유니티 시뮬레이션 환경 내 강화학습을 통한 무인기 자율운항

김경범 [1], 김도연 [1], 임태헌 [1], 최형석 [1], 황지웅 [1], 진우빈 [1], 이혁준 [1]
광운대학교 컴퓨터정보공학부 [1]

kyungbeom8@kw.ac.kr[1], dy0817@kw.ac.kr[1], icecoffee2500@kw.ac.kr[1], gudtjr5666@kw.ac.kr[1], chopmojji@kw.ac.kr[1], ubinjin2@naver.com[1], hlee@kw.ac.kr[1]

Autonomous operation of unmanned aerial vehicles through reinforcement learning in the Unity simulation environment

KyungBeom Kim[1], DoYeon Kim[1], TaeHeon Lim[1], HyungSeok Choi[1], JiUng Hwang[1], WooBeen Jin[1], HyukJoon Lee[1]
School of Computer and information Engineering Kwangwoon University[1]

요 약

본 논문은 유니티(Unity) 시뮬레이션 환경 내에서 드론의 자율비행에 대한 강화학습을 진행하며 드론이 유니티의 산악 환경 내에서 안정적으로 자율비행을 하는 것을 목표로 한다. 드론 비행에는 DDPG 알고리즘과 PPO 알고리즘을 적용하여 자율 비행할 수 있도록 하고 보상함수를 설정하여 다양한 상황에서 드론이 장애물 충돌 없이 목적지까지 비행할 수 있도록 한다. 자율 비행의 과정 및 결과를 직접 유니티 환경 내에서 확인하며 두 알고리즘을 텐서보드로 성능 비교 분석하였다.

Ⅰ. 서 론

드론은 소비자, 산업, 정부 및 군사 어플리케이션 등에 사용된다. 특수한 목적을 가진 임무용, 레저/완구용, 산업용 드론 등 다양한 분야에 드론의 활용 사례가 늘고 있다. 또한 국토의 70%가 산악환경인 우리나라에서도 산불의 감시와 진화에 드론을 투입하고 있으며 이외에도 조림, 벌채, 산지 관리, 훼손 감지 등 다양한 분야로 확대하고 있다.

현재 소형드론에 적용된 센서의 정밀도 및 오차범위가 크고 외부 환경적 요인에 영향을 많이 받고 있어 임무용 드론의 경우 전문 조종사의 투입이 필수적이다. 드론이 자율 비행할 수 있다면 전문 조종사의 인력 투입에 들어가는 자원을 아낄 수 있다.

최근 드론의 자율비행에 가장 보편적으로 사용되는 기법은 강화학습이다. 드론의 자율 비행은 비행 중 장애물 회피, 풍속 극복, 전파 간섭 등 여러 다양한 상황이 수시로 발생한다. 강화학습의 경우 학습 데이터를 직접 만들 필요가 없이 규칙과 보상 함수만으로 학습을 하기 때문에 다양한 변수가 존재하는 비행환경에 적합하다.

드론 자율 비행에 가장 많이 이용되는 대표적인 알고리즘 중에 해당 논문에서는 Deep Deterministic Policy Gradient (DDPG) 알고리즘[1]과 Proximal Policy Optimization (PPO) 알고리즘[2]을 선택하여 다양한 상황에서 드론이 장애물 충돌 없이 목적지까지 비행할 수 있도록 시뮬레이션을 진행하였고 위 두 알고리즘을 비교 분석하였다.

Ⅱ. 학습 알고리즘 및 환경 구성

Ⅱ.1 학습 알고리즘

DDPG 알고리즘은 Deterministic Policy Gradient (DPG) 알고리즘에 Deep Neural Network 기법을 적용한 Actor-Critic 기반 강화학습 알고리즘으로, 기존 Deep Q-Network (DQN) 알고리즘의 문제점인 이산적인 행동 환경에만 적용 가능한 점을 보완하여 연속적인 행동이 필요한 환경에 적용 가능하다.

