Check for updates

Journal of Korea Robotics Society (2024) 19(1):117-129
https://doi.org/10.7746/jkros.2024.19.1.117 ISSN: 1975-6291 / eISSN: 2287-3961

커리큘럼 기반 심층 강화학습을 이용한 좁은 틈을 통과하는 무인기 군집 내비게이션

Collective Navigation Through a Narrow Gap for a Swarm of UAVs Using Curriculum-Based Deep Reinforcement Learning

최 명 열[1] · 신 우 재[1] · 김 민 우[1] · 박 휘 성[2] · 유 영 빈[3] · 이 민[3] · 오 현 동[†]
Myong-Yol Choi[1], Woojae Shin[1], Minwoo Kim[1], Hwi-Sung Park[2], Youngbin You[3], Min Lee[3], Hyondong Oh[†]

Abstract: This paper introduces collective navigation through a narrow gap using a curriculum-based deep reinforcement learning algorithm for a swarm of unmanned aerial vehicles (UAVs). Collective navigation in complex environments is essential for various applications such as search and rescue, environment monitoring and military tasks operations. Conventional methods, which are easily interpretable from an engineering perspective, divide the navigation tasks into mapping, planning, and control; however, they struggle with increased latency and unmodeled environmental factors. Recently, learning-based methods have addressed these problems by employing the end-to-end framework with neural networks. Nonetheless, most existing learning-based approaches face challenges in complex scenarios particularly for navigating through a narrow gap or when a leader or informed UAV is unavailable. Our approach uses the information of a certain number of nearest neighboring UAVs and incorporates a task-specific curriculum to reduce learning time and train a robust model. The effectiveness of the proposed algorithm is verified through an ablation study and quantitative metrics. Simulation results demonstrate that our approach outperforms existing methods.

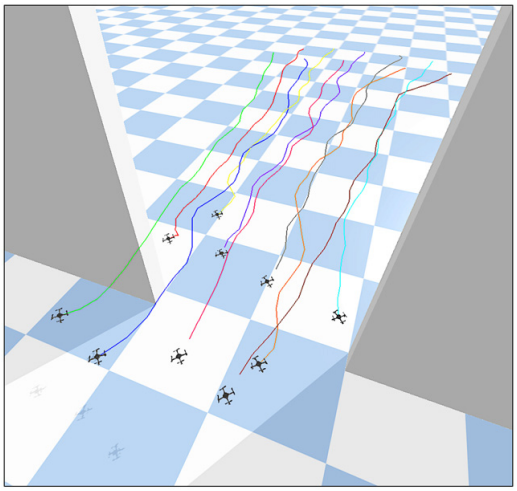
Keywords: Collective Navigation, Flocking, Collision Avoidance, Deep Reinforcement Learning, Curriculum Learning

1. 서 론

최근 들어, 무인 항공기(Unmanned Aerial Vehicles, UAVs)의 상용화가 급속도로 진행되고 있다. 이러한 UAV 시스템은 재난 구조부터 물류, 도심 환경 운송에 이르기까지 다양한 분야

Received : Oct. 31. 2023; Accepted : Nov. 22. 2023
※ This research was supported by Defense Acquisition Program Administration (DAPA) and Agency for Defense Development (ADD) (UG223047VD)
1. Master Student, Mechanical Engineering, UNIST, Ulsan, Korea (mychoi, oj7987, red9395@unist.ac.kr)
2. Researcher, ADD, Daejeon, Korea (7hwisung7@add.re.kr)
3. Researcher, LIG Nex1, Gyeonggi-do, Korea (youngbin.you, min.lee@lignex1.com)
† Associate Professor, Corresponding author: Mechanical Engineering, UNIST, Ulsan, Korea (h.oh@unist.ac.kr)

Copyright ⓒ KROS



[Fig. 1] A swarm of UAVs passing through a narrow gap

에서 적용 가능하다[1-5]. 특히 다중 UAV 시스템은 UAV 간에 효율적인 협력이 필요한 다양한 응용 분야에서 중요한 연구 주제이다.

협력적인 행동을 위한 주요 개념 중 하나는 flocking이다[6,7]. 이 개념은 자연 세계에서 새나 물고기와 같은 동물의 flocking behavior를 모방하여, 다수의 에이전트가 특정 규칙에 따라 움직이는 것을 의미한다. 이러한 flocking은 collective navigation의 핵심 요소로서, 에이전트들이 복잡하고 변동적인 환경에서도 효율적으로 목표지점까지 군집을 이루며 이동할 수 있게 도와준다[8].

Collective navigation은 자연에서 관찰되는 현상이며, 여러 동물의 계절적 이동이 대표적인 예이다[9-11]. 이 현상은 개체 간의 상호작용과 협력을 통해 효율적인 이동을 가능하게 하며, 이는 로보틱스, 특히 다중 UAV 시스템의 navigation 연구에 영감을 준다[12].

고전적인 방법은 navigation 작업을 mapping, planning, 그리고 control의 세 가지 구성 요소로 나누어 처리한다[13-15]. 하지만 이러한 접근법은 여러 한계를 가지고 있다. 첫째, 각 구성 요소 간의 상호 작용과 누적 오류로 인해 모델링 되지 않은 환경 요소에 민감하다. 둘째, 정보를 전달하거나 기다리는 추가적인 대기 시간이 필요하다. 따라서, 복잡한 환경에서 실시간으로 여러 에이전트가 flocking을 이루며 collective navigation을 수행하는 것을 어렵게 한다.

최근에는 이 문제를 해결하기 위해 end-to-end 딥러닝 기반의 접근 방법이 제안되고 있다[16-18]. 그러나 이러한 방법들 역시 한계를 가진다. 대부분의 학습 기반 알고리듬은 장애물이 없거나 특정한 환경에 최적화된 reward 함수를 가지고 있으며, 복잡하거나 매우 좁은 공간과 같은 도전적인 환경에 적용하기 어렵다[19-21]. 또한, 대부분은 리더 역할을 하는 UAV가 경로를 제공해야 하며, 이러한 리더 UAV가 없을 경우 collective navigation을 성공적으로 수행하기 어렵다[22].

본 논문에서는 이러한 문제점을 해결하기 위해 curriculum[23] 기반의 심층 강화학습 알고리듬을 제안한다. 이 알고리듬은 [Fig. 1]과 같이 다수의 UAV가 flocking을 이루면서 최대한 빠르게 좁은 틈을 지나 목표지점에 도달하는 collective navigation을 가능하게 한다. 이를 위해 첫번째로, 사전에 주어진 경로 없이 UAV가 좁은 틈을 통과할 수 있도록 특화된 observation, action, reward를 사용하여 환경을 구성한다. 두번째로, 관찰/정보 교환이 가능한 이웃 UAV 수에 따른 성능을 비교하여 이웃 정보를 효율적으로 활용하는 방법을 탐구한다. 세번째로, task-specific curriculum을 도입함으로써 학습 속도를 향상시키고 강건한 모델을 학습하여 우수한 일반화 능력을 보여준다. 마지막으로, 이러한 요소들의 중요성은 ablation study를 통해 확인되며, 정량적 평가지표와 시뮬레이션[24] 결과를 통해 본 논문에서 제안한 기법이 기존 방법보다 우수함을 입증한다.

본 논문의 구성은 다음과 같다. 2장에서는 문제 정의에 대해서 설명한다. 3장에서는 다양한 시뮬레이션 환경에서의 실험 결과를 통해 제안된 알고리듬이 어떻게 기존 방법보다 우수한 성능을 보이는지 분석한다. 마지막으로, 4장에서는 본 논문의 주요 결론과 향후 연구 방향에 대해서 기술한다.

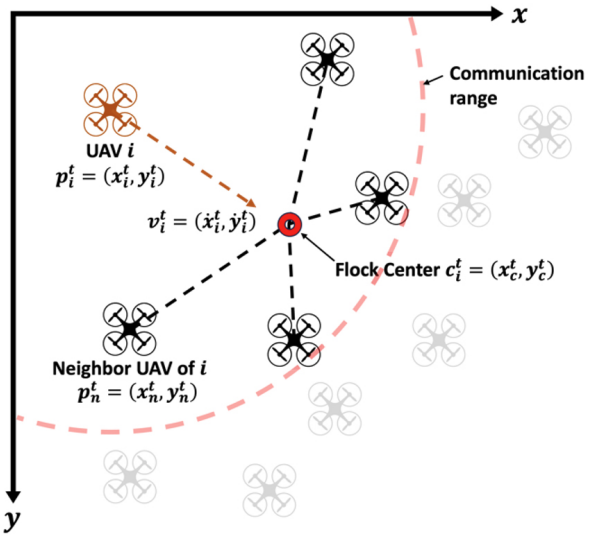
2. 문제 정의

본 섹션에서는 [21]을 참고하여 collective navigation의 목적을 설명하고, 이를 순차적 의사결정 문제로 정의한다. 특히 환경을 구성하는 observation, action, 그리고 reward에 대해 자세히 설명한다.

