https://doi.org/10.7840/kics.2023.48.12.1568

다중 스펙트럼 채널 접근을 위한 강화학습 알고리즘 연구

채 준 병◆, 박 종 인*, 최 계 원°

A Study on Reinforcement Learning Algorithm for Multi-Spectrum Channel Access

Jun-byung Chae◆, Jong-in Park*, Kae-won Choi°

요 약

본 논문은 802.11ax 기반 무선 통신 환경에서 다중 스펙트럼 채널 접근을 위해 다양한 강화학습 알고리즘을 적용하여 성능을 비교 분석한다. 본 논문에서는 강화학습 알고리즘을 통해 충돌을 방지하고 채널 자원을 최적으로 활용하는 채널 접근 기술을 제시한다. 이 기술의 적용은 통신속도의 향상과 미래 주파수 활용 기술 개발에 도움이 될 것이다.

키워드 : 강화학습, 802.11ax, 다중 스펙트럼 채널 접근, DDPG
Key Words : Reinforcement Learning, 802.11ax, Multi-Spectrum Channel Access, DDPG

ABSTRACT

This paper compares and analyzes performance by applying various reinforcement learning algorithms for multi-spectrum channel access in an 802.11ax-based wireless communication environment. In this paper, we present a channel access technology that uses reinforcement learning to prevent collisions and optimally utilizes channel resources. The application of this technology will help improve communication speed and develop future frequency utilization technology.

Ⅰ. 서 론

통신 기술은 빠른 속도로 발전하고 있으며, 통신 기기의 종류는 더욱 다양해지면서 수요가 증가하고 있다. 이러한 흐름은 무선 서비스 품질에 대한 요구사항을 더욱 높여놓고 있다.

이러한 상황 속에서 한정된 무선 채널 자원을 효율적으로 분배하는 것은 중요한 과제 중 하나이다. 다양한 통신기기와 사용자들로 인해 무선 채널은 급격히 혼잡해지고 있으며, 이로 인해 통신 간 충돌과 성능 저하의 문제가 대두되고 있다.

기존의 연구 중에서는 IEEE 802.11표준을 기반으로 한 무선 채널 액세스 메커니즘에 대한 강화학습 기반의 프레임워크가 제안되었다. 이 연구는 6G 무선 통신에서의 대규모 사물 인터넷 환경에서 최적의 자원 할당을 위해 무선 환경에서 수집된 데이터를 기반으로 실제로 측정된 채널 충돌 확률을 활용한다[1].

본 논문에서는 이러한 문제를 극복하고자 802.11ax

※ 본 연구는 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임. (2019-0-00964, 스펙트럼 챌린지를 통한 기존 무선국 보호 및 주파수 공유기술 개발)
◆ First Author : Sungkyunkwan University, Department of Electrical and Computer Engineering, chaejb88@skku.edu, 학생회원
° Corresponding Author : Sungkyunkwan University, Department of Electrical and Computer Engineering, kaewonchoi@skku.edu, 종신회원
* LG Electronics, pj5807@naver.com, 정회원
논문번호 : 202307-021-C-RE, Received July 31, 2023; Revised September 14, 2023; Accepted September 15, 2023

환경에서 다양한 강화학습 알고리즘을 적용하여 한정된 무선 채널 자원을 다수의 사용자들에게 효과적으로 분배하는 방법을 연구하였다. 802.11은 무선 로컬 영역 네트워크를 구현하기 위한 표준 규격으로, 무선 통신 환경에서 안정적이고 효율적인 데이터 전송을 위해 무선 채널을 관리하는 방법과 데이터 전송을 위한 물리적인 인터페이스를 정의하고 있다[2].

실험은 총 4개의 시나리오로 구성되었으며, 사용한 강화학습 알고리즘은 DQN, Actor-Critic, PPO, DDPG이다. 각 실험은 공유하는 다중 스펙트럼 채널, 최대 전송 패킷 길이, BSS들이 공유하는 채널, 그리고 충돌 보상 (reward)를 변경하며 진행하였다. 이러한 변화를 통해 각각의 알고리즘이 무선 채널 자원을 어떻게 활용하고, 효과적으로 분배하는지를 평가하였다.