PPO 알고리즘은 Policy Gradient (PG) 알고리즘에서 발생하는 정책의 과한 학습으로부터 일어나는 문제를 방지하고자 대리 함수를 이용한 범위 제한인 신뢰 구간을 만들어 안정적인 신경망의 학습을 가능하게 하였다. 기존 Trust Region Policy Optimization (TRPO) 알고리즘에서 복잡한 계산인 2차 미분 대신 1차 미분으로 근사화 시켜 보완한 알고리즘이다.

DDPG 알고리즘의 경우 탐험을 할 때 하나의 결정된 값에 무작위로 노이즈를 더한 행동 값을 리플레이 메모리에 저장해서 학습에 이용한다. 이 때 학습의 방향이 더해진 노이즈의 값에 크게 영향을 받는데, 노이즈는 무작위로 정해진 값이므로 학습의 점진적인 향상을 보장할 수 없다[3]. 또한 보상이 부족할 때 엑터 네트워크와 크리틱 네트워크가 모두 업데이트가 일어나지 않는 교착상태가 발생한다[4]. 반면 PPO 알고리즘의 정책 업데이트 방식은 이전의 모델과 현재 모델을 비교해서 더 나은 방향으로 학습이 진행되는 방식이기 때문에 학습 성능의 점진적인 향상을 보장하고, 이로 인해 보상이 점진적으로 증가하여 DDPG 알고리즘보다 교착상태가 덜 발생한다.

Ⅱ.2 시뮬레이션 학습 환경

시뮬레이션 환경으로는 산악 환경을 선정하였다. 그림 1 은 전체 시뮬레이션 환경 화면으로 나무, 바위와 같은 환경 요소를 이용하여 드론이 피해야 하는 정적 장애물을 배치하였고 새 떼와 같은 동적 장애물을 배치하였다. 두 종류의 장애물을 회피하여 출발지로부터 목적지까지 드론이 이동하는 환경을 구성하였다.

그림 2 은 그림 1 에서의 환경에서 동적 장애물 및 정적 장애물의 위치만 바꾼 9 개의 다른 환경이다. 이러한 9 개의 다른 환경에서 강화학습을 진행하였다. 드론의 경우 실제와 유사하게 하기 위해 속도는 최대 8m/s 로 설정했으며 초기 환경에서 목표의 높이는 약 180m, 드론과 목표지점 사이의 거리는 약 900m 로 설정했다. 위 환경은 유니티에서 제작한 환경으로 유니티 2022.1.21f1, 파이썬 3.8, 파이토치 1.13.0, 유니티 ML-Agent 2.3.0 버전에서 진행되었다.

그림 1. 시뮬레이터 학습 환경

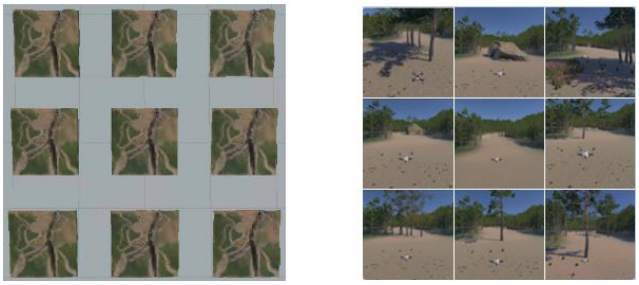

그림 2. 장애물이 배치된 9 개의 다른 환경

그림 3 은 본 연구에서 사용된 보상함수이다. 먼저, 드론이 목표 지점에 도달하면 보상을 주고 에피소드를 종료한다. 목표 지점과 너무 많이 멀어지는 경우, 학습 지역을 벗어나는 경우, 바닥으로부터의 거리가 너무 멀어지는 경우, 장애물을 감지하여 가깝다고 판단하는 경우 음의 보상을 주고 에피소드를 종료한다. 이외에는 이전 스텝에서 측정한 목표 지점과의 거리와 현재 스텝에서 측정한 목표 지점과의 거리의 차이만큼 보상을 누적한다. 거리가 가까워지면 양의 보상, 멀어지면 음의 보상이 누적되고 다음 스텝을 진행한다.

