https://doi.org/10.7840/kics.2023.48.2.176

자율적인 UAM 시스템의 효율적인 무인 정보수집 및 감시를 위한 멀티 에이전트 기반 심층 강화학습

박 찬 영◆, 김 규 선*, 이 경 진**, 윤 일 수°

Multi-Agent Deep Reinforcement Learning for Efficient Unattended Information Gathering and Monitoring of Autonomous UAM Systems

Chanyoung Park◆, Gyu Seon Kim*, Kyeongjin Lee**, Ilsoo Yun°

요 약

멀티 에이전트 심층 강화학습은 여러 에이전트 간의 통신을 통해 에이전트들이 협력적으로 공동의 목표를 달성하는 기계학습이다. 이러한 심층 강화학습 기술을 통해서, 도심 환경에서 교통정보 수집 및 보안을 위해 필수적인 CCTV의 감시 역할을 여러 개의 도심 항공 모빌리티 (UAM, Urban Air Mobility)가 대체할 수 있다. 기존의 CCTV는 고정된 위치에서 한정적인 시각 정보를 제공할 수 있지만, UAM을 통한 자율적인 CCTV 시스템을 구축하면 실시간으로 추적할 대상의 위치에 따라 유연하고 신뢰성 있는 시각 정보를 제공할 수 있다. 따라서, 본 논문은 에이전트 간 통신 임무를 수행하는 CommNet 알고리즘을 통해 여러 UAM들이 효율적으로 정보수집 및 감시가 가능한 시스템을 구축하는 기법을 제안한다.

키워드 : 심층 강화학습, 멀티 에이전트, UAM, CCTV
Key Words : Deep Reinforcement Learning, Multi-Agent, UAM, CCTV, CommNet

ABSTRACT

Multi-agent deep reinforcement learning is machine learning in which agents cooperate to achieve a common goal through communication between multiple agents. With this deep reinforcement learning technology, multiple Urban Air Mobility (UAM) can replace the surveillance role of CCTV, which is essential for security and data collection in urban environments. Existing CCTV can provide limited visual information in a fixed location, but building and autonomous CCTV system through UAM can provide flexible and stable visual information according to the location of the surveillance target in real-time. Therefore, this paper proposes a method to build a system where multiple UAMs efficiently perform monitoring services through the CommNet algorithm, which plays the role of inter-agent communication.

※ 본 연구는 2022년도 정부(경찰청)의 재원으로 과학치안진흥센터의 지원을 받아 수행되었습니다. (No. 092021C29S01000, 네트워크 제어를 위한 교통정체 및 혼잡 운영관리 기술 개발)
◆ First Author : Korea University School of Electrical Engineering, cosdeneb@korea.ac.kr, 학생회원
° Corresponding Author : Ajou University School of Transportation System Engineering, Ilsooyun@ajou.ac.kr, 정회원
* Inha University School of Aerospace Engineering, kingdom0545@inda.edu
** Ajou University School of Transportation System Engineering, aajom1012@ajou.ac.kr, 학생회원
논문번호 : 202211-275-B-RN, Received November 8, 2022; Revised November 23, 2022; Accepted November 26, 2022

Ⅰ. 서 론

심층 강화학습은 기계 학습을 위한 기술 중 하나이며, 자율 주행[1], 통신[2,3], 의료[4], 교통운영[5] 등 다양한 분야에서 활용되고 있는 기계 학습 기술 중 하나이다. 기계는 심층 강화학습을 통해 여러 번의 학습 과정에서 시행착오를 겪으며 자신이 관찰한 상태 정보를 가지고 문제를 해결하기 위한 해결책을 스스로 학습한다. 이때 학습하는 주체인 기계가 에이전트(Agent)에 해당하며, 이러한 에이전트가 여러 개 존재하는 환경을 멀티 에이전트 환경이라고 한다. 즉, 멀티 에이전트 심층 강화학습은 멀티 에이전트 환경에서 각각의 에이전트가 심층 강화학습을 통해 문제를 해결하는 과정을 의미한다. 단일 에이전트 심층 강화학습에서는 문제를 해결하는 주체가 하나이기 때문에 오직 자신이 관찰한 정보에 대한 최적의 의사결정을 내리면 된다. 그러나, 멀티 에이전트 심층 강화학습에서는 다른 에이전트도 존재하기 때문에 공동의 목표를 협력적으로 달성하기 위해서는 자신이 관찰한 상태 정보만이 아니라, 다른 에이전트가 관찰한 상태 정보도 고려하여 최적의 의사결정을 내려야 한다. 따라서 단일 에이전트와 같은 방법으로 에이전트를 학습시키면 협력적으로 목표를 달성하는 데에 있어 한계를 가지고 있다. 또한, 실제 환경은 훨씬 복잡하고 상태 정보가 고차원적이며 에이전트가 여러 개인 경우가 많으므로 본 논문에서는 멀티 에이전트가 협업적으로 공동의 목표를 달성하기 위한 기법을 제안한다.

본 논문에서는 2장에서 강화학습에 대한 기본 개념과 연구 동향을 서술한다. 3장에서는 멀티 에이전트 환경에서 에이전트들이 협력적으로 공동의 문제를 해결할 수 있도록 에이전트 간의 통신 기능을 제공하는 CommNet[6] 알고리즘을 소개한다. 이를 이용한 멀티 에이전트 심층 강화학습을 통해 여러 UAM이 자신이 관찰한 상태 정보를 공유하며 협력적으로 최적의 무인 감시 시스템을 구축하는 기법을 제안하고, 이에 대한 성능 평가를 서술한 뒤 4장에서는 결론을 맺는다.

