# Faster Target Encirclement with Utilization of Obstacles via Multi-Agent Reinforcement Learning

**Yuxi Zheng** 516545651@qq.com
*Intelligent Game and Decision Lab, Beijing, China*
*Tianjin Artificial Intelligence Innovation Center, Tianjin, China*

**Yongjun Zhang**✉ yjzhang@nudt.edu.cn
*National Innovation Institute of Defense Technology, Beijing, China*

**Chenran Zhao** 260109852@qq.com and **Huanhuan Yang** yanghh94@126.com
*National University of Defense Technology, Changsha, China*

**Tongyue Li** li_tong_yue@163.com and **Qianying Ouyang** oyqy@nudt.edu.cn
*Intelligent Game and Decision Lab, Beijing, China*

**Ying Chen** selina.ychen@foxmail.com
*National Innovation Institute of Defense Technology, Beijing, China*

**Editors:** Berrin Yanıkoğlu and Wray Buntine

## Abstract

Multi-agent encirclement refers to controlling multiple agents to restrict the movement of a target and surround it with a specific formation. However, two challenges remain: encirclement in obstacle scenarios and encirclement of a faster target. In obstacle scenarios, we propose the utilization of obstacles for facilitating encirclement and introduce the concept of contributing angle to quantify the contribution of agents and obstacles, which enables agents to effectively utilize obstacles while mitigating the credit assignment problem. To address the challenge of encircling a faster target, we propose a two-stage encirclement method inspired by lions' hunting strategy, effectively preventing target escape. We design the reward function based on the contributing angle and the lion encirclement method, integrating it with the Multi-Agent Deep Deterministic Policy Gradient(MADDPG). The simulation results demonstrate that our method can utilize obstacles to complete encirclement and has a higher success rate. In some conditions with insufficient numbers of agents, our methods can still accomplish the task. Ablation experiments are conducted to verify the effectiveness of the contributing angle and the lion encirclement method respectively.

**Keywords:** Encirclement; Obstacle Utilization; Faster Target; Reinforcement Learning

## 1. Introduction

Multi-agent encirclement involves coordinating multiple agents to restrict the movement of a target, enclosing it with a specific formation. This technique finds applications in various domains such as reconnaissance, surveillance, target capture, and target protection using multi-robot systems.

Encircling a target in the presence of obstacles is an important and practical area of research (Sani et al. (2020)). Previous studies often treat obstacles as hindrances, which

© 2023 Y. Zheng, Y. Zhang✉, C. Zhao, H. Yang, T. Li, Q. Ouyang & Y. Chen.

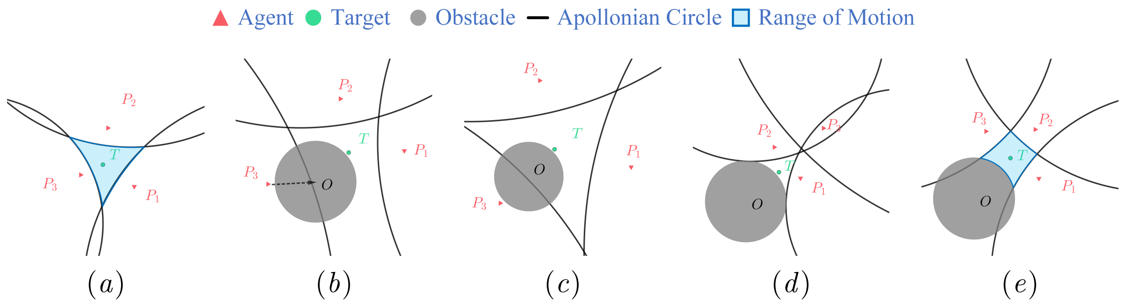


Figure 1: Several Scenarios of Encirclement.

can be attributed to several factors. Firstly, regular polygonal formations, which are effective in obstacle-free scenarios (as shown in Fig. 1a), may fail near obstacles (as shown in Fig. 1b). Secondly, the presence of obstacles may make it more difficult to determine the agents' contributions to the task (as shown in Figs. 1c and 1d), thereby exacerbating the credit assignment problem. Additionally, obstacles impede agents' actions, requiring collision avoidance strategies. However, in natural systems and human societies, various encirclement strategies utilize environmental cues, including obstacles, for efficient encirclement. Therefore, we argue that agents should possess the ability to strategically utilize obstacles for encirclement purposes, rather than merely considering them as obstacles. However, limited research exists on how agents can effectively utilize obstacles and organize formation near obstacles for encirclement.

In addition, despite extensive research and discussions on encirclement strategies for the faster target, the problem remains highly challenging. On one hand, the agility of the faster target allows it to quickly change its movement direction or escape from gaps between agents, resulting in suboptimal formations or incomplete encirclement. On the other hand, the faster target increases the complexity of the environment, placing greater demands on the cooperative and collision avoidance capabilities of the agents.

In this study, we aim to address the challenges associated with obstacle scenarios and the faster target in the encirclement task. To address these challenges, we incorporate obstacle utilization into the strategy, empowering agents to complete encirclement by utilizing obstacles. Additionally, we introduce the concept of the contributing angle to guide agents in effectively utilizing obstacles and address the credit assignment issues that may arise. About the problem of the faster target, we draw inspiration from lions (Stander (1992))and propose a two-stage encirclement method. We integrate the contributing angle and the lion encirclement method into the Multi-Agent Deep Deterministic Policy Gradient(MADDPG) (Lowe et al. (2017)) algorithm and evaluate the performance of our method in Multi-Agent Particle Environments(MPE) (Lowe et al. (2017)).

Simulation results demonstrate that our method can effectively utilize obstacles for encirclement. Compared to the baseline method, our approach achieves a higher success rate. Ablation experiments validate the effectiveness of the contributing angle and two-stage encirclement method. Furthermore, we validate our method in specialized scenarios, such as two-agent encirclement scenario and scenarios with concave-shaped and tunnel-shaped obstacles. Results demonstrate that our method enables agents to complete encirclement even with insufficient number of agents and adapt to special obstacle shapes.