Ⅱ. 본 론

2.1 802.11ax (Wi-Fi 6E)

무선 기기의 수요 증가와 함께 무선 서비스 품질에 대한 요구도 상승하였고, 이에 따라 무선랜은 현대 사회에서 필수적인 무선 네트워크 기술로 자리잡았다. IEEE에서 개발한 802.11 국제 표준 규격 이후의 진화 과정 중 802.11ax는 차세대 국제 표준 규격으로 등장하였다. 이 규격은 기존의 2.4GHz와 5GHz 주파수 대역을 포함하여 이전 규격과의 호환성을 유지하는 동시에 6GHz 대역에서 더 많은 채널을 추가한다. 이를 통해 밀집된 환경에서도 안정적이고 빠른 통신을 가능하게 한다.

IEEE 802.11ax는 이전 버전의 IEEE 802.11 표준 개정판과 비교하여 고밀도 배포 상황의 네트워크 성능과 사용자 경험에 초점을 맞췄다. 이를 위해 다중 사용자 매체접근제어(MU-MAC), 공간 재사용(SR), 대상 웨이크업 시간(TWT) 등의 매체접근제어 계층 기술을 도입되어 네트워크 접근 효율이 증가됨을 확인하였다.

본 연구에서는 이러한 802.11ax의 성능과 효율성을 분석하고 검증하기 위해 802.11ax 환경 시뮬레이터를 개발하였다.

또한, 다양한 강화학습 알고리즘을 적용하여 시뮬레이터의 검증 과정을 진행하였다.

2.2 다중 스펙트럼 채널 환경

본 연구에서는 802.11ax를 기반의 다중 채널 환경에서 실험을 진행하였다.

실험 구성을 위해 3개의 BSS(Basic Service Set)를 인접하게 배치하여 실험을 구성하였다. 이들 BSS는 각각 n개의 다중 스펙트럼 채널을 공유하고, 각 BSS는 하나의 AP와 8개의 Station으로 구성되어 있다. 각 BSS는 자체 채널을 설정하고 각자의 매체 접근제어 방식을 통해 통신을 진행한다. 그림 1의 BSS 1 같이 하나의 BSS에 강화학습을 수행하는 단일 에이전트를 연결할 수 있다.

각 AP는 그림 2에 표시된 것처럼, 자신의 BSS 내에 속한 임의의 Station에 전송할 패킷을 개별적으로 큐에 저장한다. 강화학습을 수행하는 에이전트는 Simulator Interface를 통해 BSS 내 AP가 얻은 채널 환경, 데이터 큐에 대한 정보를 observation으로 받는다.

이를 바탕으로 에이전트는 현재 통신 환경에 가장 적합한 매체 접근 제어 방식을 학습한다. 학습 후 현재 사용하기 적합한 채널, Station, 패킷길이를 결정하여 action을 한다. 강화학습 모델이 연결되지 않은 나머지 2개의 BSS는 Time Division Multiplexing Access (TDMA) 방식을 이용하여 시간의 겹침없이 데이터를 전송하도록 하였다. 강화학습 모델은 이러한 BSS들 간의 최적의 채널 공유 방식을 학습하도록 설계하였다.

본 연구에서 사용된 시뮬레이터는 그림 3에 나타나는 바와 같이 이벤트 발생에 따라 환경을 변화시키는 Discrete Event Simulation (DES) 형태로 구현되었다. 전체 시스템은 하나의 연속된 시간에서 동작하며, 시스템 내의 모든 AP와 Station은 독립적으로 작동한다. 또한, 에이전트의 매 action마다 설정된 시간에 따라 시뮬레이션 시간이 달라진다.

시뮬레이터의 모든 동작은 이벤트 기반으로 관리되며, 각 이벤트 처리를 위한 프로세스는 독립적으로 실행된다. 모든 이벤트에 의해 생성된 모든 데이터는 로그로

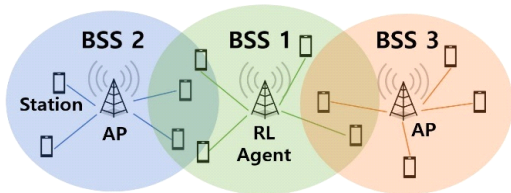


그림 1. 무선 통신 환경 및 BSS 구조
Fig. 1. Wireless communication environment and BSS structure