Ⅱ. 강화학습 개요

2.1 MDP

강화학습은 에이전트와 환경(Environment)으로 구성되어 있다. 에이전트가 현재 상태(State)에서 행동(Action)을 취하면, 그에 따라 환경으로부터 보상(Reward)을 얻고 다음 상태로 넘어가게 된다. 이러한 과정이 루프처럼 반복되는 것을 순차적 의사결정 문제라고 하며, 이러한 문제를 해결하기 위한 최적의 정책(Policy)을 학습하는 것이 강화학습이다. 정책이란, 에이전트의 현재 상태를 입력으로 받아서 에이전트가 취할 수 있는 행동을 출력하는 함수이다. 에이전트는 자신의 정책에 종속적으로 행동을 취하기 때문에 높은 보상을 받기 위해서는 최적의 정책을 학습하는 것이 핵심이다.

강화학습에서 에이전트의 기본적인 의사결정 모델은 마르코프 결정 프로세스(MDP, Markov Decision Process)이다. MDP는 과거와는 무관하게 오로지 현재에 의해 미래가 결정되는 마르코프 성질을 따른다. MDP의 구성 요소는 (S, A, P, R, γ)이며, 각각 상태 집합, 행동 집합, 전이 확률, 보상 함수, 그리고 감쇠 인자이다. 상태 집합과 행동 집합은 앞서 말한 것과 같이 에이전트가 가질 수 있는 상태와 취할 수 있는 행동의 집합이며, 보상 함수는 특정 상태에서 특정 행동을 취했을 때 보상을 출력하는 함수를 의미한다. 전이 확률은 에이전트가 현재 상태에서 다음 상태로 도착할 확률을 의미한다. 감쇠 인자는 0에서 1 사이의 값을 가지는데, 에이전트가 미래에 얻을 보상에 비해 현재 얻을 보상에 얼마나 가중치를 줄 것인지에 대한 요소이다. 감쇠 인자는 에피소드를 진행해가며 계속해서 곱해지는 가중치이기 때문에 가중치의 값이 0에 가까울수록 미래에 대한 보상이 0에 가까워지는 속도가 빨라진다. 즉, 감쇠 인자가 작을수록 현재 상태에 받을 수 있는 보상에 더 집중하게 되는 것이다.

정리하자면, 에이전트는 현재 상태를 관측하고, 이를 정책의 입력으로 넣어 가장 큰 보상을 얻을 수 있는 행동을 예측한다. 그리고 예측한 행동을 취함으로써 다음 상태로 넘어가게 된다. 이때 현재 상태 s에서 특정한 행동 a를 취하도록 하는 정책 함수를 수식 1과 같이 표현할 수 있다.

$$\pi(s|a) = P[A_t = a|S_t = s] \tag{1}$$

이렇게 특정 상태에서 보상이 가장 높은 행동을 예측하는 원리는 벨만 최적 방정식(Bellman optimality equation)을 이용하여 행동의 가치를 평가하는 것이다. 벨만 최적 방정식을 이용해 현재 상태 s에서 가장 보상을 높일 수 있는 행동 a을 취하는 것을 수식 2와 같이 표현할 수 있다.

$$q_*(s_t, a_t) = R + \gamma \sum_{s' \in S} P^a_{ss'} \max_{a'} q_*(s_{t+1}, a_{t+1}) \tag{2}$$

여기서 $q(s,a)$는 현재 상태 s에서 취할 수 있는 행

동 a의 가치에 해당하는데, 벨만 최적 방정식에서는 가장 최적의 가치를 가지는 행동의 가치인 $q_*(s,a)$ 를 찾는다. $P_{ss'}^{a}$는 현재 상태 s에서 행동 a을 취하여 다음 상태 s'로 전이할 때 환경으로부터 받는 전이 확률에 해당한다. 벨만 최적 방정식을 이용하여 가장 높은 보상을 받을 수 있는, 즉 가장 가치가 높은 행동을 취하는 것이 벨만 최적 방정식이며 이를 근간으로 수식 3과 같이 현재 상태에서 가장 가치가 높은 행동을 취하는 최적의 정책 함수를 학습시키는 것이 강화학습이다.

$$\pi_* = arg\max_a q_*(s_t, a_t) \tag{3}$$

이러한 원리의 강화학습은 순차적인 문제를 해결하는 데에 효과적이다. 강화학습을 푸는 대표적인 방법으로 동적 계획법(Dynamic Programming)[7]이 있지만, Pseudo-Polynomial[8]의 연산 복잡도를 가지기 때문에 상태 정보가 단순한 문제만 해결할 수 있고 알파고[9]와 같이 매우 많은 상태 정보를 가지는 복잡하고 정교한 문제를 해결할 수 없다. 따라서 다음 항에서는 강화학습에서 에이전트의 정책을 효과적으로 학습하기 위해서 개발된 알고리즘들을 서술한다.

2.2 Q-Learning[10]

Q 러닝(Q-Learning)은 강화학습 기반의 알고리즘이기 때문에 기본적으로 수식 4와 같이 벨만 최적 방정식을 대상으로 학습을 진행한다.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(R + \gamma \max_{a'} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \tag{4}$$

여기서 a는 얼마나 빠른 속도로 학습을 진행할지 결정하는 학습률(Learning Rate)이다. 즉, Q 러닝에서는 현재 상태 s에서 행동 a를 취할 때 받을 수 있는 보상의 기댓값을 출력하는 가치함수에 해당하는 Q 함수인 $Q(s,a)$를 학습하는 과정을 통해 최적의 정책을 학습한다.