Our contributions include: ***a*)** incorporating obstacle utilization into the encirclement strategy, enabling agents to effectively use obstacles for achieving encirclement; ***b*)** introducing the concept of the contributing angle to accurately quantify the contribution of agents and obstacles, thereby mitigating credit assignment problem; ***c*)** proposing a two-stage encirclement algorithm to improve the success rate when dealing with a faster target; and ***d*)** conducting experiments to validate the efficacy of our method.

## 2. Related Works

### 2.1. Encirclement in Scenarios with Obstacles

In scenarios with obstacles, most research considers obstacles as hindrances to the encirclement task. Consequently, they decompose the task into two subtasks: encirclement and collision avoidance (Ma et al. (2019); Zhang et al. (2020, 2022); Fan et al. (2022)). However, Oyler et al. (2016) argues that the presence of obstacles can have both favorable and unfavorable effects on both parties involved, depending on their relative positions. Zheng et al. (2021) propose that in discrete scenarios, if the target is near obstacles, agents can achieve encirclement by occupying alternative positions around the target. However, this study solely conducted experiments in discrete scenarios and did not explicitly incentivize agents to utilize obstacles for encirclement.

Hence, we contend that obstacles should not be regarded merely as hindrances to encirclement. Instead, agents should possess the capability to actively utilize obstacles for effective encirclement strategies.

### 2.2. Encirclement with a Faster Target

The challenge of the faster target presents difficulties for the effectiveness of the strategy. Previous research mainly focuses on pursuit-evasion games, with a limited exploration of the encirclement tasks. In unbounded scenarios, existing research primarily focuses on the requirements of agent number, velocity, and initial positions for achieving encirclement, with

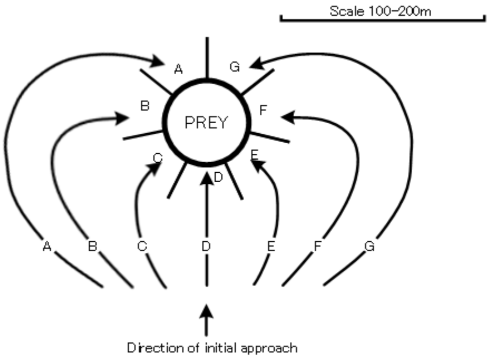


Figure 2: Seven typical encirclement roles in cooperative lion hunting (Stander (1992)).

subsequent discussions on encirclement strategies (Chen et al. (2016); Wang et al. (2013); Fang et al. (2020); Liang et al. (2023)). And Apollonian circle (Isaacs (1999)) is widely applied in task analysis. In bounded scenarios, the emphasis shifts towards attaining a higher success rate and fewer steps under random initialization conditions, such as de Souza et al. (2020). Despite the growing attention dedicated to the encirclement of the faster target, it remains an open problem (Kamimura and Ohira (2019)).

In this study, we investigate encirclement of the faster target in bounded scenarios. Drawing inspiration from lion encirclement strategies for a faster prey, we propose a bio-inspired approach to address this challenge.

### 2.3. The Encirclement Strategy of Lions

Stander (1992) conducted an analysis of lions' collaborative hunting behavior in 486 instances. It was found that lions typically use a strategy when hunting a faster prey as shown in Fig. 2. This strategy has two main features: (i) a priority of forming an enclosure. During the hunting process, lions on the left and right sides of the hunting group would circle around the target until they gain a favorable encirclement position, before closing in on the prey to launch an attack; and (ii) the division of roles among the members. The hunting formation of lions typically consists of seven roles (A to G). Each lion would play a designated role based on their relative position during the hunt.

## 3. Preliminary

### 3.1. Scenario Description and Task Modeling

In this study, we define the multi-agent encirclement task as follows: encircling a faster target(denoted as $T$) in a two-dimensional bounded environment with $m$ stationary obstacles (denoted as $O_j$, where $j \in \{1, 2, ..., m\}$). The goal is for $n$ agents (denoted as $P_i$, where $i \in \{1, 2, ..., n\}$) to limit target's movement and quickly encircle it.

We formulate the multi-agent encirclement task as a Markov Decision Process (MDP) consisting a five-tuple $\{S, A, R, P, \gamma\}$. At each time step $t$, the state received by agent $P_i$ can be represented as $s_t^i = [\text{pos}_i, v_i, s_{i,T}, s_{i,1}, s_{i,2}, \ldots, s_{i,n-1}, s_{i,O_1}, s_{i,O_2}, \ldots, s_{i,O_m}]$, where $\text{pos}_i$ and $v_i$ denote the absolute position and velocity of $P_i$, respectively. $s_{i,T}$, $s_{i,j}$ (for $j \in \{1, 2, \ldots, n-1\}$), and $s_{i,O_*}$ represent the relative positions of $T$, other agents, and obstacles, respectively. The action $a_t^i \in A$ denotes the chosen action by $P_i$ at current time step $t$. The reward function $R : S \times A \to R$ assigns a scalar reward to each state-action pair, while the state transition function $P : S \times A \to S$ defines the probabilistic dynamics of state transitions. The discount factor $\gamma \in [0, 1]$ weighs the importance of future rewards. The objective of the agents is to maximize the accumulated reward $G = \sum_{t=1}^{T} \gamma^t r_t$ through collaborative decision-making and action execution.