2.1 Collective Navigation의 목적

Collective navigation은 다음 세 개의 하위 목표로 정의될 수 있다. 첫번째로, 각 UAV는 flocking을 이루기 위해 flock의 중심에 가까이 있어야 한다. 두번째로, 각 UAV는 이웃 UAV 및 장애물과의 충돌을 피해야 한다. 마지막으로, 각 UAV는 목표지점에 가능한 한 빠르게 도달해야 한다.

[Fig. 2]에서 볼 수 있듯이, flock은 UAV i가 관찰할 수 있는 이웃 UAV들을 지칭하고, flock의 중심은 해당 이웃 UAV들의 평균 위치를 나타낸다. 관찰할 수 있는 이웃 UAV에 대해서는 Section 2.3 Observation에서 자세하게 설명하도록 한다.



[Fig. 2] At time t, UAV i's position and velocity are denoted by $p_i^t = (x_i^t, y_i^t)$ and $v_i^t = (\dot{x}_i^t, \dot{y}_i^t)$, respectively, while the position of a neighbor UAV is given as $p_n^t = (x_n^t, y_n^t)$. The center of the flock represents the average position of the neighbor UAVs

시간 t일 때 UAV i의 위치와 속도는 각각 $p_i^t=(x_i^t,y_i^t)$와 $v_i^t=(\dot{x}_i^t,\dot{y}_i^t)$이며, i의 이웃 UAV의 위치는 $p_n^t=(x_n^t,y_n^t)$이다. UAV의 총 수는 N이며, 목표지점에 도달하기까지 걸리는 비행 시간은 T이다. UAV i에 인접한 flock의 중심 $c_i^t=(x_c^t,y_c^t)$은 식 (1)과 같이 정의된다.

$$\begin{cases} x_c^t=\dfrac{1}{|\mathbb{SU}_i^t|}\sum_{i\in\mathbb{SU}^t}x_i^t \\ y_c^t=\dfrac{1}{|\mathbb{SU}_i^t|}\sum_{i\in\mathbb{SU}^t}y_i^t \end{cases} \tag{1}$$

여기서, $\mathbb{SU}_i^t$는 i의 관찰 가능한 이웃 UAV들 중에서 시간 t까지 충돌이 일어나지 않은 UAV들의 집합이다. 충돌을 피하기 위해서는 UAV 간의 거리가 최소 안전 거리 d_s 보다 커야 한다. 시간 t까지 충돌이 일어나지 않은 모든 UAV들의 집합 $\mathbb{SU}^t$는 식 (2)와 같이 정의된다.

$$\mathbb{SU}^t=\left\{i\middle|\sqrt{(x_i^{t'}-x_n^{t'})^2+(y_i^{t'}-y_n^{t'})^2}>d_s\right\} \tag{2}$$

여기서, $i\in N, n\in N_i$이고 N_i는 $N=\{1,2,\cdots,N\}$에서 i를 제외한 i의 관찰 가능한 이웃 UAV들의 집합이며 $0\le t'\le t$이다. 모든 UAV는 목표지점 중심 $g=(x_g,y_g)$에 대한 정보를 가지고 있다고 가정한다. 목표지점의 반경은 d_t로 정의된다. $\mathbb{TU}$는 목표지점에 도달한 모든 UAV들의 집합 의미하며, 이는 식 (3)과 같이 정의된다.

$$\mathbb{TU}=\left\{i\middle|\sqrt{(x_i^t-x_g)^2+(y_i^t-y_g)^2}<d_t\right\} \tag{3}$$

여기서, $i\in\mathbb{SU}^T$이고 $0\le t\le T$이다.

Collective navigation의 목적은 각 UAV i에 대해 다음과 같은 제약 최적화 문제를 해결하는 control policy를 찾는 것이다.

$$\begin{aligned} \min\textstyle\sum_{t=0}^{T}\sqrt{(x_i^t-x_c^t)^2+(y_i^t-y_c^t)^2} \\ s.t.\ i\in\mathbb{SU}^T \\ i\in\mathbb{TU} \end{aligned} \tag{4}$$

식 (4)에서, 목적 함수는 UAV i와 flock의 중심 사이의 거리를 최소화해야 한다는 것을 의미한다. 이는 flocking behavior를 정량화 하기 위해 일반적으로 사용되는 지표이다[25]. 두 가지 제약 조건은 UAV가 다른 UAV와 충돌하지 않고 목표지점에 도달해야 한다는 것을 의미한다.

정확한 모델이 주어졌을 경우, 전통적인 최적화 알고리듬을 활용해 문제를 해결할 수 있지만, 강화 학습은 이러한 모델에 의존하지 않고 더 높은 수준의 유연성을 제공한다.

2.2 순차적 의사결정 문제 정의

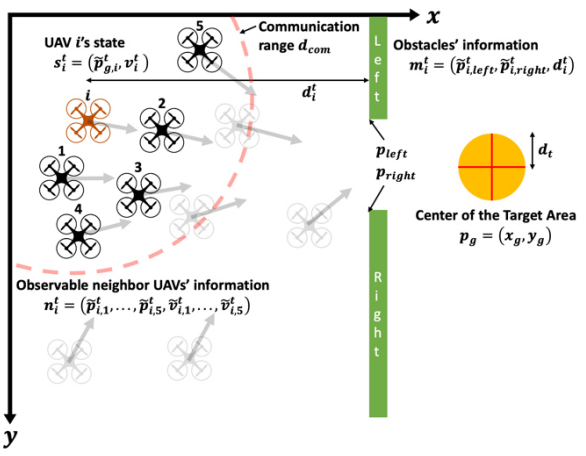
순차적 의사결정 문제는 주로 Markov decision processes (MDPs)로 표현된다. UAV들의 collective navigation 문제는 이러한 순차적 의사결정의 한 예다. 여기서 우리는 장기적인 목표 달성을 위해 각 UAV를 단계적으로 제어한다. MDPs는 이러한 장기적인 목표를 cumulative reward의 형태로 나타내고, 제어 신호를 transition probability로 표현한다. MDPs는 식 (5)와 같이 다섯개의 요소로 설명할 수 있다.

$$(S,A,P,R,\gamma) \tag{5}$$

여기서 S와 A는 state와 action의 집합을 각각 나타낸다. $P: S\times A\times S\to R$은 환경의 transition probability를 의미한다. 또한 $R: S\times A\times S\to R$은 reward 함수를 나타내고 γ는 discount factor이다. State, action, reward에 대한 자세한 설명은 다음 섹션에서 이어진다.

2.3 Observation

실제 환경에서의 의사결정 문제는 대개 복잡하고 다양한 정보에 기반한다. MDPs에서의 state는 환경의 완전한 정보를 반영하는 것이 이상적이지만, 실제 환경에서는 모든 정보를 완벽하게 알 수 없다. 따라서 본 연구에서는 UAV의 observation을 MDPs의 state로 간주함으로써, 이러한 실제 환경의 제약 조건을 반영하고자 한다.



[Fig. 3] The observation o_i^t of UAV i at time t comprises its own state s_i^t, information of observable neighbor UAVs n_i^t, and obstacle information m_i^t

시간 t일 때 UAV i의 observation o_i^t는 식 (6)과 같이 정의된다.

$$o_i^t = (s_i^t, n_i^t, m_i^t) \tag{6}$$

[Fig. 3]에서 볼 수 있듯이, $s_i^t = (\tilde{p}_{g,i}^t, v_i^t)$은 UAV i의 state를 나타내고, $n_i^t = (\tilde{p}_{i,1}^t, ..., \tilde{p}_{i,5}^t, \tilde{v}_{i,1}^t, ..., \tilde{v}_{i,5}^t)$은 UAV i의 관찰 가능한 이웃 UAV의 정보를 나타내며, $m_i^t = (\tilde{p}_{i,left}^t, \tilde{p}_{i,right}^t, d_i^t)$은 장애물의 정보를 나타낸다. p_{left}와 p_{right}는 좁은 틈을 형성하는 장애물의 모서리 위치를 나타내며, d_i^t는 UAV i와 장애물 사이의 거리를 나타낸다. $p_g = (x_g, y_g)$는 목표지점의 중심 위치를 나타낸다. $\tilde{p}_{i,j}^t = p_j^t - p_i^t$와 $\tilde{v}_{i,j}^t = v_j^t - v_i^t$는 시간 t일 때 i에 상대적인 j의 위치와 속도를 나타낸다. $p \in \mathbb{R}^2$와 $v \in \mathbb{R}^2$는 world frame에서의 위치와 속도를 나타낸다.

목표지점과 장애물의 위치는 사전에 지정된다. UAV i의 state 정보는 해당 UAV의 온보드 센서를 통해 얻으며, 이웃 UAV들의 정보는 통신을 통해 실시간으로 얻을 수 있다고 가정한다. 또한, 효율성을 위해 통신 범위 내의 모든 이웃 UAV의 정보를 활용하는 대신, UAV i와의 거리 기준으로 이웃을 정렬하고 가장 근접한 몇몇의 이웃 정보만을 선택하여 활용한다. 이는 성능 변화가 크지 않은 선에서 신경망의 입력 차원을 축소시키기 위한 조치이다. 만약 충분한 수의 이웃 UAV가 존재하지 않을 경우, 해당 observation 값을 기본 값인 0으로 채운다. 또한, 모든 observation 값들을 $[-1,1]$의 범위로 정규화 하였으며, 전체 observation의 차원은 39로 설정하였다. 관찰 가능한 이웃 UAV를 결정하는 기준에 대해서는 Section 3.3.1 Ablation Study에서 상세하게 다루도록 한다.