2.3 DQN

딥 Q 러닝(DQN, Deep Q-Network)은 위에서 설명한 Q 러닝에서 발전된 알고리즘이다. DQN은 Q 함수를 인공 신경망(Neural Network)을 사용하여 표현한 알고리즘이다. 인공 신경망을 사용하였기 때문에 상태 정보가 무수히 많은 고차원적인 환경에서도 에이전트가 자신의 정책을 학습할 수 있다. DQN으로 학습하는 에이전트는 자신의 명시적인 정책이 없는 Model-free 강화학습이다. 명시적인 정책을 학습하는 대신, 신경망의 파라미터 θ를 학습한다. 이때 DQN에서 신경망을 학습하기 위해서 벨만 최적 방정식을 정답으로 설정하고, 현재 상태에 대한 신경망의 추측을 통해 손실 함수를 수식 5와 같이 정의하였다.

$$L(\theta) = E\left(R + \gamma \max_{a'} Q_\theta(s_{t+1}, a_{t+1}) - Q_\theta(s_t, a_t)\right)^2 \tag{5}$$

다음과 같이 손실 함수를 기댓값으로 정의하여 Q 함수를 근사하였으며, 이를 Q 러닝과 같이 벨만 최적 방정식에 가까워지도록 학습을 진행하는 식으로 표현하면 수식 6과 같다.

$$\theta' \leftarrow \theta + \alpha(R + \gamma \max_{a'} Q_\theta(s_{t+1}, a_{t+1} - Q_\theta(s_t, a_t)) \times \nabla_\theta Q_\theta(s_t, a_t)) \tag{6}$$

DQN은 신경망의 도입과 함께, 두 가지의 기법을 새롭게 제안하였다. 먼저, Experience Replay Buffer를 사용하여 에피소드를 진행하며 에이전트가 환경에서 상태 전이를 하며 겪은 경험(상태 정보, 행동, 보상)을 리플레이 버퍼에 저장하는 기법이 있다. 리플레이 버퍼에 경험, 즉 데이터가 쌓이면 미니 배치(Mini-batch) 크기만큼 데이터를 임의 추출하여 인공 신경망의 파라미터 학습에 사용한다. 연속된 데이터에 대한 상관성이 학습의 안정성과 성능에 좋지 않은 영향을 미치지만. 리플레이 버퍼를 사용함으로써 학습에 사용하는 데이터의 연속성이 줄어든다. 따라서, 리플레이 버퍼를 통해 데이터 간의 상관성을 줄일 수 있게 되어 에이전트의 학습 성능을 향상할 수 있다.

DQN에서 제안한 나머지 하나의 기법은 타겟 네트워크(Target Network)이다. 학습을 진행하는 인공 신경망과 별개로 타겟 네트워크를 생성한 뒤, 이에 대한 파라미터를 업데이트하지 않고 얼려 놓는다. 기존의 인공 신경망을 학습하다가 어느 정도의 시점이 지나면, 학습을 진행하던 인공 신경망의 파라미터를 기반으로 하여 얼려 두었던 타겟 네트워크 신경망의 파라미터를 업데이트한다. 업데이트의 횟수를 제한하는 타겟 네트워크를 사용하는 이유는 수식 1에서 찾아볼 수 있다. 손실 함수에서 정답으로 사용하는 벨만 최적 방정식은 네트워크의 파라미터 θ에 의존적이다. 따라서, 정답이 네트워크 신경망의 파라미터에 의존적이기 때문에 θ가 업데이트될 때마다 정답도 같이 변화하게 된다. 이러한 현상은 학습의 안정성을 떨어뜨려, 에이전트의 보상이 수렴하지 못하고 발산할 수도 있다. 따

라서 타겟 네트워크를 사용함으로써 에이전트가 학습을 안정적으로 최적화할 수 있다.

DQN의 등장으로 인해 일반적인 강화학습에 인공신경망을 사용한 심층 강화학습(Deep Reinforcement Learning)의 발전이 시작되었다. 본 논문에서는 심층 강화학습을 통해 학습하는 멀티 에이전트의 시스템을 최적화하기 위한 기법인 CommNet 기법을 제안한다.

Ⅲ. 실험 및 평가

3.1 실험 설계

이번 장에서는 심층 강화학습 기반의 멀티 에이전트들이 효과적으로 무인 자료 수집 및 감시 시스템을 구축할 수 있는 기법을 제안한다. 그 전에, 시스템을 구축하기 위한 실험의 설계에 관해 서술한다. 먼저, 여러 UAM이 최적의 무인 감시 시스템을 구축하기 위해서는 자율적으로 감시할 대상의 위치에 따라 사방위로 움직이며 최대한 많은 대상을 감시하는 공동의 목표를 달성해야 한다. 그러나 UAM이 관찰할 수 있는 범위는 기계의 물리적인 한계에 의해 제한되어 있으므로 모든 감시 대상의 위치를 알 수 없다. 또한, 자신이 관찰한 정보만을 가지고 최대한 많은 대상을 감시하기 위한 최적의 의사결정을 내린다면 다른 UAM과 충돌하거나 감시하는 범위가 겹치는 현상이 발생하기 때문에 무인 감시 시스템의 성능을 크게 떨어뜨릴 수 있다. 따라서 CommNet 알고리즘을 이용한 에이전트 간 통신을 통해 자신이 관찰한 정보만이 아니라 다른 UAM이 관찰한 정보를 바탕으로 공동의 목표를 달성해야 한다.