### 3.2. Apollonian Circle

All points that satisfy the ratio of their distance to two fixed points equal to $\lambda (\lambda \neq 1)$ form a circle, which is named the Apollonian circle. In the encirclement task, given a target $T(x_T, y_T)$ with a maximum speed $v_T$ and an agent $P_i(x_{P_i}, y_{P_i})$ with a maximum speed $v_P$, the Apollonian circle can describe positions where $P_i$ and $T$ can simultaneously arrive at their maximum speeds. The speed ratio $\lambda = \frac{v_T}{v_P}$. When $\lambda > 1$, indicating that the target is faster, the Apollonian circle is constructed around the agent $P_i$, as depicted in Fig. 3, and it is denoted as $o_i(A_i, r_i)$:

$$\begin{cases} A_i = \left( \frac{x_{P_i} - \lambda^2 x_T}{1-\lambda^2}, \frac{y_{P_i} - \lambda^2 y_T}{1-\lambda^2} \right) \\ \\ r_i = \frac{\lambda \sqrt{\left(x_{P_i} - x_T\right)^2 + \left(y_{P_i} - y_T\right)^2}}{1-\lambda^2} \end{cases} \tag{1}$$

Where $A_i$ represents the center of the Apollonian circle, and $r_i$ represents the radius. The region where $P_i$ can intercept $T$ is formed by the Apollonian circle $o_i$ (Bao-fu et al. (2012)). Specifically, $P_i$ can intercept $T$ only if the $T$'s escape path intersects with $o_i$.

### 3.3. MADDPG

The MADDPG algorithm employs the Actor-Critic framework. Each agent $P_i$ has an independent Actor network $\mu_i(\theta_i)$, while a centralized Critic network is shared among all agents. MADDPG adopts a framework that combines centralized training and distributed execution. During execution, each agent makes action decisions based on its local perception information $o_i$, following the policy $a_i = \mu_i(o_i^j)$. In the centralized training phase, the Critic network calculates individual value functions $Q_i$ for each agent using global information from all agents. To update the Actor network parameters $\theta_i$ for each agent $P_i$, policy update gradients are computed based on the individual value functions using a sampled batch of interaction data with a batch size of $b$:

$$\nabla_{\theta_i} J \approx \frac{1}{b} \sum_{j=1}^{b} \nabla_{\theta_i} \mu_i(o_i^j) \nabla_{a_i} Q_i^{\mu}(\mathbf{s}^j, a_1^j, ..., a_n^j) \mid_{a_i = \mu_i(o_i^j)} \tag{2}$$

Additionally, the Critic network is updated by minimizing the loss function:

$$L(\theta_i) = \frac{1}{b} \sum_{j=1}^{b} (y^j - Q_i^{\mu}(\mathbf{s}^j, a_1^j, ..., a_n^j))^2, y^j = r_i^j + \gamma Q_i^{\mu}(\mathbf{s}^j, a_1^j, ..., a_i, ..., a_n^j) \mid_{a_k = \mu_k(o_k^j)} \tag{3}$$

MADDPG effectively addresses non-stationarity, ensuring robust agent strategies. Its independent computation of individual values enables diverse reward structures, making it well-suited for collaborative tasks such as encirclement task.

## 4. Method

In this section, we introduce a new algorithm for encircling a faster target with utilization of obstacles. Our algorithm comprises two key components: the contributing angle and the two-stage lion encirclement strategy. The contributing angle allows us to measure the contribution of agents and obstacles in the encirclement process, effectively solving the obstacle utilization and the credit assignment problem. Using the contributing angle as foundation, we develop a two-stage strategy inspired by lions. This strategy involves forming the enclosure and then gradually compressing the target's range of movement. Furthermore, we design the reward function based on these components and integrate it with MADDPG.

### 4.1. Contributing Angle

We begin with the concept of occupied angle (Wang et al. (2013); Fang et al. (2020)), which is defined as follows:

**Definition 1** *(Occupied Angle) The occupied angle $\alpha_i$ of agent $P_i$ is defined as the angle between two tangent lines drawn from the position of $T$ to the Apollonian circle $o_i$ of $P_i$, as illustrated by the orange angle in Fig. 3. The expression for $\alpha_i$ is given by*

$$\alpha_i = 2 \arcsin \frac{1}{\lambda} \tag{4}$$

Where $\alpha_i$ is solely dependent on $\lambda$. When $\lambda$ is fixed, $\alpha_i$ remains constant as well. When $T$ moves towards the range of $\alpha_i$, agent $P_i$ is likely to intercept it; otherwise, it will not be intercepted. To encircle $T$, agents must have the ability to intercept $T$ from all directions, thus forming a closed enclosure around $T$ using the Apollonian circles, as shown in Fig. 1a. The completeness of the enclosure can be estimated by calculating the union of the occupied angles of all agents, denoted as the group occupied angle $\theta_G$.

$$\theta_G = \bigcup_{i=1}^{n} \alpha_i \quad \left(0 \leq \theta_G \leq \sum_{i=1}^{n} \alpha_i \leq 2\pi\right) \tag{5}$$

Without considering the obstacle utilization, $\theta_G = 2\pi$ becomes a necessary condition for encirclement. As a result, the number of agents $n$ must satisfy certain requirements for encirclement to be possible, which are solely related to $\lambda$:

$$n \geq \frac{\pi}{\arcsin \frac{1}{\lambda}} \tag{6}$$

We believe that obstacles can help, thus we incorporate obstacles into the construction of the closed enclosure (as shown in Fig. 1e). This alleviates the pressure on agents to

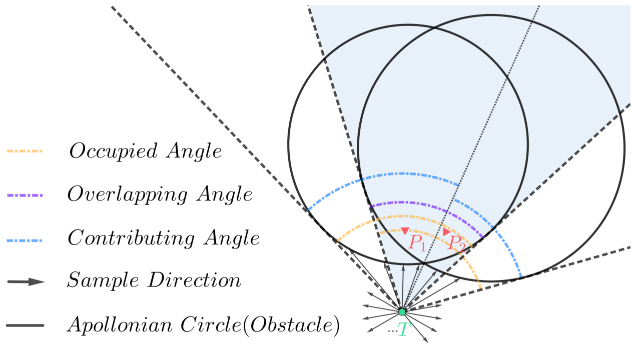


Figure 3: The Apollonian circle and relative angles in encirclement task.

achieve a complete enclosure, allowing for encirclement even with higher target velocity or insufficient number of agents. Additionally, the overall occupied angle, denoted as $\theta_A$, can be evaluated by considering the occupied angle of both agents and obstacles.

$$\theta_A = \left(\bigcup_{i=1}^{n} \alpha_i^P\right) \cup \left(\bigcup_{j=1}^{m} \alpha_j^O\right) \tag{7}$$

Unlike $\alpha_i^P$, the occupied angle of obstacle $\alpha_j^O$ varies based on its shape and distance to $T$. For convex obstacles, it ranges from 0 to $\pi$, and for concave obstacles, it ranges from 0 to $2\pi$. This variability highlights the potential of utilizing obstacles. In this study, we specifically consider circular convex obstacles and special shaped obstacles formed by them.