2.4 Action

UAV의 제어 신호는 연속적인 특성을 가지며, 이는 collective navigation 환경에서의 action space도 연속적이다는 것을 의미한다. [Fig. 2]에서 볼 수 있듯이, 시간 t일 때 UAV i가 수행하는 action a_i^t는 식 (7)과 같고, 각각 x 방향과 y방향 속도를 제어하는 실수 값을 의미한다.

$$a_i^t = v_i^t = (\dot{x}_i^t, \dot{y}_i^t) \tag{7}$$

2.5 Reward

본 연구 환경에서의 UAV들은 세 가지 주요 목표를 가지고 있다. 첫번째로, flocking behavior를 유지하여 flock의 중심에 가까이 있어야 한다. 두번째로, 이웃 무인기를 포함한 장애물을 피하며 안전한 거리를 유지해야 한다. 마지막으로, 목표지점까지 최대한 빠르게 이동해야 한다. 이러한 목표들을 반영하기 위해 적절한 reward 함수가 설계되었다.

모든 UAV들은 본 섹션에서 설계된 reward 함수를 공유한다. 해당 reward 함수는 식 (8)과 같이 flocking behavior에 대한 reward, 장애물 회피에 대한 reward, 그리고 목표지점까지의 navigation에 대한 reward, 총 세 가지로 구성되어 있다.

$$\begin{aligned} R = &(w_{coh}R_{coh} + w_{sep}R_{sep} + w_{ali}R_{ali}) \\ &+ (w_{gap_{ali}}R_{gap_{ali}} + w_{gap_{sep}}R_{gap_{sep}} + w_{wall}R_{wall}) + w_{nav}R_{nav} \end{aligned} \tag{8}$$

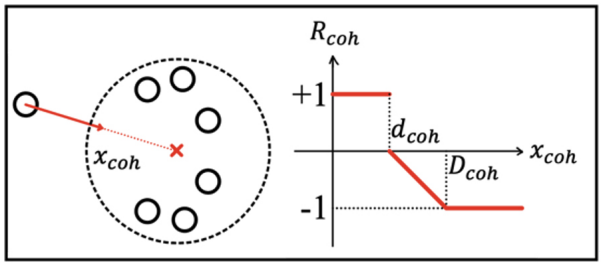
2.5.1 Flocking Reward

Flocking behavior를 달성하기 위해, UAV가 flock의 중심을 향하도록 (cohesion), 가까운 이웃과 충돌하지 않도록 (separation), 그리고 주변 UAV들과 속도 방향을 일치하도록 (alignment) reward 함수를 설계하였다[26].

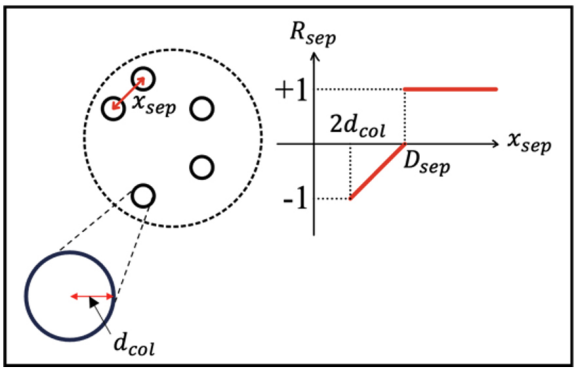
먼저, [Fig. 4]와 같이 UAV가 flock의 중심에 가까이 가도록 하기 위한 cohesion reward를 설계하였다. UAV의 중심과 flock의 중심 사이의 거리 x_{coh}가 desired cohesion distance d_{coh}보다 가까우면 $+1$의 reward가 주어진다. 그러나 d_{coh}를 초과하면, penalty가 주어지고 과도하게 penalty가 커지는 것을 방지하기 위해 maximum cohesion distance D_{coh}을 초과하는 경우 -1로 제한을 둔다. 그 외의 경우에는 -1과 0 사이의 penalty가 주어진다.

Separation에 대해서는, [Fig. 5]와 같이 UAV끼리 충돌하지 않도록 하기 위한 reward를 설계하였다. UAV의 중심과 가장 가까운 다른 UAV의 중심 사이의 거리 x_{sep}이 충돌 반경 d_{col}의 두 배보다 가까운 경우 -1의 penalty가 주어진다. 만약 desired separation distance D_{sep}를 초과하면 $+1$의 reward가 주어진다. 그 외에는 -1과 0사이의 penalty가 주어진다.

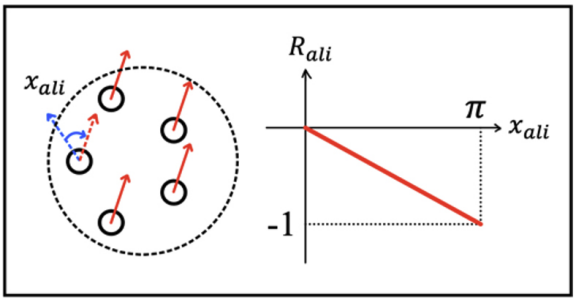
마지막으로 alignment에 대해서는, [Fig. 6]과 같이 주변 UAV들의 속도 방향과 일치시키기 위해 UAV의 속도 방향과 주변 UAV의 평균 속도 방향 사이의 차이 x_{ali}만큼 penalty가 주어진다. x_{ali}는 최대 π만큼 차이가 나기 때문에 이 경우 -1



[Fig. 4] Reward for cohesion



[Fig. 5] Reward for separation



[Fig. 6] Reward for alignment

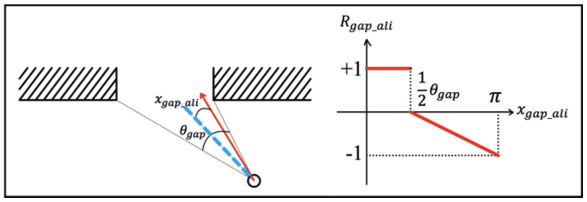
의 penalty가 주어지도록 하였다. 이러한 flocking의 세 가지 하위 목표들, 즉 cohesion, separation, 그리고 alignment는 UAV들이 군집 내에서 효율적으로 움직이며, 동시에 안전성을 보장하고 목표지점에 성공적으로 도달할 수 있게 하는 역할을 한다.

2.5.2 장애물 회피 Reward

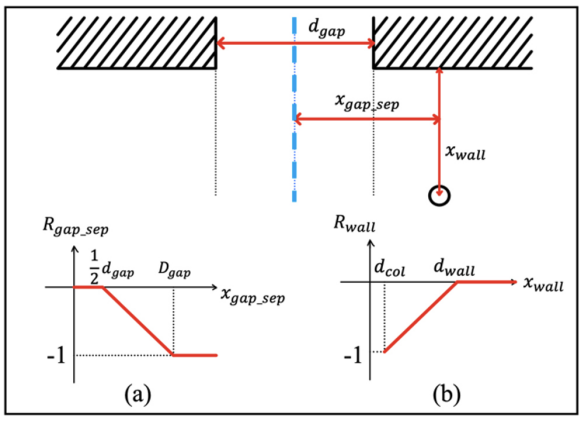
장애물 회피를 위해 먼저 [Fig. 7]과 같이 UAV가 좁은 틈을 통과하기 위한 reward를 설계하였다. UAV의 속도 방향과 사전에 정의된 θ_{gap}을 이등분하는 파란 점선 사이의 각도 x_{gap_ali}가 $\frac{1}{2}\theta_{gap}$보다 작은 경우 $+1$의 reward가 주어진다. 그렇지 않을 경우, UAV의 속도 방향이 틈에서 멀어짐에 따라 최대 -1의 penalty가 주어진다. 이를 통해, UAV는 장애물을 피해 좁은 틈으로 이동할 수 있게 된다.

또한, UAV가 벽으로부터 멀어질 수 있도록 하기 위한 reward를 설계하였다. [Fig. 8(a)]와 같이 UAV와 장애물 사이 틈을 이등분하는 파란 점선 사이의 거리 x_{gap_sep}가 $\frac{1}{2}\theta_{gap}$보다 큰 경우, UAV가 틈으로부터 멀어진다는 것을 알 수 있고, 이 때 UAV가 벽 앞에 위치하게 되며, 그 결과로 penalty가 점진적으로 주어진다. 만약 maximum distance from gap D_{gap}보다 커지게 되면 최대 -1의 penalty가 주어진다.