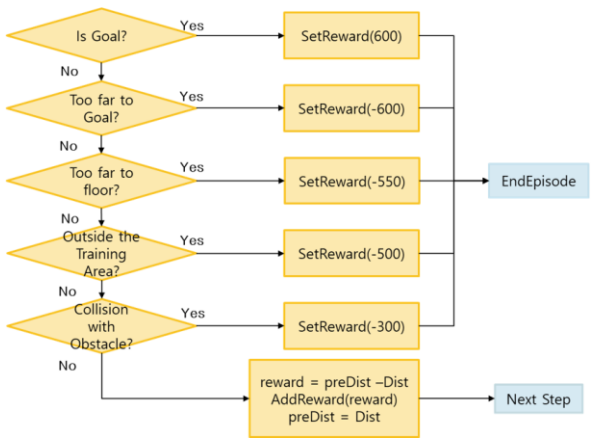

그림 3. 보상함수 설계

Ⅲ. 실험

그림 2 의 동일한 환경에서 DDPG 알고리즘을 적용하여 실험을 진행하였고 이어서 PPO 알고리즘을 적용하여 실험을 진행하였다. 알고리즘을 제외한 나머지 부분에서는 차이가 없게 하기 위해서 하이퍼파라미터 값을 동일하게 하였다. 그리고 동일하게 이천만 스텝으로 실험을 진행하였고 9 개 환경에 대한 평균 보상 값을 측정하였다.

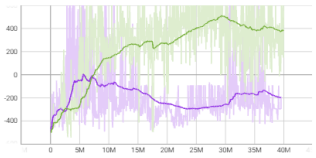

그림 4. 보상

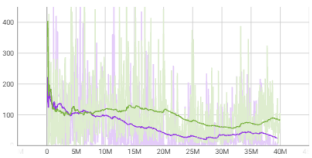

그림 5. 손실 값

두 그래프는 보상과 손실 값에 대한 결과 그래프이다. 그래프에서 x 축은 학습 스텝 수, y 축은 각각 평균 보상 값, 손실 값을 의미하며 초록색과 보라색은 각각 PPO 와 DDPG 알고리즘이다. 학습 초기에는 DDPG 가 PPO 에 비해 보상 값이 높고 손실 값이 낮은 것을 확인할 수 있다. 학습이 점차 진행됨에 따라 PPO 가 DDPG 보다 보상 값 이 높아지고 손실 값이 낮아진다. 이를 통해 PPO 알고리즘이 DDPG 알고리즘보다 학습 성능의 점진적이고 안정적인 향상을 알 수 있다.

Ⅳ. 결론

본 논문에서는 유니티로 제작된 산악 환경에서 강화학습이 적용된 드론이 여러 장애물을 회피하여 목표지점까지 도달하는 시나리오를 구성하여 학습을 진행하고 PPO 알고리즘과 DDPG 알고리즘을 비교 분석하였다. 간단한 환경이 아닌 복잡한 환경에서는 학습 성능의 점진적인 향상을 기대할 수 있는 PPO 알고리즘이 드론의 자율비행에 더 효과적인 학습 알고리즘이다. 향후 이러한 PPO 알고리즘을 이용한 멀티 에이전트 강화학습을 적용하여 여러 대의 드론이 편대를 이루어 목표지점에 도달하는 연구를 수행할 예정이다.

ACKNOWLEDGMENT

본 연구는 과학기술정통부 및 정보통신기술진흥센터의 SW 중심대학지원사업의 연구결과로 수행되었음. (2017-0-00096)

참 고 문 헌

[1] Lillicrap, Timothy P., et al. "Continuous control with deep reinforcement learning." *arXiv preprint arXiv:1509.02971* (2015).

[2] John Schulman, Fliip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov “Proximal Policy Optimization Algorithms” (2017).

[3] Shim, Wooil, Taehwa Park, and Kyungjoong Kim. "Comparison of Policy Optimization Reinforcement Learning for Simulated Autonomous Car Environment." *Korea Information Science Society* (2018): 833-835.

[4] Matheron, Guillaume, Nicolas Perrin, and Olivier Sigaud. "The problem with DDPG: understanding failures in deterministic environments with sparse rewards." *arXiv preprint arXiv:1911.11679* (2019).