However, obstacle utilization presents challenges for credit assignment between agents and obstacles. On one hand, the lack of precise measurement for the positive contribution of obstacles hampers their effective utilization(Fig. 1c). On the other hand, the absence of a reliable metric for agents' contribution may lead to partial engagement(Fig. 1d). To ensure rational cooperation among agents and optimal obstacle utilization, accurately measuring the contribution of agents and obstacles is crucial. Inspired by this idea, we introduce the concept of encirclement contributor and the contributing angle.

**Definition 2** *(Encirclement Contributor) The encirclement contributor for a specific direction is defined as the owner of the Apollonian circle $o_i$ (or obstacle $O$) that the target $T$ first encounters when moving straight along that direction.*

In a given direction, the encirclement contributor would first intercept the target, making the primary contribution in that direction. This concept enables us to partition the contribution of overlapping angles between occupied angles, as depicted in Fig. 3. Subsequently, we can define the contributing angle for each entity.

**Definition 3** *(Contributing Angle) The contributing angle $\theta_c$ is defined as the angular range within which the individual agent's Apollonian circle (or obstacle) actually contributes to the encirclement task, as illustrated by the blue angle in Fig. 3.*

We employ a sampling-based approach to approximately estimate $\theta_c$. We sample the target's motion direction $k$ times from the interval $[0, 2\pi]$. We then categorize and aggregate the encirclement contributor for each of these sampled directions, allowing us to determine $\theta_c^E$ for each entity in the scenario.

$$\theta_c^E = \frac{2\pi}{k} \sum_{i=1}^{k} \delta_{c_i,E} \quad (E \in \{P_1, P_2, ..., P_n, O_1, O_2, ..., O_m\}) \tag{8}$$

where $c_i$ represents the encirclement contributor for the $i$-th sampled direction. The function $\delta_{c_i,E}$ serves as an indicator for the sampling process, taking the value of 1 when $c_i$ is $E$, and 0 otherwise.

$\theta_c$ serves as an effective measure to quantify the encirclement contributions of agents and obstacles in the task. By the $\theta_c$ associated with each agent and obstacle, we can calculate the overall occupied angle $\theta_A$ using the following expression:

$$\theta_A = \sum_{i=1}^{n} \theta_c^{P_i} + \sum_{j=1}^{m} \theta_c^{O_j} \tag{9}$$

By quantifying the encirclement contribution of agents and obstacles through contributing angle, we can effectively guide agents in strategically utilizing obstacles during the encirclement task. This quantification also encourages agents to actively participate and contribute to the encirclement, thereby addressing credit assignment problem.

### 4.2. The Two-Stage encirclement Method Inspired by Lions

In the preceding section, we covered the formation of a closed enclosure around $T$. However, Eq.(4) shows that $\alpha_i^P$ is distance-independent. Thus, even when a closed enclosure is formed, $T$ may still have a significant range of movement $R_m$, as illustrated by the shaded blue area in Figs. 1a and 1e. Hence, in addition to the enclosure, it is important to collectively reduce $d_i$ between $P_i$ and $T$. Consequently, the problem of encirclement can be approximated as an optimization problem:

$$\begin{cases} V-\max & f(s) = [\theta_A(s), \frac{1}{d_1(s)}, \frac{1}{d_2(s)}, ..., \frac{1}{d_n(s)}] \\ s.t. & s \in S \end{cases} \tag{10}$$

where $\theta_A(s)$ and $d_i(s)$ represent $\theta_A$ and $d_i$ under the current state $s$ respectively. The encirclement task can be decomposed into the optimization problem of maximizing $\theta_A(s)$ and minimizing $d_i(s)$.

To summarize, achieving encirclement requires meeting two crucial conditions: (1) forming a closed enclosure around the target using the Apollonian circles and obstacles, and (2) limiting the target's range of movement within a certain threshold, specifically $R_m < \delta_r$.

The hunting strategy of lions, as discussed in Section 2.3, proves effective in hunting faster prey by prioritizing the establishment of a surrounding advantage. This aligns with our research focus on the faster target. Furthermore, the orientation-first characteristic of the lions' strategy resonates with the theoretical principles of Apollonian circles.

Our research proposes a two-stage encirclement strategy inspired by the hunting characteristics of lions. This strategy is built upon the principles of the Apollonian circle theory and the contributing angle concept discussed in Section 4.1. It consists of two stages: the collaborative surround stage and the collaborative contraction stage. We employ $\theta_A$ as the stage division indicator and $\delta_e$ as the threshold for stage division.

**The Collaborative Surround Stage.** When $\theta_A < \delta_e$, agents should establish an encirclement advantage by circling around the target. This requires a balance between collective awareness to eventually form a closed enclosure, appropriating work allocation, and efficient use of obstacles. We utilize individual contributing angle $\theta_c^{P_i}$ and obstacle contributing angle $\theta_c^{O}$, as well as the overall occupied angle $\theta_A$, to guide agents towards advantageous positions. We constrained the experiment to only allow agents to use obstacles within the scene, rather than the scene boundary, to assess their ability in utilizing obstacles for encirclement.