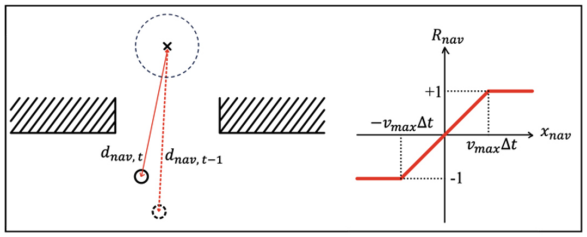
[Fig. 8(b)]와 같이 UAV와 벽 사이의 거리 x_{wall}가 safe



[Fig. 7] Reward for moving towards the narrow gap



[Fig. 8] Rewards for moving away from the walls. (a) shows the reward according to the distance from the center of the gap. (b) shows the reward based on distance from the walls



[Fig. 9] Reward for navigation

distance to wall d_{wall}보다 작은 경우, 벽과 충돌할 가능성이 커지므로 penalty가 점진적으로 주어진다. 그리고 UAV의 충돌 반경과 같아지는 순간 최대 -1의 penalty가 주어진다. 이러한 장애물 회피 reward 함수 설계는 UAV가 장애물과의 충돌을 피하고 좁은 틈을 통해 목표지점으로 이동할 수 있도록 한다.

2.5.3 Navigation Reward

목표지점에 도달하기 위해 navigation reward R_{nav}는 $d_{nav,t-1} - d_{nav,t}$에 비례한다. [Fig. 9]에서 볼 수 있듯이 $d_{nav,t-1}$는 action을 취하기 전의 UAV 중심과 목표지점 중심 사이의 거리 x_{nav}이며, $d_{nav,t}$는 action 이후의 해당 x_{nav}이다. 그 결과, UAV가 목표지점에 접근할 때마다 reward가 주어지고, 목표지점으로부터 멀어질 때마다 penalty가 주어진다. 한 time step 동안 움직일 수 있는 최대 거리를 초과하는 경우, 최대 크기 1의

reward 또는 penalty가 주어진다. 이러한 방식으로, UAV는 목표지점에 효과적이고 신속하게 이동하도록 한다. 여기서 $v_{\max}$는 UAV의 최대 이동 속도이고, Δt는 time step의 크기이므로 $v_{\max}\Delta t$는 한 time step 동안 이동할 수 있는 최대 거리를 나타낸다.

2.5.4 Total Reward

매 time step마다 action을 취한 후 UAV에게 주어지는 total reward는 설명한 reward들의 가중 합이고, 식 (8)과 같이 나타낼 수 있다. 이를 통해, 각 하위 목표에 따른 reward들이 적절하게 조합되어 최종적으로 UAV들이 flocking을 이루며 좁은 틈을 지나 목표지점에 도달하는 collective navigation이 가능하게 된다. 각 reward term에 대한 가중치는 [Table 1]과 같이 중요성에 따라 조절할 수 있으며, 이러한 가중치 조절을 통해 성능을 최적화할 수 있다. 또한, UAV가 목표 state에 빠르게 도달하고 에피소드의 종료 조건을 명확하게 알 수 있도록 termination reward가 에피소드 종료 시 추가적으로 주어진다. Termination reward는 [Table 2]와 같다. 목표지점 도달 시 10.0의 reward가 주어지고, 충돌이 발생하거나 UAV가 flock으로부터 멀리 떨어질 경우 -5.0의 penalty가 주어진다. Reward 설계 시 사용된 parameters는 [Table 3]과 같다.

[Table 1] Weights of reward

Weight	Value
w_{coh}	0.1
w_{sep}	0.2
w_{ali}	0.1
w_{gap_ali}	0.2
w_{gap_sep}	0.2
w_{wall}	0.1
w_{nav}	0.4

[Table 2] Termination reward

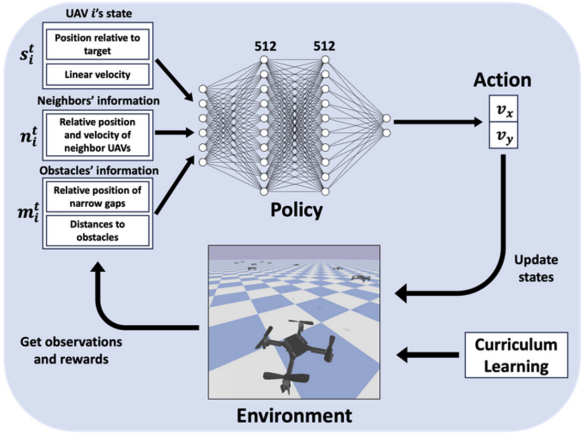
Termination condition	Value
Reaching the target	10.0
Collision	-5.0
Moving away from the flock	-5.0

[Table 3] Parameters of reward

Parameter	Value
d_{coh}	1.0 *m*
D_{coh}	2.0 *m*
d_{col}	0.06 *m*
D_{sep}	0.6 *m*
d_{gap}	2.0 *m*
D_{gap}	4.0 *m*
d_{wall}	1.0 *m*
$v_{\max}$	1.0 *m/s*
Δt	1/48 *Hz*

2.6 Policy Training

본 논문에서는 Gym-pybullet-drones[27] 시뮬레이터와 RLlib[28]를 활용해 [Fig. 10]과 같이 Proximal Policy Optimization[29] (PPO) 기반 policy를 학습하였다. PPO는 on-policy 기반 강화학습 알고리듬으로 일반적인 환경에서 안정적인 성능을 보여준다. Policy 신경망과 value 신경망은 각각 두 층의 fully connected layer로 표현된다. 첫번째와 두번째 layer는 각각 512개의 뉴런을 가지고 있다. 각 hidden layer의 활성화 함수로는 hyperbolic tangent 함수를 사용하였다. 신경망의 학습에 필요한 하이퍼파라미터들은 [Table 4]에 기술되어 있으며, 이러한 하이퍼파라미터들은 RLlib의 grid search를 통해 결정되었다.



[Fig. 10] The framework for training deep neural network

[Table 4] Hyperparameters of PPO

Hyperparameter	Value
Learning rate	3×10^{-4}
Optimizer	Adam
GAE lambda	0.95
Initial coefficient for KL Divergence	0.2
Target value for KL Divergence	0.01
Coefficient of the entropy regularizer	0.01
Size of each SGD epoch	4096
Minibatch size within each epoch	256
Size of batches collected from each worker	256

3. 시뮬레이션

본 섹션에서는 시뮬레이션의 task 및 setup에 대해서 설명하며, 그에 따른 결과와 해석을 다룬다.

3.1 Task Overview

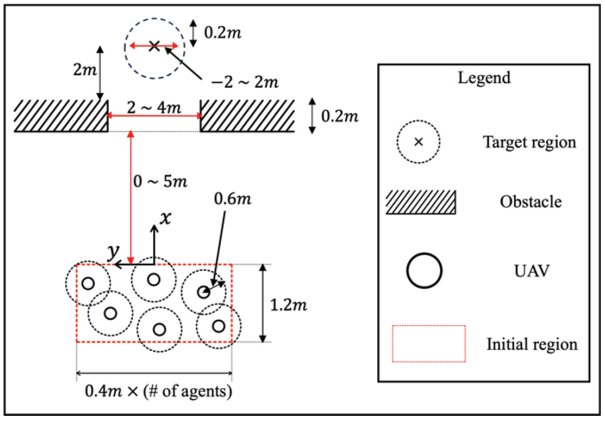
시뮬레이션 상에서 여러 대의 UAV가 flocking behavior (cohesion, separation and alignment)를 유지하며 좁은 틈을 통과하여 목표지점에 도달하는 collective navigation task를 수행한다. 매 에피소드마다 각 UAV들은 사전에 정의된 초기 영역 내에서 랜덤하게 생성되고, 장애물과 목표지점 또한 랜덤하게 생성된다. 이때 UAV들끼리 겹치는 상황을 피하기 위해, UAV 간에 적어도 $0.6m$의 거리를 두도록 하였다. 또한, 목표지점은 좁은 틈을 형성하는 장애물의 $2m$ 뒤에 위치하도록 하였다.

[Fig. 11]은 대략적인 학습 환경을 나타낸다. 각 에피소드마다 UAV의 시작 위치, 장애물의 위치, 목표지점 위치가 바뀌기 때문에 다양한 환경에서 강건하게 동작할 수 있는 policy 학습이 가능하다. 또한, 학습 성능 향상을 위해 curriculum learning을 적용했다. Curriculum learning에 대한 자세한 설명은 Section 3.2.2 Task-Specific Curriculum에서 하도록 한다.

3.2 Simulation Setup

문제를 단순화하기 위해, 동일한 비행 고도를 사용한다. 이 설정에서는 UAV들이 고정된 고도에서만 비행하므로, 수직 축에서의 충돌 가능성이 제거된다. 그럼에도 불구하고, 모든 UAV들이 동일한 고도에서 비행할 때 충돌이 심각한 문제가 될 수 있기 때문에, 고정 고도 비행 모드는 제안된 collective navigation 알고리듬을 검증하는 데 적절한 선택이다.