이번 장에서는 CommNet 알고리즘을 이용한 멀티 에이전트 심층 강화학습을 사용하여 여러 UAM이 협력적으로 효과적인 자료 수집 및 감시 시스템을 구축하는 기법을 제안한다. 먼저 무인 감시 시스템의 에이전트에 해당하는 UAM 모델의 사양을 소개하고, 구축한 시스템의 환경을 서술한다. 이후, 구축한 무인 감시 시스템의 성능을 여러 측면에서 평가한다.

3.1.1 UAM 모델

이번 항에서는 설계한 실험에서 에이전트에 해당하는 UAM의 모델과 자세한 사양에 관해 설명한다. 이후 UAM의 실제 사양을 사용하여 UAM 에이전트의 행동에 따른 에너지 방전량을 구하고, 이를 기반으로 구체적인 실험을 설계한다.

실험에서 에이전트에 해당하는 UAM의 모델은 JOBY AVIATION 사의 S4이다. UAM은 전동화된 비행체로, 화석연료를 사용하는 내연기관 기반의 항공기와 다르게 배터리를 사용한다. 그러므로 비연료소모율(SFC, Specific Fuel Consumption) 고려 없이 항공역학적으로 계산된 에너지를 통해 UAM의 에너지를 나타낼 수 있다. 수식 7은 UAM이 목표 고도까지 올라가며, 제자리 비행을 할 때의 소모 에너지이다.

$$\mu_m(t) = \underbrace{\frac{\delta}{8}\rho s A \Omega^3 R^3}_{P_0} + \underbrace{(1+k)\frac{W^{3/2}}{\sqrt{2\rho A}}}_{P_i} \tag{7}$$

P_0는 로터 블레이드를 회전시키는 데 필요한 에너지이며, P_i는 유도 항력(Induced drag)을 극복하기 위한 에너지이다. 유도 항력은 양력에 의해 발생하는데, 이는 날개 길이가 유한한 3차원 날개이기 때문에 발생한다. 이는 날개 끝에서 발생하는 날개 끝 와류(wing tip vortex)에서 발생하는 하강기류(downwash)에 의해 양력이 비행체 뒤로 밀리면서 발생하는 수평 방향 분력이 만드는 항력이다. 날개 끝 와류는 블레이드 밑면의 높은 압력의 기체가 압력이 낮은 블레이드 윗면으로 휘몰아 올라가면서 발생하는 와류로, 비행체의 에너지를 기술하는 과정에서 필수적으로 고려돼야 한다.

δ는 항력계수로 유체에서 물체의 항력을 정량화 하기 위한 무차원 계수이고, ρ는 공기의 밀도로 고도가 상승함에 따라 비선형적으로 감소한다. s는 로터의 고형비로, 로터 디스크 면적에 대한 로터 블레이드 면적의 비율이다. A, Ω, R, W는 순서대로 로터 디스크 면적, 블레이드의 각속도, 로터의 반지름, payload를 포함한 UAM의 총무게이다. k는 UAM의 로터 블레이드 가로세로비(AR, Aspect Ratio)에 반비례하는 유도 항력 계수로, k에 양력계수의 제곱이 곱해져 항력이 증가한다. 유도 항력에 의해 증가한 항력을 극복하기 위해 UAM은 에너지를 더 사용해야 하므로 k는 수식 7와 수식 8에서 고려돼야만 한다. UAM이 전기 수직이착륙(eVTOL, Electric Vertical Take-Off and Landing)을 통해 목표 고도까지 상승하고 전진 추진을 하게 되면, x축에 대한 분력이 추가되기 때문에, 추진 전력 소비 $\mu_m(t)$를 고려해야만 한다. 추진 전력 소비 $\mu_m(t)$는 수식 8과 같다.

$$\mu_m(t) = P_0\left(1+\frac{3v^2}{U_{tip}}\right) + P_i\left(\sqrt{1+\frac{v^4}{4v_0^4}}-\frac{v^2}{2v_0^2}\right)^{\frac{1}{2}} + \frac{1}{2}d_0\rho s A v^3 \tag{8}$$

v는 UAM의 순항속도로 블레이드 회전속도, 로터

표 1. UAM 모델 사양
Table 1. Specification of UAM model

Notation	Value
비행 속도, v	$73.762m/s$
기체 질량, m	$1{,}815{,}000g$
기체 무게, W= mg	$17{,}799N$
로터 길이, R	$1.45m$
로터의 디스크 면적, A = πR^2	$6.61m^2$
블레이드 수, b	5
로터 고형비, $s = \frac{0.2231b}{\pi R}$	0.2449
블레이드 각속도, Ω	$78rad/s$
블레이드 끝단 회전속도, U_{tip}	$112.776m/s$
공기 밀도, ρ	$1{,}225g/m^3$
동체 항력 비, $d_0 = \frac{0.02975}{sA}$	0.01
유도 속도, $v_0 = \sqrt{\frac{W}{s\rho A}}$	$26.45m/s$
항력계수, C_d	0.045
유도항력 계수, k	0.052

의 틸트 각에 따라 바뀐다. U_{tip}은 블레이드 끝단의 회전속도로 블레이드 각속도 Ω에 로터 반지름 R을 곱하여 유도할 수 있다. d_0는 동체 항력 비(fuselage drag ratio)이다. v_0는 평균 유도속도(mean rotor induced velocity)로 날개 끝 와류에 의해 움직이는 유동의 평균 속도이다.

수식 7과 수식 8에 사용되는 물리량에 대한 값은 표 1을 통해 확인할 수 있다. 표 1의 값은 JOBY AVIATION 사에서 제공하는 s4 모델에 대한 항공 역학적인 물리량들이다. 제공되지 않은 물리량은 제공된 값을 통해 항공역학적으로 계산되었다. 단위는 값 뒤에 기재되어 있으며, 단위가 없는 값은 물리적 차원이 없는(dimensionless) 값이다.