**The Collaborative Contraction Stage.** When $\theta_A \geq \delta_e$, indicating a certain encirclement advantage has been obtained, the agents initiate the contraction of their formation. To achieve this, we utilize the distance $d_i$ between $P_i$ and $T$ to guide the actions. Notably, during this stage, the collaborative surround strategy is maintained to further enhance the advantage while compressing the target's range of movement $R_m$.

### 4.3. Reward Design

At each time step $t$, each agent $P_i$ receives an individual reward encompassing four distinct components: the collaborative surround reward $r_s$, the collaborative contraction reward $r_d$, the collision avoidance reward $r_a$, and the task completion reward $r_{\text{done}}$:

$$r_i = \begin{cases} r_s + r_a \quad (\theta_A < \delta_e) \\ r_s + r_d + r_a + r_{\text{done}} \quad (\theta_A \geq \delta_e) \end{cases} \tag{11}$$

**The Collaborative Surround Reward $r_s$.** For each agent $P_i$, $r_s^i$ can be defined as follows:

$$r_s^i = r_A + r_c^i + r_o \tag{12}$$

where $r_A$ positively correlated with $\theta_A$, promoting a collective incentive for accomplishing the closed enclosure. $r_c^i$ positively related to $\theta_c^{P_i}$, serving as individual motivation for active participation. Moreover, $r_o$ is an obstacle utilization reward, positively related to $\theta_c^O$, which stimulates the use of obstacles.

**The Collaborative Contraction Reward $r_d$.** For $P_i$, $r_d^i$ exhibits a negative correlation with the distance $d_i$ from $T$, which encourages agents to close the distance to the target:

$$r_d^i \propto \frac{1}{d_i} \tag{13}$$

**The Collision Avoidance Reward $r_a$.** As for $P_i$, $r_a^i$ is proportionate to the number of collisions occurring between $P_i$ and other entities at the current time-step.

**The Task Completion Reward $r_{done}$.** $r_{done}$ is a collective reward obtained by all agents upon completing the encirclement task.

## 5. Simulation Experiments

In this experiment, we use the following parameters: The scene size ranges from $x = -1.0$ to 1.0 and $y = -1.0$ to 1.0. The maximum agent velocity $v_P$ is fixed at 0.1 per step, while the maximum target velocity $v_T$ varies between 0.11 and 0.14. The number of agents, $n$, ranges from 2 to 5, and the number of obstacles, $m$, ranges from 0 to 3. In scenarios with special obstacles, multiple obstacles are combined into a single unit, with the configuration determining the number of obstacles. At the start of each round, the target, agents, and obstacles are randomly placed within the scene. A round is considered successful when a closed enclosing circle is formed and the target's range of movement, $R_m$, is less than 0.05.

We utilize a curriculum learning approach during the training phase, similar to de Souza et al. (2020). Initially, in the early stages of training, we set a larger safety distance $d^P_{safe}$ between $T$ and each $P_i$. As training progresses, $d^P_{safe}$ gradually decreases to its normal value. Each game session has a maximum duration of 60 steps, and the training process spans 200K episodes to ensure sufficient training. To mitigate the impact of training randomness, we use 5 different random seeds. During evaluation, we conduct 200 rounds for each model and collect a total of 1000 game results based on the 5 seeds for each configuration.

### 5.1. Target Behavior

We implemented a target behavior inspired by the repulsive mode described in de Souza et al. (2020). In this approach, each $P_i$ exerts a repulsive force on $T$ along the vector connecting them. Furthermore, obstacles and boundaries contribute forces to assist the target in avoiding collisions. The magnitude of these forces diminishes with the square of the distance between objects. The motion vector for the target is as follows:

$$\overrightarrow{v} = norm\left(\sum_{i=1}^{n}\left(\frac{\overrightarrow{P}_i - \overrightarrow{T}}{d_i^2}\right) + \sum_{j=1}^{m}\left(\frac{\overrightarrow{O}_j - \overrightarrow{T}}{d_j^2}\right) + \sum_{k=1}^{4}\left(\frac{\overrightarrow{W}_k - \overrightarrow{T}}{d_k^2}\right)\right) \times v_T \tag{14}$$

Where $\overrightarrow{T}$, $\overrightarrow{P}_i$, and $\overrightarrow{O}_j$ represent the positions of $T$, $P_i$, and $O_j$, respectively. $\overrightarrow{W}_k$ represents the coordinates of wall $k$. Additionally, $d_i$, $d_j$, and $d_k$ represent the distances to the target $T$. $v_T$ denotes the maximum speed of $T$.

Building upon the approach above, we incorporate action validity checking using the safety distance $d_{\text{safe}}$. To achieve this, we uniformly sample the motion path of $\overrightarrow{v}$. Those sampled points which fall within a distance of $d_{\text{safe}}$ from either the agent or the obstacle are deemed invalid. The action to be executed is determined by selecting the maximum sampled point along the motion path that contains no invalid points.

### 5.2. Experiment Models

We compare our approach with the work conducted by Ma et al. (2019). Ma also follows a centralized training and distributed execution mode, prioritizing guiding the agent toward the target and forming a regular n-gon formation. Ma utilizes encirclement radius, phase difference, and angular velocity as the reward. However, to ensure a fair comparison, we exclude the angular velocity reward from Ma since it aims to achieve circumnavigation. Moreover, we set $\delta_e$ to 0.93 and conduct ablation experiments to evaluate the impact of our method, including obstacle assistance, individual contributing angle, and the lion encirclement method. The ablation model used in our study is summarized in Table 1.