본 논문에서는 UAV 간 정보 교환이 한정된 통신 범위 내에서만 이루어진다고 가정한다. 각 UAV는 위치와 속도를 주변 UAV에게 전송하고, 받은 정보를 바탕으로 action을 수행한다. 최대 통신 범위 d_{com}를 $5m$로 설정하였으며, 이보다 더 먼 UAV의 정보는 얻을 수 없다. 또한, 각 UAV는 환경 내에서 충돌을 피해야 한다. [Fig. 5]에서 볼 수 있듯이 충돌 반경 d_{col}을 정의한다. d_{col} 이하의 거리에 다른 UAV 또는 장애물이 있는 경우 충돌로 간주된다. 마지막으로, 목표지점 영역은 $x-y$ 평면 상에서 반지름 d_t를 가지는 원형으로 정의된다. [Fig. 11]에서 볼 수 있듯이 목표지점 영역의 반지름 d_t는 $0.2m$로 설정하였다. 목표지점 영역에 UAV가 도달하는 순간 task를 성공한 것으로 간주한다.



[Fig. 11] Overview of training environment

3.2.1 Kinematic model of UAV

각 UAV는 식 (9)와 같은 kinematic model을 따른다.

$$\begin{cases} \dot{p} = v \\ \dot{v} = \dfrac{v_d - v}{\Delta t} \end{cases} \tag{9}$$

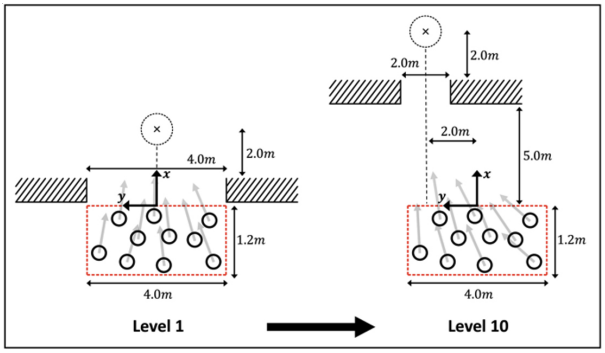
$$v_d^t = 0.85v^{t-1} + 0.15v_d^t \tag{10}$$

여기서 $p=(x,y)$는 UAV의 위치를 나타내고, $v=(\dot{x},\dot{y})$는 속도를 나타낸다. v_d는 신경망에서 생성되는 UAV의 속도 제어 명령이다. UAV의 부드러운 움직임을 위해 식 (10)과 같이 이전 time step의 속도와 현재 time step의 제어 명령을 각각 85%와 15%의 비율로 섞어서 최종 속도 제어 명령을 생성한다.

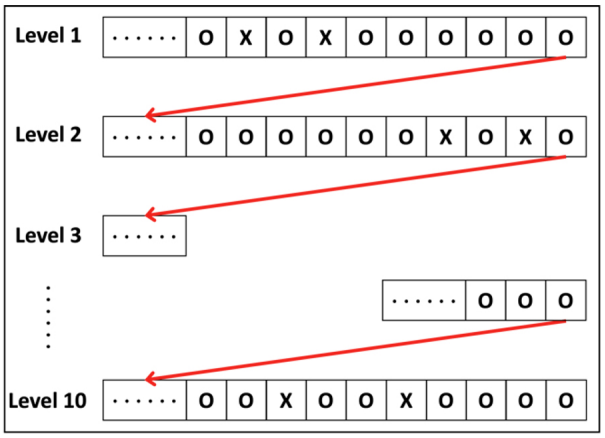
3.2.2 Task-Specific Curriculum

본 논문에서는 보다 빠른 학습과 성능 향상을 위해 task-specific curriculum을 사용한다. 이 curriculum은 [Fig. 12]와 같이 환경의 난이도를 점진적으로 높여가며 학습을 진행한다. 난이도를 조절하는 요소는 1) 장애물의 $+x$ 방향 위치, 2) 장애물로 인해 생기는 틈이 x축 기준 y 방향으로 랜덤하게 위치할 수 있는 거리, 3) 틈 사이의 거리로 총 3가지가 있다. 환경의 난이도는 총 10단계로 나누어져 있으며 1단계는 처음 시작 단계로서 가장 쉬우며 점차 단계가 높아질수록 환경의 난이도가 상승한다.

[Fig. 12]에서 볼 수 있듯이, 1단계에서 장애물로 인해 생기는 틈의 $+x$ 방향 위치는 $0m$이고, x축 기준 y방향으로 틈이 랜덤하게 위치할 수 있는 거리는 $0m$이며, 틈 사이의 거리는 $4m$이다. 마지막 단계인 10단계에서 장애물로 인해 생기는 틈의 $+x$ 방향 위치는 $5m$이고 x축 기준 y방향으로 틈이 랜덤하게 위치할 수 있는 거리는 최대 $2m$이며, 틈 사이의 거리는 최소 $2m$이다.



[Fig. 12] Difficulty of task with level



[Fig. 13] Criteria for increasing the difficulty of the environment

환경 난이도를 높이는 기준은 [Fig. 13]과 같다. 최근 10번의 에피소드 중 적어도 8번 이상 UAV가 목표지점 도달에 성공하면 환경의 난이도를 높인다. 이러한 접근 방식은 모델을 점진적으로 복잡한 환경에 적응시키며, 학습 수렴 속도와 일반화 능력을 향상시킨다.

3.2.3 평가 지표

본 논문에서는 알고리듬의 성능을 평가하기 위해 학습 시 사용하지 않은 환경에서 총 100번 검증했다. 평가지표는 목표지점 도달 성공률 success rate (SR), 평균 비행 시간 average episodic length (AEL), flock의 중심과 UAV들 사이의 평균 거리 average distance between flock center and UAVs ($ADFCU$)로 총 세 가지이다.

첫번째로, 목표지점 도달 성공률 SR은 식 (11)과 같이 전체 시도 횟수 중에서 모든 UAV가 목표지점에 도달한 비율을 나타낸다.

$$SR = \frac{SE}{M} \times 100\% \tag{11}$$

여기서 M은 전체 시도 횟수, SE는 전체 시도 횟수 중에서 $N = |\mathbb{SU}^T| = |\mathbb{TU}|$인 에피소드 수를 의미한다. $|\mathbb{SU}^T|$와 $|\mathbb{TU}|$는 각각 마지막까지 충돌이 일어나지 않은 UAV 수와 목표지점에 도달한 UAV 수를 의미한다.

두번째로, 평균 비행 시간 AEL은 식 (12)와 같이 모든 UAV가 목표지점에 도달하는 경우에 한하여, 초기 영역에서 목표지점 영역까지 이동하는데 걸린 시간을 의미한다.

$$AEL = \frac{1}{SE}\sum_{k=1}^{M} T_k \tag{12}$$

$$T = \begin{cases} T, \text{if } N = |\mathbb{SU}^T| = |\mathbb{TU}| \\ 0, \ otherwise. \end{cases} \tag{13}$$

여기서 T는 식 (13)과 같이 모든 UAV가 목표지점에 도달한 경우의 에피소드 길이를 의미하고, 그렇지 않을 경우에는 0의 값을 가진다.

마지막으로 flock의 중심과 UAV들 사이의 평균 거리 $ADFCU$는 식 (14)과 같이 모든 UAV가 목표지점에 도달하는 경우에 대한 해당 거리들의 평균이다.

$$ADFCU = \frac{1}{SE \cdot N}\sum_{t=0}^{T}\sum_{i \in \mathbb{RU}} \sqrt{(x_i^t - x_c^t)^2 + (y_i^t - y_c^t)^2} \tag{14}$$

여기서 $\mathbb{RU}$는 $N = |\mathbb{SU}^T| = |\mathbb{TU}|$를 만족하는 UAV들의 집합이다. 이러한 지표들은 알고리듬의 성능을 종합적으로 평가할 수 있도록 한다.

3.3 결과

본 섹션에서는 제안된 알고리듬의 유효성과 중요성을 ablation study를 통해 보이며, 정량적 지표를 사용한 성능 평가와 시뮬레이션 결과를 통해 검증한다.

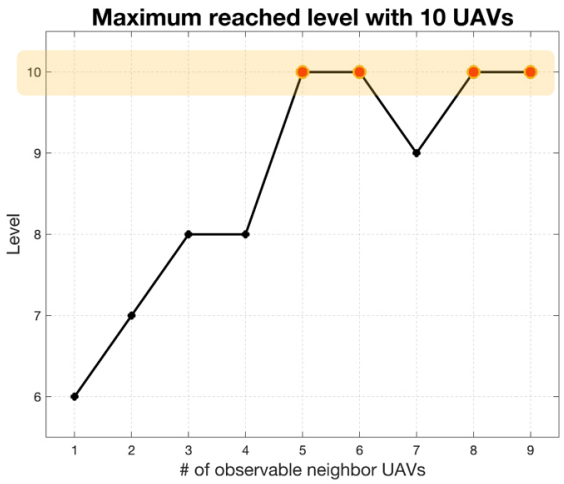
3.3.1 Ablation Study

먼저, 관찰 가능한 적절한 이웃 UAV 수를 결정하는 기준에 대해서 다루도록 한다. 통신 범위 내에 있는 모든 이웃 UAV들의 정보를 활용하는 방법은 다양한 주변 정보를 얻는 이점이 있지만, 가까운 UAV 수가 많아질 수록 불필요한 정보가 담기는 경우가 있을 수 있다. 또한, 관찰 가능한 이웃 UAV 수를 임의로 설정하는 경우, 그 수에 따른 성능 차이가 있을 수 있다. 따라서 본 논문에서는 통신 범위 내에 있는 관찰 가능한 이웃 UAV 수에 따른 성능을 비교하여 이웃 정보를 효율적으로 활용하고자 한다.