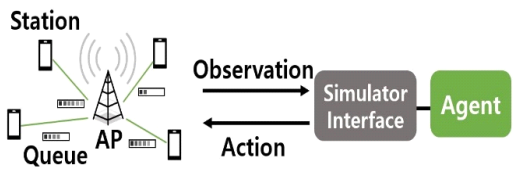


그림 2. 환경과 에이전트의 상호작용 과정
Fig. 2. Interaction between environment and agent

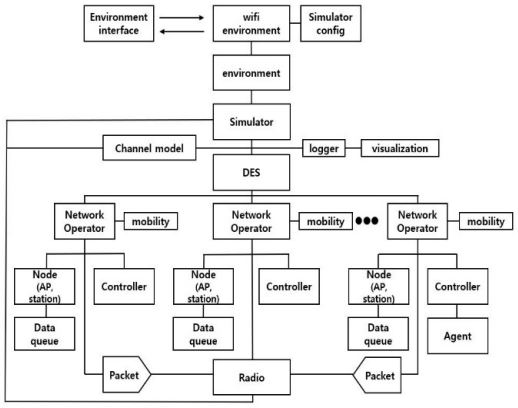


그림 3. 무선 통신 시스템
Fig. 3. Wireless communication system

기록되어, 이를 바탕으로 시뮬레이션의 진행 상황을 시각화하고 확인할 수 있도록 하였다.

본 연구에서 구현한 이러한 시뮬레이션 환경을 통해, 채널 자원을 최대한 효율적으로 활용할 수 있는 알고리즘을 제안한다. 이 알고리즘은 Wi-Fi 통신의 효율성과 안정성을 크게 향상시키는데 기여할 것이다.

2.3 강화학습 모델

2.3.1 강화학습

본 연구는 단일 에이전트를 활용한 강화학습 방식을 적용하였다. 강화학습에서 사용되는 데이터셋은 일련의 시간 순서에 따라 발생한 일련의 경험들로 구성된다. 에이전트는 주어진 환경과 상호작용하며 학습 단계에서 얻은 현재 상태(state), action, reward, 다음 상태를 하나의 Sample로 묶어 Replay buffer에 저장한 후, 평가 단계에서 이를 활용한다.

본 연구에서는 단일 에이전트가 AP와 상호작용하면서 학습 과정을 이루어지게 한다. 초기 단계에서 에이전트는 결정된 action을 AP로 전달하며, 이에 대한 응답으로 AP는 해당 action을 한다. action이 완료된 후 AP가 현재 채널 상태의 일부 정보인 관찰 (observation)을 에이전트에게 반환한다.

observation을 바탕으로 에이전트는 reward를 획득하며, reward를 최대화하는 방향으로 학습 파라미터를 업데이트 하게 된다. 이런 식으로, 에이전트는 환경과 상호작용하며 지속적으로 자신의action을 최적화하는 학습 과정을 수행한다.

(1) Action

에이전트가 주어진 상태에서 취하는 action을 나타내며, 에이전트는 action을 선택함으로써 환경에 영향을 미치게 된다.

강화학습 에이전트가 수행할 수 있는 action 유형을 두 가지로 구분하였다.

첫째, Sensing은 모든 채널에서 수신된 신호를 확인하여 각 채널의 사용 여부를 파악하는 action이다.

둘째, Transmit은 전체 채널 중에서 사용할 채널, Station, 그리고 패킷 전송 길이를 결정하는 action이다.

각 시간 slot t 에서의 action은 a_t로 표현되며, 패킷 전송 길이는 $P_L \in \{1, 2, \cdots, n\}$ 중에서 하나의 값을 선택한다. $P_L = 1$인 경우 1개의 패킷을, $P_L = 4$인 경우 4개의 패킷을 연속으로 전송한다.

사용할 채널은 $a_t^{ch} \in \{0,1, \ldots, \mathrm{n}-1\}$ 중에서 중복 선택이 가능하도록 설정하였다.

선택한 채널에 연결될 Station은 $a_t^{sta} \in \{0,1, \ldots, \mathrm{n}-1\}$ 중에서 채널당 하나의 Station을 결정해야 하며, 본 연구에서는 8개의 Station을 사용하였다.

따라서 본 연구에서의 action은 다음과 같이 표현된다.

$$Action = \begin{cases} sensing & ,if\ \ a_t = sensing, \\ (a_t^{ch},\ a_t^{sta}, P_L) & ,if\ \ a_t = transmit. \end{cases} \quad (1)$$

(2) Observation

각 시간 단계에서의 환경의 상태를 나타내며, 상태는 에이전트가 결정을 내리는 기반이 되는 정보를 포함한다. 본 연구에서는 상태를 사용하지 않고 observation을 활용하여 환경의 정보를 수집하였다. 이러한 observation은 action에 따른 채널 정보, 데이터 트래픽 처리를 위한 큐의 정보로 구성된다. observation은 수행하는 action의 종류에 따라 다르게 수집된다.