### 5.3. Experiment Design

We conducted experiments with different parameter values: $\lambda \in \{1.1, 1.2, 1.3, 1.4\}$, $n \in \{2, 3, 4, 5\}$, and $m \in \{0, 1, 2, 3\}$. The performance metrics used to evaluate the results were the success rate (S%) and the mean of successful episode length (MEL). It is worth noting that according to Eq.(6), the cases where $n = 2$ with a faster target and $n = 3$ with

Table 1: Experimental Model Options

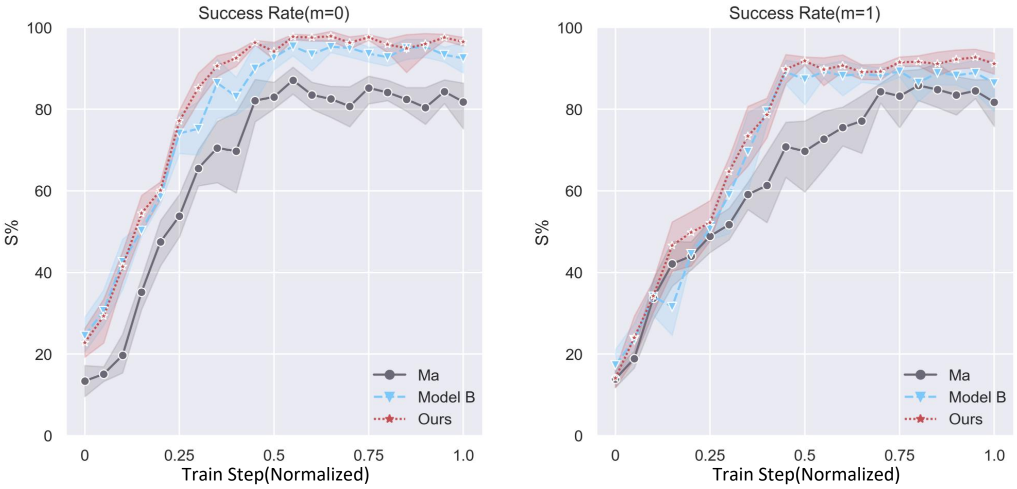
| Model Name | Options |  |  |
|---|---|---|---|
|  | Lion Encirclement | Obstacle-Assisted | Individual Contributing Angle |
| Model A | ✓ |  | ✓ |
| Model B |  | ✓ | ✓ |
| Model C | ✓ | ✓ |  |
| Ours | ✓ | ✓ | ✓ |

$\lambda = 1.2, 1.3, 1.4$ do not meet the minimum agent requirement for successful encirclement. In such scenarios, methods that solely rely on the agents are inadequate to achieve the desired target encirclement. Through these experiments, we validate that our approach successfully overcomes the limitations of Eq.(6).

In the comparison experiment between Model C and our method, the main focus was on evaluating the effect of individual contributing angles on credit assignment. This evaluation was quantified by calculating the standard deviation of $\theta_c^{P_i}$ for each agent in a round.

$$\sigma_c^P = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\theta_c^{P_i} - \overline{\theta_c^P})^2} \tag{15}$$

A larger value of $\sigma_c^P$ indicates a greater disparity in the contributions of agents to the task. Conversely, a smaller value suggests a more uniform performance among agents.



$(a)$ $(b)$

Figure 4: Success Rate in Scenarios with and without Obstacle.

Table 2: The performance under different $\lambda$

| Methods | m=1,n=3 | | | |
|---|---|---|---|---|
| | $\lambda = 1.1$ | $\lambda = 1.2$ | $\lambda = 1.3$ | $\lambda = 1.4$ |
| | S(%)/MEL | S(%)/MEL | S(%)/MEL | S(%)/MEL |
| Ma | 85.80/29.68 | – | – | – |
| Model A | 89.80/26.42 | – | – | – |
| Model B | 89.49/26.48 | 88.32/**27.42** | 84.20/29.10 | 70.60/**31.93** |
| Ours | **91.80**/**25.91** | **89.84**/27.82 | **88.70**/**28.52** | **73.70**/32.59 |

”–” denotes that the model cannot achieve encirclement in that setting.

Table 3: The performance under different $n$

| Methods | $\lambda = 1.1, m = 1$ | | | |
|---|---|---|---|---|
| | n=2 | n=3 | n=4 | n=5 |
| | S(%)/MEL | S(%)/MEL | S(%)/MEL | S(%)/MEL |
| Ma | – | 85.80/29.68 | 97.40/19.90 | 98.00/19.60 |
| Model A | – | 89.80/26.42 | 98.60/19.96 | 98.81/19.42 |
| Model B | 74.24/**28.43** | 89.49/26.48 | 98.25/**19.26** | 98.72/19.95 |
| Ours | **77.75**/29.48 | **91.80**/**25.91** | **98.77**/19.57 | **98.82**/**18.80** |

”–” denotes that the model cannot achieve encirclement in that setting.

Table 4: The performance under different $m$

| Methods | $n = 3, \lambda = 1.1$ | | | |
|---|---|---|---|---|
| | m=0 | m=1 | m=2 | m=3 |
| | S(%)/MEL | S(%)/MEL | S(%)/MEL | S(%)/MEL |
| Ma | 87.09/29.04 | 85.80/29.68 | 81.25/29.72 | 79.50/29.89 |
| Model A | 95.40/25.35 | 89.80/26.42 | 87.23/**26.61** | **85.49**/**29.59** |
| Model B | 95.40/25.35 | 89.49/26.48 | 86.37/28.20 | 82.23/29.97 |
| Ours | **97.88**/**23.51** | **91.80**/**25.91** | **87.25**/27.11 | 83.39/30.23 |

We further conducted experiments in scenes with special obstacle configurations, including concave obstacle scenes and tunnel scenes. These special obstacles were composed of multiple small circular obstacles, as illustrated in Figs. 6d and 6e.

## 6. Experiment Results

Fig. 4 illustrates the variations in success rate throughout the training process for scenarios with and without obstacles. Our method, which incorporates the lion encirclement strategy, demonstrates more stable training and achieves higher success rates compared to Model B, which does not utilize this strategy. Additionally, Model B also exhibits notable advantages over the baseline method.

### 6.1. Effect of Target Speed

Table 2 presents the performance of different models across various target speeds. It is evident that our method consistently achieves higher success rates when faced with targets of different speeds. Moreover, Model B demonstrates superior performance compared to the baseline method. Notably, both Ma and Model A fail to achieve encirclement when Eq. (6) is not satisfied. In contrast, our method and Model B, which utilize obstacles, remain effective even in these challenging scenarios.