3.1.2 실험 환경

이번 항에서는 에이전트가 존재하는 환경에 관해 서술한다. 위의 표 2는 실험 초기 환경을 설정하기 위한 하이퍼 파라미터에 해당한다. 멀티 에이전트 환경에서는 여러 개의 에이전트가 존재한다. 본 논문에서 UAM, 즉 감시하는 주체가 되는 에이전트의 수는 4개이며, 25명의 감시할 수 있는 대상의 수가 임의로 분포하고 있다. UAM이 대상을 감시할 수 있는 하나의 에피소드의 단위는 실제 모델의 사양을 고려했을 때 40분이며, 이러한 에피소드가 총 100,000번 반복하며 에이전트들의 학습이 진행된다. 인공 신경망의 높이는

표 2. 실험 환경 설정
Table 2. Simulation setup

Notation	Value
에이전트 수, M	4
감시 대상의 수, N	25
에피소드별 시간, T	40분
학습을 진행한 에피소드 수	100,000
신경망 층 높이, L	6개
학습률	0.001
초기 입실론 값, ε	0.3
에피소드 진행에 따른 입실론의 감쇄 정도	0.0001
입실론의 최솟값	0.01

총 6개이며, 학습의 속도를 나타내는 학습률은 0.001이다. 또한 강화학습에서 에이전트들의 충분한 탐색을 위해 $\varepsilon - greedy$방법을 사용한다. 여기서 입실론(ε)의 값은 학습한 정책에 상관없이 임의의 행동을 취할 확률을 의미한다. 따라서 학습이 많이 진행되지 않은 학습 초기의 입실론 값을 크게 설정하여 에이전트가 특정 상태에서 다양한 경험을 할 수 있도록 하고, 정책의 학습이 어느 정도 진행되면 임의의 행동을 취하기보다는 학습한 정책에 따른 행동을 내리는 것에 좀 더 가중치를 두어야 한다. 따라서 하나의 에피소드가 진행하면 입실론의 크기를 0.0001씩, 최대 0.01까지 감소시켜 학습을 마칠 시점에는 에이전트가 관측하는 상태 정보와 무관하게 임의의 행동을 취할 확률을 크게 줄였다. 즉, 되도록 학습한 정책에 따라서 에이전트가 행동을 선택하도록 구현하였다.

$$R = \frac{Support\ Rate}{1+Overlapped\ Rate} \tag{9}$$

다음으로 강화학습에서 핵심이 되는 보상 식에 관해서 서술한다. 앞서 말했듯이 강화학습은 특정 상태에서 보상을 높이는 행동을 취하는 정책을 학습하는 것이 가장 기본적인 원리이기 때문에, 달성하고자 하는 목표를 수식으로써 보상 식으로 잘 설계하는 것이 중요하다. 감시 시스템의 성능을 높이기 위해서는 멀티 에이전트가 감시하는 대상의 수가 증가해야 하고, 더 많은 범위를 커버하기 위해서 에이전트 간 감시하는 범위가 덜 겹쳐야 한다. 이러한 요소를 고려하여 수식적으로 설계한 보상 식이 수식 9에 해당한다. 즉, 멀티 에이전트가 감시하는 대상이 많을수록, 감시하는 구역이 겹친 정도가 작을수록 에이전트가 높은 보상 R을 받게끔 보상 식을 설계하였다. 강화학습에서 어

떻게 보상 식을 설계하는지에 따라 시스템의 성능이 크게 달라지기 때문에 에이전트가 원하는 목표를 효과적으로 달성할 수 있도록 적절하게 보상 식을 설정해야 한다.

3.2 모델 학습

3.2.1 CommNet

이번 항에서는 멀티 에이전트 심층 강화학습을 통해 에이전트들이 효과적으로 협업할 수 있도록 에이전트 간 통신 기능을 수행하는 CommNet 알고리즘을 소개한다. 그림 2는 CommNet의 전반적인 구조를 나타낸다. CommNet을 이용해 학습하는 M개의 각 에이전트는 L개의 신경망 층을 가진다. 신경망의 입력으로 자신이 관찰한 상태 정보를 가지며, 모든 신경망 층을 통과한 이후에 Softmax 층을 통과하며 현재 상태에서 에이전트가 취할 수 있는 모든 행동에 대해서 0에서 1 사이의 정규화 된 값을 출력한다. 에이전트는 가장 큰 값을 가지는 행동을 취함으로써 현재 상태에서 다음 상태로 전이하게 되고, 이 과정이 반복되면서 에피소드를 진행한다. 그림 1에 나와 있는 것처럼 하나의 신경망 층에서 다음 신경망 층으로 가는 과정을 살펴보자.

$$h_m^1 = f(s_m) \tag{10}$$

먼저, 인코더 함수 $f()$는 신경망의 입력에 해당하는 m번째 에이전트가 관찰한 상태 정보 s_m을 받아 첫 번째 신경망 층의 은닉 변수 h_m^1 을 출력한다.

$$c_m^l = \frac{1}{M-1}\sum_{m \neq m'} h_{m'}^l \tag{11}$$

따라서 M개의 에이전트는 인코딩 과정을 통해 독립적인 M개의 은닉 변수를 갖게 된다. 이때 각 에이전트가 관찰한 정보를 공유하기 위해서 자신을 제외한 모든 에이전트의 은닉 변수에 대해 평균을 취한다. 이러한 계산 과정을 통해 구한 통신 변수 c_m^l을 통해서 자신이 관찰한 정보만이 아니라 다른 에이전트가 관찰한 정보도 고려할 수 있게 된다.

