments. *Appl. Phys. Lett.* **2013**, 298–310.
6. McGuire, K.N.; De Wagter, 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*, eaaw9710. [CrossRef]
7. Herianto; Sakakibara, T.; Koiwa, T.; Kurabayashi, D. Realization of a pheromone potential field for autonomous navigation by radio frequency identification. *Adv. Robot.* **2008**, *22*, 1461–1478. [CrossRef]
8. Tang, Q.; Xu, Z.; Yu, F.; Zhang, Z.; Zhang, J. Dynamic target searching and tracking with swarm robots based on stigmergy mechanism. *Robot. Auton. Syst.* **2019**, *120*, 103251. [CrossRef]
9. Wei, C.; Hindriks, K.V.; Jonker, C.M. Dynamic task allocation for multi-robot search and retrieval tasks. *Appl. Intell.* **2016**, *45*, 383–401. [CrossRef]
10. Alitappeh, R.J.; Jeddisaravi, K. Multi-robot exploration in task allocation problem. *Appl. Intell.* **2022**, *52*, 2189–2211. [CrossRef]
11. Reid, C.R.; Sumpter, D.J.; Beekman, M. Optimisation in a natural system: Argentine ants solve the Towers of Hanoi. *Appl. Intell.* **2011**, *214*, 50–58. [CrossRef]
12. Menzel, R.; Fuchs, J.; Kirbach, A.; Lehmann, K.; Greggers, U. Navigation and Communication in Honey Bees. In *Honeybee Neurobiology and Behavior: A Tribute to Randolf Menzel*; Springer: Dordrecht, The Netherlands, 2012.
13. Wei, C.; Hindriks, K.V.; Jonker, C.M. Altruistic coordination for multi-robot cooperative pathfinding. *Appl. Intell.* **2016**, *44*, 269–281. [CrossRef]
14. Wu, M.; Zhu, X.; Ma, L.; Wang, J.; Bao, W.; Li, W.; Fan, Z. Torch: Strategy evolution in swarm robots using heterogeneous–homogeneous coevolution method. *J. Ind. Inf. Integr.* **2022**, *25*, 100239. [CrossRef]
15. Ajanic, E.; Feroskhan, M.; Mintchev, S.; Noca, F.; Floreano, D. Bioinspired wing and tail morphing extends drone flight capabilities. *Sci. Robot.* **2020**, *5*, eabc2897. [CrossRef] [PubMed]
16. Ramezani, A.; Chung, S.J.; Hutchinson, S. A biomimetic robotic platform to study flight specializations of bats. *Sci. Robot.* **2017**, *2*, eaal2505. [CrossRef] [PubMed]
17. Roderick, W.R.T.; Cutkosky, M.R.; Lentink, D. Bird-inspired dynamic grasping and perching in arboreal environments. *Sci. Robot.* **2021**, *6*, eabj7562. [CrossRef]
18. Viswanathan, G.M.; Afanasyev, V.; Buldyrev, S.V.; Havlin, S.; Da Luz, M.G.E.; Raposo, E.P.; Stanley, H.E. Lévy flights in random searches. *Phys. A Stat. Mech. Appl.* **2000**, *282*, 1–12. [CrossRef]
19. Wei, J.; Chen, Y.; Yu, Y.; Chen, Y. Optimal randomness in swarm-based search. *Mathematics* **2019**, *7*, 828. [CrossRef]
20. Martinez, F.; Jacinto, E.; Acero, D. Brownian motion as exploration strategy for autonomous swarm robots. *J. Anim. Ecol.* **2012**, 2375–2380.
21. Sims, D.W.; Humphries, N.E.; Bradford, R.W.; Bruce, B.D. Lévy flight and Brownian search patterns of a free-ranging predator reflect different prey field characteristics. *J. Anim. Ecol.* **2012**, *81*, 432–442. [CrossRef]
22. Sims, D.W.; Southall, E.J.; Humphries, N.E.; Hays, G.C.; Bradshaw, C.J.; Pitchford, J.W.; James, A.; Ahmed, M.Z.; Brierley, A.S.; Hindell, M.A.; et al. Scaling laws of marine predator search behaviour. *Nature* **2008**, *451*, 1098–1102. [CrossRef]

23. Reynolds, A.M.; Swain, J.L.; Smith, A.D.; Martin, A.P.; Osborne, J.L. Honeybees use a Lévy flight search strategy and odour-mediated anemotaxis to relocate food sources. *Behav. Ecol. Sociobiol.* **2009**, *64*, 115–123. [CrossRef]

24. 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]

25. Pang, B.; Song, Y.; Zhang, C.; Yang, R. Effect of random walk methods on searching efficiency in swarm robots for area exploration. *Appl. Intell.* **2021**, *51*, 5189–5199. [CrossRef]

26. Fioriti, V.; Fratichini, F.; Chiesa, S.; Moriconi, C. Levy foraging in a dynamic environment–extending the levy search. *Int. J. Adv. Robot. Syst.* **2015**, *12*, 98. [CrossRef]

27. Jeddisaravi, K.; Alitappeh, R.J.; Pimenta, L.C.A.; Guimarães, F.G. Multi-objective approach for robot motion planning in search tasks. *Appl. Intell.* **2016**, *45*, 305–321. [CrossRef]

28. Blum, C.; Merkle, D. Swarm Intelligence: Introduction and Applications. In *Applied Intelligence* ; Springer Science & Business Media: New York, NY, USA, 2008; pp. 19–20.

29. 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]

30. 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]

31. 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]

32. Vásárhelyi, G.; Virágh, C.; Somorjai, G.; Nepusz, T.; Eiben, A.E.; Vicsek, T. Optimized flocking of autonomous drones in confined environments. *Sci. Robot.* **2018**, *3*, eaat3536. [CrossRef]

33. Mantegna, R.N. Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes. *Phys. Rev. E* **2000**, *282*, 1–12. [CrossRef]

34. Pal, A.; Tiwari, R.; Shukla, A. Communication constraints multi-agent territory exploration task. *Appl. Intell.* **2013**, *38*, 357–383. [CrossRef]