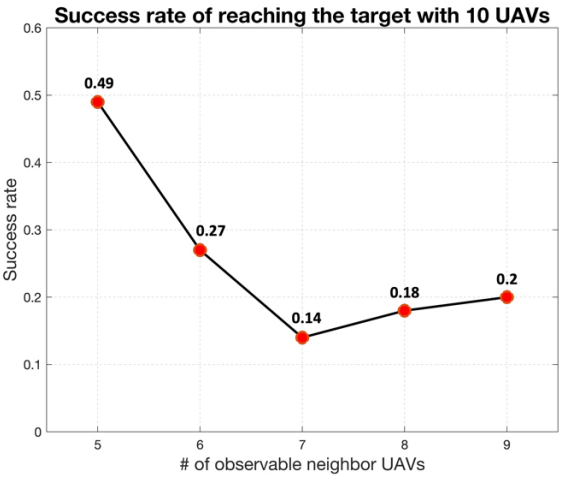
관찰 가능한 이웃 UAV는 거리에 따라 정하도록 한다. 예를 들어 관찰 가능한 UAV의 수가 5라고 정해지면 통신 범위 내에 있는 가장 가까운 이웃 UAV 5대의 정보만을 활용하는 것이다.

환경의 최종 난이도에 도달하는 관찰 가능한 이웃 UAV 수를 확인하기 위해 UAV 10대로 task-specific curriculum을 이용한 학습을 진행하였다. 전체 UAV 수가 10대이므로 이웃 UAV의 정보를 1대부터 9대까지 이용하여 각각의 성능을 비교하였다. [Fig. 14]에서 볼 수 있듯이 환경의 최종 난이도에 도달한 관찰 가능한 이웃 UAV 수는 5, 6, 8, 그리고 9가 있다. 이를 통해, 제안된 알고리듬은 관찰 가능한 이웃 UAV 수가 5 이상인 경우 시뮬레이션 환경에서 대부분 최종 난이도에 도달할 수 있다는 것을 확인할 수 있다.

관찰 가능한 이웃 UAV 수가 5인 경우부터 9인 경우까지 성능을 비교하기 위해 각각 최종 난이도의 환경에서 목표지점



[Fig. 14] Maximum reached level in curriculum learning with a different number of observable neighbor UAVs



[Fig. 15] Success rates of reaching target for UAVs with the number of observable neighbor UAVs ranging from 5 to 9

도달 성공률을 비교하였다. [Fig. 15]에서 볼 수 있듯이 관찰 가능한 이웃 UAV 수가 5인 경우의 성능이 가장 높다. 이는 통신 범위 내에 있는 모든 이웃 UAV들을 활용하는 것보다 가장 가까운 5대의 이웃의 정보를 활용하는 것이 더 효과적이라는 것을 나타낸다.

이러한 현상이 발생하는 이유는 더 적은 수의 이웃 정보를 활용하게 되면 정보에 노이즈가 덜 포함될 수 있지만, 정보의 범위가 짧아져서 대규모 행동 패턴을 이해하거나 예측하는 데에 한계가 있을 수 있기 때문이다. 반면에 더 많은 수의 이웃 정보를 활용하게 되면, 정보의 범위는 넓어지겠지만 노이즈가 늘어날 가능성이 있다. 또한, 더 많은 이웃의 정보를 처리해야 하므로, 정보의 복잡성이 증가하고 학습이나 의사결정에 있어 수렴이 더 어려워질 수 있다. 그러므로 적절한 수의 이웃 정보를 활용하는 것이 중요하며, 이는 [30]에서도 2차원 상에서는 3-5개, 3차원 상에서는 6-7개의 이웃을 고려하는 것이 flocking behavior에 최적임을 보여준다. 따라서 본 논문에서는 통신 범위 내에 있는 가장 가까운 5대의 이웃 UAV 정보를 활용하도록 한다.

이제 curriculum의 영향을 정량적 평가 지표를 통해 확인하도록 한다. 관찰 가능한 이웃 UAV 수를 5로 설정하였으므로, 총 6대로 학습시킨 후, 여기서 학습된 신경망을 사용하여 추가적인 학습 없이 전체 UAV 수 8, 10, 12, 15대까지 테스트를 진행하여 결과를 확인하였다.

Curriculum을 사용하지 않고 학습을 진행한 경우, [Table 5]에서 볼 수 있듯이, 전반적으로 낮은 목표지점 도달 성공률을 보인다. 이는 학습 초기부터 매우 도전적인 환경을 마주하게 되면 collective navigation을 학습하기 어렵기 때문이다. 반면에, curriculum을 사용하여 학습을 진행한 경우 아주 쉬운 난이도의 환경부터 점진적으로 학습을 진행함으로써 curriculum 개념을 사용하지 않은 방법보다 높은 목표지점 도달 성공률을 달성할 수 있다. [Table 6]에서 볼 수 있듯이 평균 비행 시간 측면에서 curriculum을 사용한 방법이 curriculum을 사용하지 않은 방법보다 더 빠르게 목표지점에 도달하였고, [Table 7]에서 볼 수 있듯이 flock의 중심과 UAV 간의 평균 거리에서도 curriculum을 사용한 방법이 curriculum을 사용하지 않은 방법에 비해 더 작은 값을 가지므로, 더 나은 flocking behavior를 보인다.

또한, [Fig. 15]와 [Table 5]를 살펴보면 총 10대로 학습시킨 신경망을 이용한 10대의 목표지점 도달 성공률 49%와 총 6대로 학습시킨 신경망을 이용한 10대의 목표지점 도달 성공률 46%의 차이는 아주 미비하다는 것을 알 수 있다. 이를 통해 효율적인 관찰 가능한 이웃 UAV 수를 설정함으로써 적은 수의 UAV로 학습시킨 신경망으로 더 많은 수의 UAV를 성공적으로 제어할 수 있음을 알 수 있으며, 이는 제안된 알고리듬의 확장성을 보여준다.

또한, 학습 수렴 속도 측면에서도 curriculum의 영향을 확인

하였다. 최종 난이도의 환경에서 UAV 6대, 관찰 가능한 이웃 UAV 수 5대를 이용하여 학습을 진행하였다. 학습 수렴 속도는 [Fig. 16]에서 볼 수 있듯이 curriculum을 사용한 방법이 더 빠르고 안정적으로 학습된다. 반면에 curriculum을 사용하지 않은 방법은 curriculum을 사용한 방법에 비해 더 학습 시간이 더 오래 걸리고 최종 성능 또한 좋지 못했다.

[Table 5] Success rate of reaching the target

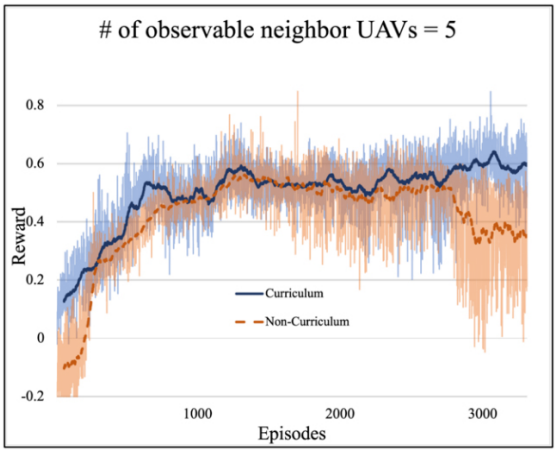
Number of UAVs	Non-Curriculum / Curriculum *SR* (%)
6 UAVs	38 / 89
8 UAVs	29 / 65
10 UAVs	31 / 46
12 UAVs	22 / 37
15 UAVs	11 / 18

[Table 6] Average episodic length

Number of UAVs	Non-Curriculum / Curriculum *AEL* (time steps)
6 UAVs	465.7 / 417.6
8 UAVs	463.6 / 392.6
10 UAVs	443.1 / 379.0
12 UAVs	400.3 / 376.3
15 UAVs	381.4 / 372.9

[Table 7] Average distance between flock center and UAVs

Number of UAVs	Non-Curriculum / Curriculum *ADFCU* (cm)
6 UAVs	0.85 / 0.90
8 UAVs	1.15 / 1.01
10 UAVs	1.32 / 1.03
12 UAVs	1.73 / 1.06
15 UAVs	1.96 / 1.04



[Fig. 16] Effect of curriculum learning

3.3.2 기존 연구와의 비교

본 섹션에서는 제안된 알고리듬과 기존 방법[19]의 성능을 비교한다. 기존 방법은 [19]에서 사전 학습된 모델을 사용하지 않고 [Fig. 11]과 같이 본 논문과 동일한 학습 환경에서 새롭게 학습했다. Curriculum 방법을 사용하지 않았기 때문에 학습 난이도는 10단계로 고정했다. 학습을 위한 UAV 수는 10대를 이용한다.