Sensing에 대한 observation은 감지된 채널 정보 $o_t^S \in \{1,2,\ldots,\mathrm{n}\}$를 포함하며, 모든 채널의 신호 세기를 측정하여 일정 임계치를 초과한 경우 해당 채널을 사용중인 것으로 간주한다.

Transmit에 대한 observation은 전송이 성공한 채널 정보 $o_t^T \in \{1,2,\cdots,\mathrm{n}\}$를 수집한다.

AP가 ACK를 성공적으로 수신하면 해당 패킷의 전송이 성공했다고 판단한다. 패킷 성공은 제한 시간 내에 성공적으로 전송된 패킷의 수를 나타내며, 패킷 지연은 패킷이 전송되었지만 제한 시간이 지나 지연된 패킷의 수를 의미한다.

큐의 정보는 큐의 길이 Q_t와 HOL(Head of the Line) delay D_t로 구성된다. AP는 유한한 길이의 데이터 패킷 큐를 가지고 있으며 시간 slot t에서 큐에 있는 패킷의 수는 Q_t로 표시된다. 또한, Q_{max}는 큐에 저장할 수 있는 최대 패킷의 수를 나타내며, $Q_t \leq Q_{max}$가 된다. 시간 slot (t-1)에서 큐에 도착하는 패킷 수는 A_t로 표시된다. 큐의 패킷 수는 식 (2)와 같이 시간 slot에 따라 변화한다.

$$Q_t = min(Q_{t-1} - P_L + A_t, Q_{max}) \quad (2)$$

큐가 가득 찬 상황에서는 도착한 일부 패킷이 손실될 수 있다. 시간 slot t에서 손실된 패킷의 수는 L_t로 표현되며, 이는 다음과 같은 식으로 표현된다.

$$L_t = [Q_{t-1} - P_L + A_t - Q_{max}]^+ \quad (3)$$

여기서 $[x]^+$는 $max(x, 0)$를 의미하며 이는 x의 값이 0보다 큰 경우 x를, 그렇지 않은 경우 0을 반환하는 함수이다.

HOL delay는 큐에 도착한 패킷들 중에서 가장 먼저 도착한 패킷이 다음 처리 단계로 전달되기까지 대기해야 하는 시간이다. 예를 들어, 패킷 큐에 패킷이 들어간 시점(enqueue) 부터 해당 패킷이 큐에서 나오는(dequeue) 시점까지의 시간을 HOL delay라고 정의한다. 본 연구에서는 모든 Station의 HOL delay를 수집하였다. 패킷 큐는 네트워크에서 송수신되는 패킷들이 대기하는 공간을 의미한다.

따라서 본 연구에서의 observation은 다음과 같이 표현된다.

$$Observation = \begin{cases} (o_t^S, Q_t, D_t), \ if \ \ a_{t-1} = sensing, \\ (o_t^T, Q_t, D_t), \ if \ \ a_{t-1} = transmit. \end{cases} \quad (4)$$

(3) Reward

에이전트가 어떤 상태에서 어떤 행동을 했을 때 받는 reward를 나타낸다. 이는 에이전트가 학습하는 기준이 되며, 목적은 누적 보상을 최대화하는 것이다.

본 연구에서는 각 에피소드 에서의 reward를 패킷의 성공, 지연, 손실, 그리고 충돌의 네 가지 지표의 가중합으로 계산하였다.

Sensing 작업은 채널 감지 과정을 포함하고 있지만 데이터 전송은 이루어지지 않으므로, 해당 과정의 reward는 0으로 설정하였다. 반면에, Transmit 작업에서 통신이 성공적으로 이루어진 경우 reward는 1로 설정하였다. 패킷의 지연 및 손실에 대한 reward는 0으로 설정하였다. 충돌이 발생한 경우의 reward는 -n으로 설정하였다.

이때, 전송에 성공한 채널의 개수를 SC(Success Channels), 충돌이 발생한 채널의 개수를 CC(Collision Channels)로 정의하였다. 이렇게 정의된 SC와 CC를 활용하여 각 단계에서의 reward를 계산하는 식은 다음과 같다.

$$Reward = \begin{cases} 0 & , if \ \ a_{t-1} = sensing, \\ SC - \mathrm{n} \times CC & , if \ \ a_{t-1} = transmit. \end{cases} \quad (5)$$

2.3.2 학습 알고리즘

본 연구에서는 다양한 강화학습 알고리즘인 DQN, Actor-Critic, PPO, 그리고 DDPG를 적용하였다.