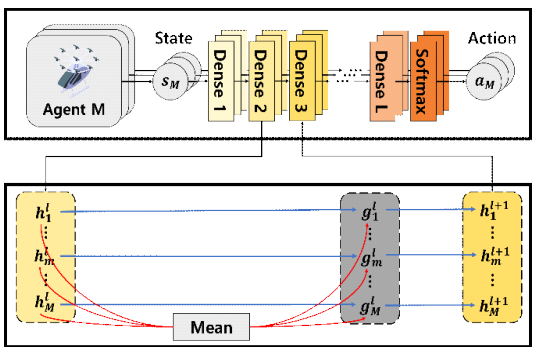


그림 1. CommNet 구조
Fig. 1. Architecture of CommNet

$$h_m^{l+1} = g(h_m^l, c_m^l) \tag{12}$$

마지막으로 m번째 에이전트의 현재 신경망 층 l에서의 은닉 변수 h_m^l와 자신을 제외한 다른 에이전트들의 은닉 변수들로부터 구한 통신 변수 c_m^l을 활성화 함수 $g()$의 입력으로 같이 넣는다. 이를 통해 다음 신경망 층인 l+1에서의 입력에 해당하는 은닉 변수 h_m^{l+1}을 구하게 되고, 이러한 과정을 신경망 층의 높이 L에 도달할 때까지 반복한다. 모든 신경망을 거쳤으면, 최종적으로 구한 은닉 변수 h_m^L을 Softmax 함수의 입력으로 넣으면, m번째 에이전트가 현재 상태에서 취할 수 있는 행동들에 대한 정규화 된 값을 얻을 수 있다.

이렇게 CommNet 알고리즘은 에이전트마다 독립적인 은닉 변수 h_m^l과 함께, 모든 에이전트에 대해 종속적인 통신 변수 c_m^l을 사용하는 단순한 방법으로 멀티 에이전트 환경에서 에이전트들이 자신의 상태 정보만이 아니라 다른 에이전트의 상태 정보도 고려하며 협력적으로 공동의 목표를 달성할 수 있다는 장점이 있다. 그러나, CommNet을 위한 에이전트 간 통신을 고려하기 위해 에이전트의 정책 학습 시 추가적인 연산이 필요하다는 단점이 있다.

3.2.2 Actor-Critic[12]

이번 항에서는 본 논문에서 사용한 심층 강화학습의 정책 네트워크와 가치 네트워크를 함께 학습하는 액터-크리틱(Actor-Critic)에 대해서 서술한다. 먼저, Actor-Critic을 알기 전에 아래 수식 13의 Policy gradient[13]에 대해 알아야한다.

$$\nabla_\theta J(\theta) = E_{\pi_\theta}[\nabla_\theta \log \pi_\theta(s,a) \times Q_{\pi_\theta}(s,a)] \tag{13}$$

강화학습에서 Policy gradient는 최적화의 대상이 되는 목적 함수(Objective function)를 최대화하기 위해서 파라미터를 업데이트하는 방향을 제시한다. 강화학습의 목적은 보상을 최대화하는 것이기 때문에 목적 함수에 정책에 해당하는 $\pi_\theta(s,a)$를 포함하여 Gradient ascent를 한다. 알고리즘에 따라 Policy

gradient의 식이 다르지만, 본 논문에서는 액터-크리틱 알고리즘에 대한 Policy gradient 식을 고려한다.

액터(Actor)는 정책 $\pi_\theta(s,a)$에 들어갈 행동 a을 선택하며, 크리틱(Critic)은 액터에 의해 취해진 행동 a에 대해 Q 값을 평가한다. 크리틱을 통해 평가한 Q 값을 기반으로 에이전트의 정책에 대한 학습의 정도를 조절한다. 만약 해당 행동에 대해 높은 가치를 평가하였다면 에이전트의 정책이 해당 상태에서 그 행동을 더 많이 취할 수 있도록 하고, 그 반대의 경우 더 적게 취할 수 있도록 정책을 학습시킨다. 이러한 과정을 통해 정책 π와 가치에 해당하는 Q를 모두 동시에 학습하는 것이 액터-크리틱 알고리즘이다.

3.3 실험 결과

3.3.1 최종 보상

앞에서 서술했듯이, 강화학습에서 에이전트는 보상을 증가시키는 방향으로 학습한다. 따라서, 달성하고자 하는 목표를 수식적으로 표현한 보상 식에 따라서 에이전트의 행동에 따른 보상이 높다면 에이전트가 목표를 잘 달성했다고 말할 수 있다. 그림 2는 모든 에피소드에 대한 에이전트들의 누적 보상 합을 나타낸다. 그림 2를 보면, 모든 에이전트가 랜덤한 행동을 하는 무인 감시 시스템에서는 모든 에피소드에서 150에서 200의 보상 값을 얻었지만, CommNet 알고리즘을 이용해 학습한 에이전트들은 모든 에피소드에서 230에서 270의 보상 값을 얻은 것을 확인할 수 있다. 앞서 달성하고자 하는 목표를 보상으로써 수식적으로 표현한 것이기 때문에 높은 보상 값을 얻은 CommNet 알고리즘 기반의 에이전트들이 대조군에서의 에이전트들보다 공동의 목표를 잘 달성했다고 말할 수 있다.