### 6.2. Effect of Agent Number

Table 3 illustrates the performance of different models under varying agent numbers. It is commonly acknowledged that a larger number of agents leads to decreased task difficulty. Specifically, when $n = 2$, our method could also overcome this challenge with fewer agents. However, as the number of agents increases, the task difficulty diminishes, resulting in a gradual reduction of this advantage. When $n > 3$, simulation results demonstrate that the final enclosure tends not to utilize obstacles and the discrepancies in success rates among the models become minimal.

### 6.3. Effect of Obstacle Number

Table 4 showcases the performance of different models for varying obstacle numbers. The results demonstrate that our approach maintains an advantage in success rates when $m = 0, 1, 2$. However, as $m$ exceeds 2, there is a noticeable decline in model performance. This degradation can be attributed to several factors. Firstly, the algorithm lacks explicit communication among agents regarding which obstacle to utilize when multiple obstacles are present. This lack of consensus on obstacle utilization may impact the overall performance.

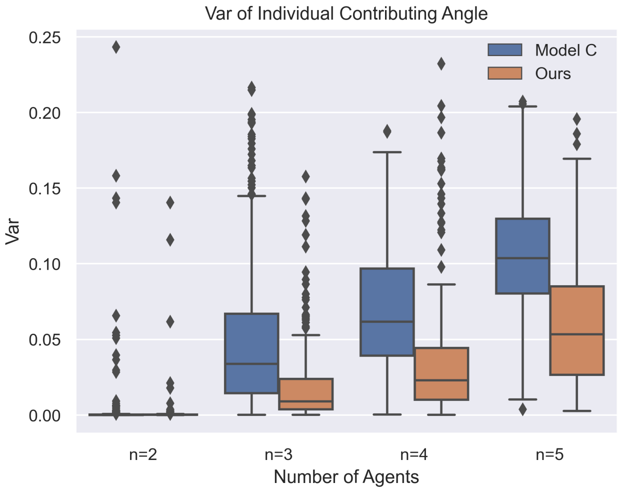


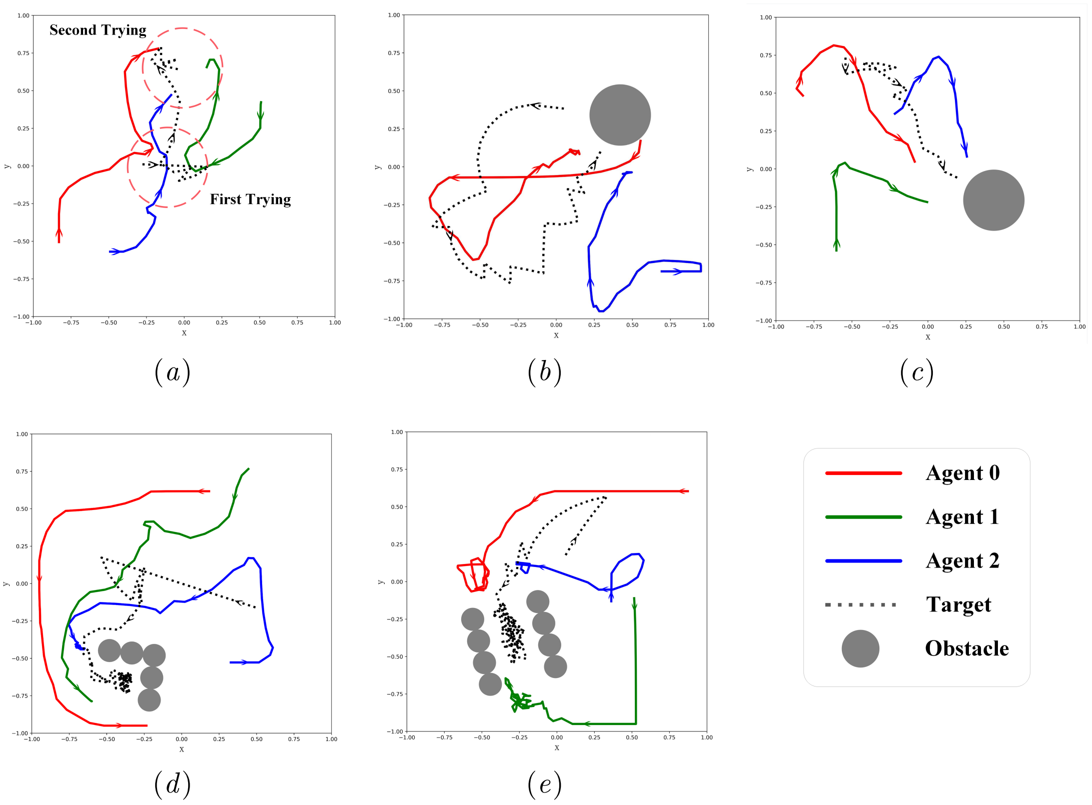
Figure 5: Effects of Individual Contributing Angle

Moreover, as the number of obstacles increases, the state space of the obstacle's relative positions grows exponentially, which subsequently affects obstacle utilization.

Furthermore, when considering MEL, the results indicate that our approach achieves greater efficiency in obstacle-free scenarios but may not always be optimal in scenarios with obstacles. This discrepancy can be attributed to several factors. Firstly, in complex scenarios with obstacles, the lion encirclement strategy may require more action steps for the formation and maintenance of the enclosure, sacrificing efficiency to improve success rates. Moreover, the biological strategy we learn from lions in open plains does not explicitly consider obstacles, which may not be well adapted to scenarios with obstacles.