(1) DQN (Deep Q Network)

DQN은 기존 Q-Learning 알고리즘에 심층학습 기술을 결합하여 강화학습 알고리즘의 성능을 향상시킨 방법이다[3]. 기존 Q-Learning은 Q-table을 사용하여 state-action쌍의 가치를 추정하였으나, 이는 state공간이나 action공간이 클 경우 계산적으로 비효율적이었다. DQN은 이러한 한계를 극복하기 위해, Q-table 대신에 Q 함수를 근사한 심층 신경망을 사용하였다.

DQN은 강화학습에서 핵심 문제 중 하나인 action 선택의 문제를 해결하기 위해 제안되었다. 상태에 대한 입력을 받아, 각 action에 대한 가치를 추정하는 심층 신경망을 사용한다. 이 신경망은 Q함수를 근사하며, 학습된 신경망을 통해 에이전트는 현재 상태에서 가장 가치 있는 action을 선택하게 된다. 상태가 신경망의 입력값으로 주어지면 상태에서 가능한 모든 action에 대한 reward의 예측값 Q를 계산한다.

DQN은 이산적인 action공간에서 뛰어난 성능을 보이지만, 연속적인 action공간에서는 적용이 어렵다. 또한, 고차원의 상태 공간에서는 수렴이 어려울 수 있다.

(2) Actor-Critic

Actor-Critic은 강화학습 알고리즘의 효율성을 향상시키기 위한 중요한 방법론 중 하나로, 작은 편향을 가진 REINFORCE 알고리즘과 작은 분산을 가진 DQN 알고리즘의 장점을 결합한다. 이는 최적의 정책을 보다 효율적으로 찾는 방법을 제공한다.

본 방법의 핵심은 두개의 서로 다른 신경망을 사용하는 것이다. 하나는 정책 (policy)을 담당하는 Actor 신경망이고, 나머지 하나는 가치 (value)를 평가하는 Critic

신경망이다. 이 두 신경망은 각각 별도의 역할을 수행하며, 서로 상호작용하면서 학습과 action을 개선한다. 이로써 안정적이고 빠른 학습이 가능하게 된다. 그러나, 두 신경망을 효과적으로 조합하기 위해서는 구현이 다소 복잡할 수 있다.

(3) PPO (Proximal Policyh Optimization)

PPO는 Actor-Critic 구조를 기반으로 하고 있으며, PPO의 핵심 아이디어는 새로운 정책 업데이트 시에 이전 정책과의 차이가 일정 범위 안에 있도록 제한하는 것이다. 이는 정책 업데이트 시에 큰 변화를 막기 위해 클리핑 (Clipping) 방법을 사용한다[4].

PPO는 파라미터 업데이트 과정을 전체 샘플에 대해 일괄적으로 수행하는 것이 아닌, 여러 epoch를 거쳐 점진적으로 진행하는 방법을 채택하였다. 이는 샘플을 한 번에 처리하는 과정에서 발생할 수 있는 잠재적인 문제를 완화하며, 이를 통해 학습의 안정성을 높이는 데 중요한 역할을 한다. 또한, PPO는 다양한 환경에서 뛰어난 성능을 보이며 고차원의 상태 공간에서도 안정적으로 작동할 수 있다.

이러한 PPO의 접근 방식은 학습 과정의 효율성과 안정성을 동시에 증진시키는데 크게 기여하지만, 대량의 샘플을 사용하여 정책을 업데이트하므로 학습 시간이 오래 걸린다.

(4) DDPG (Deep Deterministic Policy Gradient)

DDPG는 연속적인 action공간 에서의 강화학습 문제를 효과적으로 해결하기 위해 개발된 알고리즘으로, Actor-Critic 구조를 기본으로 활용한다. 이 알고리즘은 DQN과 DPG (Deterministic Policy Gradient)의 원리를 결합하여 연속적인 action을 수행할 수 있는 에이전트를 학습하는데 효과적이다[5]. 그러나, action공간이 매우 큰 경우 학습이 어려울 수 있다.

DDPG 알고리즘은 연속적인 action공간을 목표로 하고 있다. 하지만, 본 연구는 이산적인 action공간이므로 최적의 정책을 도출하는 데에는 제한점이 있다. 이러한 한계를 극복하기 위해 본 연구에서는 노이즈를 사용하지 않고 두 가지 전략을 적용하였다.