[19]에서 제안된 방법은 여러 대의 UAV가 flocking을 이루며 장애물이 있는 환경에서 목표지점까지 도달하기 위해 심층 강화학습을 사용하였다. 하지만 장애물이 sparse하게 배치된 환경에 최적화된 observation 및 reward 함수를 사용하고 관찰 가능한 이웃 UAV 수는 3으로 임의로 설정하였으며 curriculum 역시 사용하지 않았다.

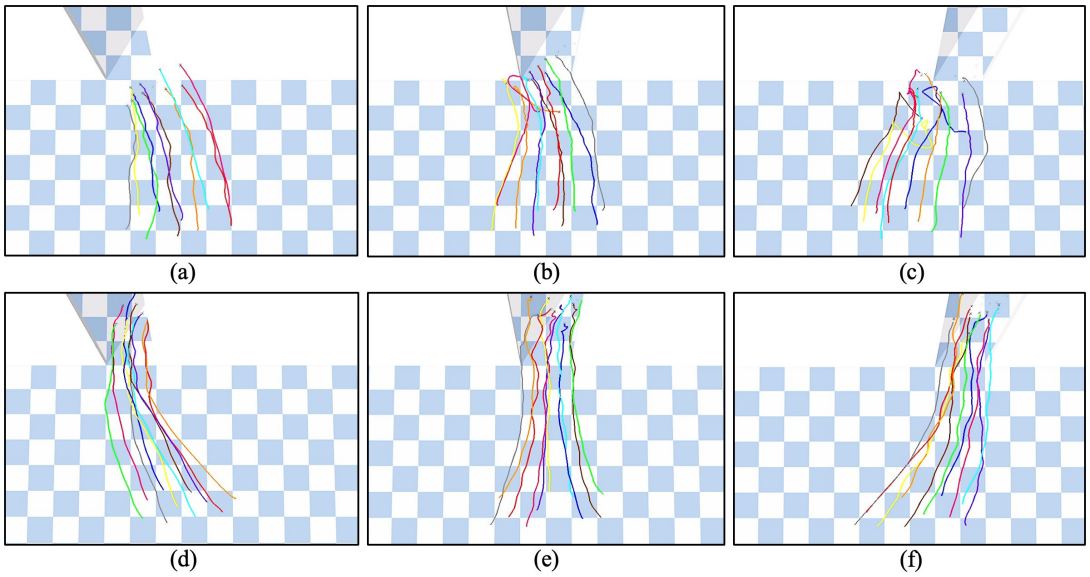
[Fig. 17]과 [Table 8]은 제안된 알고리듬이 [19]에서 제안된 방법과 비교할 때 더 뛰어난 성능을 가짐을 보여준다. 먼저, 목표지점 도달 성공률에서 유의미한 향상을 보였다. [19]에서 제안된 방법은 좁은 틈이 있는 환경에서 6%의 목표지점 도달 성공률을 보인 반면, 제안된 알고리듬은 46%의 목표지점 도달 성공률을 보였다. 이는 기존 방법에 비해 제안된 알고리듬의 강건성을 보여주는 결과이다. 또한, 평균 에피소드 길이는 본 논문에서 제안된 알고리듬이 [19]에서 제안된 방법 보다 더 작은 값을 가지기 때문에 더 짧은 시간 내에 목표지점에 도달할 수 있음을 보여준다. 마지막으로, flock의 중심과 UAV 간의 평균 거리는 본 논문에서 제안된 알고리듬과 [19]에서 제안된 방법이 비슷한 값을 가지므로 목표지점 도달 시 비슷한 flocking behavior를 가짐을 보여준다. 이러한 결과를 얻게 된 이유로는 환경 구성과 이웃 정보 활용 방식에 차이가 있고, curriculum의 사용 유무 등이 있을 수 있다. 종합적으로, 본 논문에서 제안된 알고리듬은 기존 방법과 비교하였을 때 더 뛰어난 성능을 보임을 확인하였다.

3.3.3 결과에 대한 논의

[Table 5]는 환경의 난이도가 높아짐에 따라, 제안된 알고리듬의 목표지점 도달 성공률도 점차 줄어드는 것을 보여준다. reward 함수에서 사용된 가중치와 학습 하이퍼파라미터들은 여러 차례의 실험을 통해 선정되었는데, 이 값들을 지속적으

[Table 8] Comparisons with existing method

	Yan et al. (2020)[19]	Proposed
SR (%)	6	46
AEL (time steps)	940.7	379.0
ADFCU (cm)	1.06	1.03



[Fig. 17] Flight paths of UAVs performing collective navigation in a simulation environment with a narrow gap. (a), (b), and (c) are simulation results using the method proposed in [19], and (d), (e), and (f) are simulation results using our proposed algorithm. In (a) and (d), the narrow gap is located at $+2m$ in the y-direction; in (b) and (e), it is located at $0m$ in the y-direction; and in (c) and (f), it is located at $-2m$ in the y-direction. For all scenarios from (a) to (f), obstacles are located at $+5m$ in the x-direction, the narrow gap is $2m$, and the target is located 2m behind the narrow gap

로 조정하며 최적화하는 작업이 필요하다는 점이 드러났다. 현재 제안된 알고리듬의 policy는 과거의 observation에 대한 정보를 충분히 저장하거나 반영하는데 한계가 있다. 이를 극복하기 위해선, long short-term memory (LSTM)[31] 또는 transformer[32]와 같은 아키텍처의 도입을 고려할 수 있다. 이러한 방식들은 신경망이 이전의 데이터를 더 효과적으로 활용하게 해주어, 더욱 강건한 policy 학습이 가능하게 한다.

4. 결론 및 향후 계획

본 논문에서는 군집을 이루는 UAV들의 collective navigation에 대한 주제에 초점을 맞추었다. 다양한 환경과 장애물을 효과적으로 극복하는 동시에 여러 UAV 간의 협력을 최적화하는 것은 매우 중요한 연구 주제이다. 이를 위해, 사전 경로 없이도 flocking을 이루며 좁은 틈을 통과하여 목표지점에 도달할 수 있도록 특화된 학습 환경을 구성하였다. 학습 과정에서 점진적으로 환경의 난이도를 높여가는 task-specific curriculum을 활용함으로써 UAV의 학습 수렴 속도를 높이고 강건한 모델을 학습하였다. 이를 통해 다양한 장애물에 대응할 수 있는 능력을 보였다. 또한, 관찰 가능한 이웃 UAV의 정보 활용 방식을 개선하여 제안된 알고리듬의 확장성을 향상시켰다. 이는 다른 UAV와의 충돌을 최소화하면서 목표지점까지 도달하는데 큰 역할을 하였다. 제안된 접근법의 유효성과 중요성은 ablation study를 통해 강조되었다. 최종적으로 정량적 지표를 통해 제안된 알고리듬의 성능을 검증하였고, 다양한 시뮬레이션 시나리오와 함께 제시된 결과는 본 연구의 접근 방법이 기존 방법보다 우수하다는 것을 입증하였다.

향후 연구 방향으로는 더욱 강건한 policy를 학습할 수 있도록 다양한 신경망 구조를 활용하고, 좁은 통로나 구멍과 같은 장애물이 있는 더 도전적이고 일반적인 환경에서도 collective navigation이 가능하도록 3차원으로 확장하는 작업을 진행할 예정이다. 3차원 확장을 통해 UAV 군집의 활용 범위와 효율성을 더욱 넓힐 수 있을 것으로 기대된다.

References

[1] A. Kurt, N. Saputro, K. Akkaya, and A. S. Uluagac, "Distributed Connectivity Maintenance in Swarm of Drones During Post-Disaster Transportation Applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 6061-6073, Sept., 2021, DOI: 10.1109/TITS.2021.3066843.

[2] X. Li, J. Zhang, and J. Han, "Trajectory Planning of Load Transportation with Multi-Quadrotors Based on Reinforcement Learning Algorithm," *Aerospace Science and Technology*, vol. 116, pp. 106887, Sept., 2021, DOI: 10.1016/j.ast.2021.106887.

[3] W. Yao, Y. Chen, J. Fu, D. Qu, C. Wu, J. Liu, G. Sun, and L. Xin, "Evolutionary Utility Prediction Matrix-Based Mission Planning for Unmanned Aerial Vehicles in Complex Urban Environments," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 2, pp. 1068-1080, Feb., 2022, DOI: 10.1109/TIV.2022.3192525.

[4] D. Shin, H. Moon, S. Kang, S. Lee, H. Yang, C. Park, M. Nam, K. Jung, and Y. Kim, "Implementation of MAPF-Based Fleet Management System," *The Journal of Korea Robotics Society*, vol. 17, no. 4, pp. 407-416, Nov., 2022, DOI: 10.7746/jkros.2022.17.4.407.

[5] J. Lee, "Improved Heterogeneous-Ants-Based Path Planner Using RRT*," *The Journal of Korea Robotics Society*, vol. 14, no. 4, pp. 285-292, Nov., 2019, DOI: 10.7746/jkros.2019.14.4.285.