## 6.4. Effect of Contributing Angle

Fig. 5 shows the impact of individual contributing angles. Our approach exhibits smaller $\sigma_c^P$ across different numbers of agents, indicating a more uniform performance among agents. Conversely, Model C, which does not utilize individual contributing angles, shows larger performance disparities. Furthermore, as $n$ increases, Model C experiences a more significant increase in $\sigma_c^P$. It can be attributed to the tasks becoming simpler as $n$ increases, leading to a more pronounced issue of lazy agents resulting from credit assignment.



$(a)$ $(b)$ $(c)$

$(d)$ $(e)$

Figure 6: Trajectory of the agents in various environmental settings.

### 6.5. Encirclement Trajectories

Fig. 6 illustrates the encircling trajectories of our method in various scenarios, showing the characteristics of lion hunting. In obstacle-free environments, the agents form regular polygon formations to encircle the target. When obstacles are present, the agents dynamically adjust their formations and utilize the obstacles to execute the encirclement successfully. Notably, our approach demonstrates adaptability to specific scenarios, including two-agents, concave, and tunnel obstacle environments.

Moreover, an interesting phenomenon is observed, as shown in Fig. 6a. Initially, during the first attempt at encirclement, the agents contract their formations, but the target manages to escape through the gaps between the agents. However, the agents then initiate a second attempt by adopting the lion strategy, where they split their formations and enclose the target from multiple directions, ultimately achieving a successful encirclement.

## 7. Conclusion

In this paper, we propose a distributed multi-agent encirclement deep reinforcement learning method for encircling a faster target in obstacle scenarios. We incorporate obstacle utilization into encirclement strategy and define the contributing angle, which can better quantify the specific contribution of agents and obstacles in the encirclement process. Inspired by the lion strategy in hunting faster prey, we propose a two-stage encirclement method. Simulation results demonstrate the effectiveness of our method in utilizing obstacles, improving success rates, and mitigating credit assignment.

In future work, we plan to extend our method from 2D to 3D to address more realistic scenarios. Additionally, we also plan to explore obstacle and target selection mechanisms to overcome the performance degradation of our method as the number of obstacles increases and extend our approach to multi-target scenarios.

## Acknowledgments

This work is supported by Science and Technology Innovation 2030 Major Project under Grant No. 2020AAA0104802 and National Natural Science Foundation of China under Grant No. 91948303 and 61906212.

## References

Fang Bao-fu, Pan Qi-shu, Hong Bing-Rong, Ding Lei, Zhong Qiu-bo, and Zhang Zhaosheng. Research on high speed evader vs. multi lower speed pursuers in multi pursuit-evasion games. *Information Technology Journal*, 11:989–997, 2012.

Jie Chen, Wenzhong Zha, Zhihong Peng, and Dongbing Gu. Multi-player pursuit-evasion games with one superior evader. *Autom.*, 71:24–32, 2016.

C. de Souza, Rhys Newbury, Akansel Cosgun, Pedro Castillo, Boris Vidolov, and Dana Kulić. Decentralized multi-agent pursuit using deep reinforcement learning. *IEEE Robotics and Automation Letters*, 6:4552–4559, 2020.

Zhilin Fan, Hong yong Yang, Fei Liu, Li Liu, and Yilin Han. Reinforcement learning method for target hunting control of multi-robot systems with obstacles. *International Journal of Intelligent Systems*, 37:11275 – 11298, 2022.

Xu Fang, Chen Wang, Lihua Xie, and Jie Chen. Cooperative pursuit with multi-pursuer and one faster free-moving evader. *IEEE Transactions on Cybernetics*, 52:1405–1414, 2020.

Rufus Isaacs. *Differential games: a mathematical theory with applications to warfare and pursuit, control and optimization.* Courier Corporation, 1999.

Atsushi Kamimura and Toru Ohira. Group chase and escape, fusion of pursuits-escapes and collective motions. 01 2019. doi: 10.1007/978-981-15-1731-0.

Xiao Liang, Boran Zhou, Linping Jiang, Guanglei Meng, and Yiwei Xiu. Collaborative pursuit-evasion game of multi-uavs based on apollonius circle in the environment with obstacle. *Connect. Sci.*, 35, 2023.

R Lowe, Y Wu, A Tamar, J Harb, P Abbeel, and I Mordatch. Multi-agent actor-critic for mixed cooperative-competitive environments. volume 30, 2017.

Junchong Ma, Huimin Lu, Junhao Xiao, Zhiwen Zeng, and Zhiqiang Zheng. Multi-robot target encirclement control with collision avoidance via deep reinforcement learning. *Journal of Intelligent & Robotic Systems*, 99:371 – 386, 2019.

Dave W. Oyler, Pierre T. Kabamba, and Anouck R. Girard. Pursuit-evasion games in the presence of obstacles. *Autom.*, 65:1–11, 2016.

Mukhtar Sani, Bogdan Robu, and Ahmad Hably. Pursuit-evasion game for nonholonomic mobile robots with obstacle avoidance using nmpc. *2020 28th Mediterranean Conference on Control and Automation (MED)*, pages 978–983, 2020.

P. E. Stander. Cooperative hunting in lions: the role of the individual. *Behavioral Ecology and Sociobiology*, 29:445–454, 1992.

Chen Wang, Ting Zhang, Kai Wang, Shuaiyi Lv, and Hongbin Ma. A new approach of multi-robot cooperative pursuit. *Proceedings of the 32nd Chinese Control Conference*, pages 7252–7256, 2013.

Tianle Zhang, Zhen Liu, Shiguang Wu, Z. Pu, and Jianqiang Yi. Multi-robot cooperative target encirclement through learning distributed transferable policy. *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2020.

Tianle Zhang, Zhen Liu, Z. Pu, and Jianqiang Yi. Multi-target encirclement with collision avoidance via deep reinforcement learning using relational graphs. *2022 International Conference on Robotics and Automation (ICRA)*, pages 8794–8800, 2022.

Yanbin Zheng, Wenxin Fan, and Mengyun Han. Research on multi-agent collaborative hunting algorithm based on game theory and q-learning for a single escaper. *J. Intell. Fuzzy Syst.*, 40:205–219, 2021.