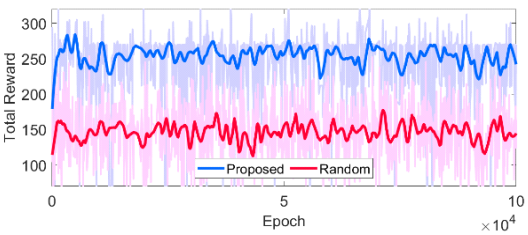


그림 2. 에피소드 진행에 따른 멀티 에이전트의 최종 보상
Fig. 2. Total multi-agent reward according to the progress of episodes

3.3.2 손실 함수

심층 강화학습에서 신경망, 즉 에이전트의 정책을 학습시킬 때 보상의 값만이 아니라, 손실 함수를 통해

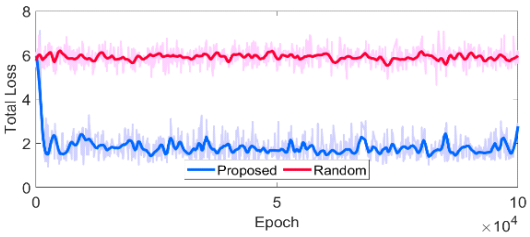


그림 3. 에피소드 진행에 따른 손실 함수의 값
Fig. 3. Value of loss function according to the progress of episodes

서 학습의 성능을 평가할 수 있다. 본 논문에서는 앞서 DQN과 Actor-Critic 학습 방법을 설명하면서 명시한 손실 함수 및 목적 함수를 사용한다. 따라서, 손실 함수를 최소화한다는 의미는 Q 값을 최대화하는 정책을 학습한다는

의미로 이어진다. 그림 3에서 CommNet 알고리즘 기반의 에이전트들이 대조군에서의 에이전트들보다 약 3배 낮은 손실 함수를 가지는 것을 확인할 수 있는데, 이를 근거로 대조군에서의 에이전트보다 CommNet 알고리즘 기반의 에이전트가 성능이 더 좋은 감시 시스템을 구축했다고 말할 수 있다.

3.3.3 감시 비율

본 논문에서는 감시 시스템 안에서 분포해 있는 전체 감시 대상의 수에 대해서 에이전트에 의해 감시를 당하는 대상의 수의 비율을 구하였고, 이를 통해 그림 4와 같이 에이전트들의 감시 비율을 구하였다. 더 많은 대상을 감시하는 감시 시스템일수록 그 성능이 좋다고 말할 수 있기 때문에 본 논문에서 제안한 보상 식에서 더 많은 대상을 감시할수록 높은 보상을 얻게끔 보상 식을 설계하였다. 그림 5를 보면, 모든 에피소드에서 CommNet 알고리즘 기반의 에이전트들은 약 55%에서 60%의 대상을 감지하는 반면, 대조군에서의 에이전트들은 약 35%에서 43%의 대상을 감지하

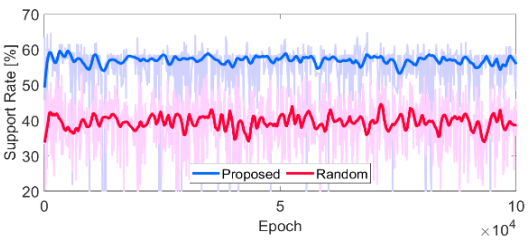


그림 4. 에피소드 진행에 따른 멀티 에이전트가 감시하는 대상의 비율
Fig. 4. Multi-agent surveillance rate according to the progress of episodes

는 것을 알 수 있다. 이를 통해, 대조군에서의 에이전트들보다 CommNet 알고리즘 기반의 에이전트들이 더 많은 수의 대상을 감시하는 것을 알 수 있다. 즉, CommNet 알고리즘 기반의 멀티 에이전트가 대조군에 비해 더 높은 성능의 감시 시스템을 구축하였다고 말할 수 있다.

3.3.4 겹친 정도

감시 비율과 더불어, 본 논문에서는 감시 시스템의 성능을 높이기 위해 에이전트가 감시하는 구역이 겹치지 않을수록 높은 보상을 받게끔 보상 식을 설계하였다. 이때 모든 에이전트가 감시하고 있는 구역의 넓이를 각 에이전트가 최대로 감시할 수 있는 구역의 넓이의 합으로 나누어, 멀티 에이전트가 감시하는 구역이 얼마나 겹쳐 있는지에 대한 정도를 계산하였다. 대조군에서의 에이전트들은 학습이 진행됨에 따라 전반적으로 약 40%에서 50%까지 감시하는 구역이 겹치지만, CommNet 알고리즘 기반의 에이전트들은 학습 초기를 제외하면 에이전트가 감시하는 구역이 20%에서 25% 정도만 겹치는 것을 확인할 수 있다. 따라서 CommNet 알고리즘 기반의 멀티 에이전트가 협력적으로 다른 에이전트의 감시 구역을 침범하지 않으며 감시 시스템을 구축하였다고 말할 수 있다. 학습 초기에도 대조군에 비해 겹친 정도가 낮은 것도 그림 5를 통해 확인할 수 있다.

또한 에피소드의 진행에 따라서 대조군에서의 감시 비율과 겹친 정도에 대한 분산이 상대적으로 큰 것을 확인할 수 있다. 즉, CommNet 알고리즘을 통해 멀티 에이전트가 제공하는 무인 감시 시스템의 안정성도 향상되었다고 그림 4, 5를 통해 말할 수 있다.