[6] A. E. Turgut, H. Celikkanat, F. Gokce, and E. Sahin, "Self-Organized Flocking in Mobile Robot Swarms," *Swarm Intelligence*, vol. 2, no. 2, pp. 97-120, Aug., 2008, DOI: 10.1007/s11721-008-0016-2.

[7] T. Vicsek and A. Zafeiris, "Collective Motion," *Physics Reports*, vol. 517, no. 3-4, pp. 71-140, Aug., 2012, DOI: 10.1016/j.physrep.2012.03.004.

[8] F. Wang, J. Huang, K. H. Low, and T. Hu, "Collective Navigation of Aerial Vehicle Swarms: A Flocking Inspired Approach," *IEEE Transactions on Intelligent Vehicles*, May, pp. 1-14, 2023, DOI: 10.1109/TIV.2023.3271667.

[9] B. Volkl and J. Fritz, "Relation Between Travel Strategy and Social Organization of Migrating Birds with Special Consideration of Formation Flight in the Northern Bald Ibis," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 372, no. 1727, Aug., 2017, DOI: 10.1098/rstb.2016.0235.

[10] C. Okasaki, M. L. Keefer, P. A. Westley, and A. M. Berdahl, "Collective Navigation Can Facilitate Passage Through Human-Made Barriers by Homeward Migrating Pacific Salmon," *The Royal Society B*, vol. 287, no. 1937, Oct., 2020, DOI: 10.1098/rspb.2020.2137.

[11] S. T. Johnston and K. J. Painter, "Modelling Collective Navigation via Nonlocal Communication," *Journal of the Royal Society Interface*, vol. 18, no. 182, Sept., 2021, DOI: 10.1098/rsif.2021.0383.

[12] M. Dorigo, G. Theraulaz, and V. Trianni, "Reflections on the Future of Swarm Robotics," *Science Robotics*, vol. 5, no. 49, Dec., 2020, DOI: 10.1126/scirobotics.abe4385.

[13] X. Zhou, X. Wen, Z. Wang, Y. Gao, H. Li, Q. Wang, T. Yang, H. Lu, Y. Cao, C. Xu, and F. Gao, "Swarm of Micro Flying Robots in the Wild," *Science Robotics*, vol. 7, no. 66, May, 2022, DOI: 10.1126/scirobotics.abm5954.

[14] D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," *2011 IEEE International Conference on Robotics and Automation*, Shanghai, China, pp. 2520-2525, 2011, DOI: 10.1109/ICRA.2011.5980409.

[15] E. Soria, F. Schiano, and D. Floreano, "Predictive Control of Aerial Swarms in Cluttered Environments," *Nature Machine Intelligence*, vol. 3, no. 6, pp. 545-554, May, 2021, DOI: 10.1038/s42256-021-00341-y.

[16] A. Loquercio, E. Kaufmann, R. Ranftl, M. Muller, V. Koltun, and D. Scaramuzza, "Learning High-Speed Flight in the Wild," *Science Robotics*, vol. 6, no. 59, Oct., 2021, DOI: 10.1126/scirobotics.abg5810.

[17] A. Singla, S. Padakandla, and S. Bhatnagar, "Memory-Based Deep Reinforcement Learning for Obstacle Avoidance in UAV With Limited Environment Knowledge," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 107-118, Jan., 2021, DOI: 10.1109/TITS.2019.2954952.

[18] M. Kim, J. Kim, M. Jung, and H. Oh, "Towards Monocular Vision-Based Autonomous Flight Through Deep Reinforcement Learning," *Expert Systems with Applications*, vol. 198, Jul., 2022, DOI: 10.1016/j.eswa.2022.116742.

[19] P. Yan, C. Bai, H. Zheng, and J. Guo, "Flocking Control of UAV Swarms with Deep Reinforcement Learning Approach," *2020 3rd International Conference on Unmanned Systems (ICUS)*, Harbin, China, pp. 592-599, 2020, DOI: 10.1109/ICUS50048.2020.9274899.

[20] P. Zhu, W. Dai, W. Yao, J. Ma, Z. Zeng, and H. Lu, "Multi-Robot Flocking Control Based on Deep Reinforcement Learning," *IEEE Access*, vol. 8, pp. 150397-150406, Aug., 2020, DOI: 10.1109/ACCESS.2020.3016951.

[21] W. Wang, L. Wang, J. Wu, X. Tao, and H. Wu, "Oracle-Guided Deep Reinforcement Learning for Large-Scale Multi-UAVs Flocking and Navigation," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10280-10292, Oct., 2022, DOI: 10.1109/TVT.2022.3184043.

[22] C. Yan, C. Wang, X. Xiang, K. H. Low, X. Wang, X. Xu, and L. Shen, "Collision-Avoiding Flocking with Multiple Fixed-Wing UAVs in Obstacle-Cluttered Environments: A Task-Specific Curriculum-Based MADRL Approach," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1-15, Feb., 2023, DOI: 10.1109/TNNLS.2023.3245124.

[23] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum Learning," *The 26th Annual International Conference on Machine Learning*, pp. 41-48, 2009, DOI: 10.1145/1553374.1553380.

[24] Collective Navigation Through a Narrow Gap for a Swarm of UAVs Using Curriculum-Based Deep RL, [Online], https://youtu.be/s0DFgJB6ODw, Accessed: Jan. 13, 2024.

[25] K. Morihiro, T. Isokawa, H. Nishimura, and N. Matsui, "Emergence of Flocking Behavior Based on Reinforcement Learning," *International conference on knowledge-based and intelligent information and engineering systems*, pp. 699-706, 2006, DOI: 10.1007/11893011_89.

[26] C. W. Reynolds, "Flocks, Herds and Schools: A Distributed Behavioral Model," *The 14th Annual Conference on Computer Graphics and Interactive Techniques*, pp. 25-34, 1987, DOI: 10.1145/37401.37406.

[27] J. Panerati, H. Zheng, S. Zhou, J. Xu, A. Prorok, and A. P. Schoellig, "Learning to Fly: A Gym Environment with PyBullet Physics for Reinforcement Learning of Multi-Agent Quadcopter Control," *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Prague, Czech Republic, pp. 7512-7519, 2021, DOI: 10.1109/IROS51168.2021.9635857.

[28] E. Liang, R. Liaw, R. Nishihara, P. Moritz, R. Fox, K. Goldberg, J. Gonzalez, M. Jordan, and I. Stoica, "RLlib: Abstractions for Distributed Reinforcement Learning," *arXiv:1712.09381*, pp. 3053-3062, 2018, DOI: 10.48550/arXiv.1712.09381.

[29] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv:1707.06347*, 2017, DOI: 10.48550/arXiv.1707.06347.

[30] M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, and M.Viale, "Interaction Ruling Animal Collective Behavior Depends on Topological Rather Than Metric Distance: Evidence from a Field Study," *National Academy of Sciences*, vol. 105, no. 4, pp. 1232-1237, Jan., 2008, DOI: 10.1073/pnas.0711437105.

[31] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, Nov., 1997, DOI: 10.1162/neco.1997.9.8.1735.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All You Need," *arXiv:1706.03762*, 2017, DOI: 10.48550/arXiv.1706.03762.

최 명 열

2022 울산과학기술원 기계공학과(공학사)

2022~현재 울산과학기술원 기계공학과 석사과정

관심분야: 무인이동체 군집 제어, Navigation, 강화학습

유 영 빈

2005 연세대학교 정보산업공학과(공학사)

2007 연세대학교 컴퓨터과학과(공학석사)

2007 삼성전자 정보통신총괄 통신연구소 연구원

2008~현재 LIG넥스원 C4I연구소 수석연구원

관심분야: MANET, 국방 전술네트워크, 군집무인기 네트워크

신 우 재

2023 울산과학기술원 기계공학과(공학사)

2023~현재 울산과학기술원 기계공학과 연구원

관심분야: SLAM, 강화학습

이 민

2006 아주대학교 전자공학부(공학사)

2014 아주대학교 전자공학부(공학박사)

2014~현재 LIG넥스원 C4I연구소 수석연구원

관심분야: 무선 MAC 프로토콜, 국방 전술네트워크, 군집무인기 네트워크

김 민 우

2019 울산과학기술원 기계공학과(공학사)

2019~현재 울산과학기술원 기계공학과 박사과정

관심분야: 영상기반 충돌회피, 역강화학습

오 현 동

2004 KAIST 항공우주공학과(공학사)

2010 KAIST 항공우주공학과(공학석사)

2013 Cranfield University 항공우주공학과(공학박사)

2013~2014 영국 University of Surrey 박사 후 연구원

2014~2016 영국 Loghborough University 조교수

2016~현재 울산과학기술원 부교수

관심분야: 무인이동체 의사 결정, 협력 제어, 경로 계획, 비선형 유도/제어, 센서/정보 융합

박 휘 성

2012 성균관대학교 전자전기공학과(공학사)

2014 KAIST 전기 및 전자공학과(공학석사)

2014~현재 국방과학연구소 선임연구원

관심분야: 무인기 통신, 무선 광통신, 무인이동체 군집 제어, 경로 계획