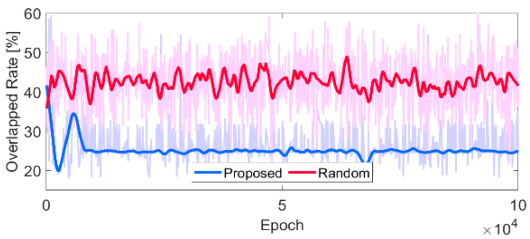


그림 5. 에피소드 진행에 따른 멀티 에이전트가 감시하는 구역의 겹친 정도
Fig. 5. Multi-agent overlapped rate according to the progress of episodes

Ⅳ. 결 론

본 논문은 강화학습에 대한 전반적인 내용을 서술하고, 단일 에이전트가 갖는 한계에 대해서 논하였다. 이후 심층 강화학습 기반의 멀티 에이전트가 CommNet 알고리즘을 이용한 통신을 통해 효율적인 자료 수집 및 감시 시스템을 구축하는 기법을 제안하였다. CommNet 알고리즘을 통해서 모든 에이전트는 자신이 관찰한 정보만이 아니라 다른 에이전트가 관찰한 정보도 고려하며 공동의 목표를 협력적으로 달성하도록 학습하였다. 이러한 방식을 통해 구축한 자료 수집 및 감시 시스템을 다양한 방법으로 평가하였고, 모든 측면에서 상대적으로 우수한 감시 시스템 성능을 보였다. 이를 통해 CommNet 알고리즘 기반의 심층 강화학습을 통해서 성공적으로 멀티 에이전트의 성능이 향상했음을 증명하였다.

References

[1] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yo-gamani, and P. Perez, “Deep reinforcement learning for autonomous driving: A survey,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 4909-4926, 2022. (https://doi.org/10.1109/TITS.2021.3054625)

[2] J. H. Lee, H. Seo, J. Park, M. Bennis, and Y. C. Ko, “Learning emergent random access protocol for LEO satellite networks,” *IEEE Trans. Wirel. Commun.*, pp. 1-1, Jul. 2022. (https://doi.org/10.1109/TWC.2022.3192365)

[3] C. Park, H. Lee, W. J. Yun, S. Jung, C. Cordeiro, and J. Kim, “Cooperative multi-agent deep reinforcement learning for reliable and energy-efficient mobile access via Multi-UAV control,” *arXiv preprint arXiv: 2210.00945*, 2022. (https://doi.org/10.48550/arXiv.2210.00945)

[4] W. J. Yun, M. Shin, D. Mohaisen, K. Lee, and J. Kim, “Hierarchical deep reinforcement learning-based propofol infusion assistant framework in anesthesia,” *IEEE Trans. Neural Netw. and Learn. Syst.*, pp. 1-12, Jul. 2022. (https://doi.org/10.1109/TNNLS.2022.3190379)

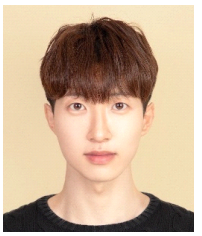
[5] S. Park, E. Han, S. Park, H. Jeong, and I. Yun, “Deep Q-network-based traffic signal control models,” *PLoS ONE*, vol. 16, no. 9, e0256405, 2021.

(https://doi.org/10.1371/journal.pone.0256405)
[6] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. Yo-gamani, and P. Perez, "Learning multiagent communication with backpropagation," in *Proc. NeurIPS*, vol. 29, Barcelona, Spain, 2016.
[7] R. Bellman, "Dynamic programming," *Science*, vol. 153, no. 3731, pp. 34-37, 1966. (https://doi.org/10.1126/science.153.3731.34)
[8] J. Kim, "Research trends on deep reinforcement learning," *Broadcasting and Media Mag.*, vol. 27, no. 2, pp. 26-34, 2022.
[9] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., "Mastering the game of go with deep neural networks and tree search," *Nature*, vol. 529, no. 7587, pp. 484-489, 2016. (https://doi.org/10.1038/nature16961)
[10] C. J. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3/4, pp. 279-292, May 1992. (https://doi.org/10.1007/BF00992698)
[11] V. Mnih, et al., "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013. (https://doi.org/10.48550/arXiv.1312.5602)
[12] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," in *Proc. NeurIPS*, vol. 12, 1999.
[13] S. M. Kakade, "A natural policy gradient," in *Proc. NeurIPS*, vol. 14, 2001.

박 찬 영 (Chanyoung Park)

2022년 8월 : 아주대학교 정보통신대학 전자공학과 졸업(공학사)
2022년 9월~현재 : 고려대학교 전기전자공학과 석박사통합과정
<관심분야> Reinforcement Learning, Machine Learning, Connected Mobility
[ORCID:0000-0002-7945-2362]

김 규 선 (Gyu Seon Kim)



2020년 2월~현재 : 인하대학교 항공우주공학과 학사과정
<관심분야> Reinforcement Learning, Aerial Mobility, Wireless Communication
[ORCID:0000-0002-5559-9749]

이 경 진 (Kyeongjin Lee)

2021년 8월 : 아주대학교 공과대학 교통시스템공학과 졸업(공학사)
2021년 9월~현재 : 아주대학교 공과대학 교통공학과 석박사 통합과정
<관심분야> 자율주행, 첨단교통체계, 교통운영, Deep Learning, 빅데이터 분석
[ORCID:0000-0002-6852-0534]

윤 일 수 (Ilsoo Yun)

1993년 2월 : 한양대학교 공과대학 도시공학과 졸업 (공학사)
1995년 2월 : 한양대학교 공과대학원 도시공학과 졸업 (공학석사)
2006년 1월 : University of Virginia 토목환경공학과 졸업(공학박사)
2009년 9월~2013년 8월 : 아주대학교 환경건설교통공학부 조교수
2013년 9월~2019년 9월 : 아주대학교 교통시스템공학과 부교수
2019년 9월~현재 : 아주대학교 교통시스템공학과 및 AI모빌리티공학과 교수
<관심분야> 자율주행 및 C-ITS, 모빌리티 서비스, 교통운영 및 안전
[ORCID:0000-0001-5618-7933]