첫째, Straight-Through Gradients with Automatic Differentiation (STG) 알고리즘을 적용하였다[6]. 이 알고리즘은 각 action에 대한 점수 (logits)를 softmax 확률 분포로 변환하고, 이 확률 분포에 따라 action을 샘플링한 후, 이 샘플링된 action에 대해 그래디언트를 계산하는 과정을 포함한다. 이 과정에서 중요한 특징은 Straight-Through Estimator를 사용하여 one hot 인코딩된 샘플링 action에 대한 그래디언트를 계산할 수 있다는 점이다. 이렇게 하면 역전파 과정에서 그래디언트를 보존하며 이산적인 선택에도 불구하고 신경망의 가중치를 업데이트하는 데 필요한 그래디언트를 제공할 수 있다. 이러한 특성으로 인해 STG 알고리즘은 이산적인 acction공간에서도 강화학습 모델을 안정적으로 학습시키는 데 도움이 된다.

알고리즘 1. Straight-Through Gradients with Automatic
Algorithm 1. Straight-Through Gradients with Automatic.

Algorithm 1 : Straight-Through Gradients with Automatic Differentiation

```
sample = ont_hot(draw(logits))
probs  = softmax(logits)
sample = sample + probs - stop_grad(probs)
```

둘째, Gumble-softmax 기술을 적용하였다. 이 기술은 DDPG 알고리즘에서 확률 분포를 생성하는 과정에서 생기는 문제를 해결하는 방법으로, 이산적인 action 공간에서도 DDPG 알고리즘을 적용 가능하게 한다[7].

Gumble-softmax 기술은 클래스 확률의 범주형 분포에서 샘플을 추출하는 간단하고 효율적인 방법을 제공한다. 이러한 기술은 특히 이산적인 샘플링 과정에서 발생하는 미분 가능성의 문제를 해결하기 위해 개발되었다. 이를 통해 이산적인 action을 결정하는 신경망의 학습이 가능해진다. 이 기술의 도입으로 확률적인 action선택 과정이 신경망의 역전파를 통한 학습에 잘 통합될 수 있게 되었다. 따라서, 이러한 Gumble-softmax 기술의 도입으로 인해 이산적인 action공간에서도 DDPG 알고리즘을 효과적으로 적용할 수 있게 되었다.

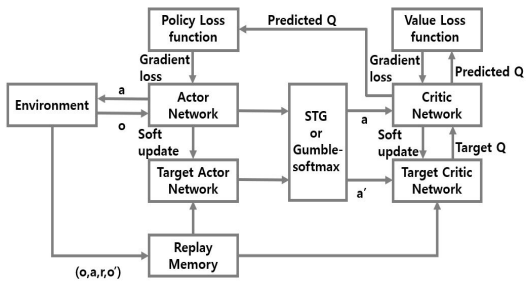


그림 4. DDPG 학습 과정
Fig. 4. DDPG learning process

Ⅲ. 실 험

3.1 학습 매개변수

본 연구에서는 실험을 진행하기에 앞서 다양한 하이퍼파라미터를 설정하였으며, 그 값들은 표 1에 정리하였다.

또한, 실험에는 4가지 시나리오 환경이 사용되었다. 공유하는 다중 스펙트럼 채널, 최대 전송 패킷 길이, BSS들이 공유하는 채널, 그리고 충돌 reward를 변경하며 진행하였으며, 이러한 환경들과 그에 해당하는 값들은 표 2에 정리하였다. 이렇게 다양한 시나리오를 통해 각 알고리즘들이 서로 다른 환경에서 어떻게 동작하는지를 평가하였다.

예로들어, 실험 1에서는 강화학습을 수행하는 단일 에이전트가 채널 0,1,2,3을 공유하고 있으며, 최대 패킷 전송길이 $P_L = 2$로 Transmit을 할 수 있다. 또한, 강화학습 모델이 연결되지 않은 BSS 2는 0,1번 채널을 공유하고, BSS 3은 2,3번 채널을 공유하고 있다. 이때 충돌이 발생하면 -2의 reward를 얻게 된다.

학습 과정은 지속적으로 모니터링되었으며, 강화학습 에이전트의 학습이 완료될 때마다 성능 평가를 위한 평과 과정을 수행하였다. 성능 평가는 다양한 지표를 고려하여 진행되었다. 이에는 전체 채널 대비 성공률, reward, 패킷의 평균 지연, 패킷의 평균 손실, 그리고 평균 충돌 이 포함되었다. 이러한 지표들은 강화학습 에이전트의 성능을 종합적으로 평가하고, 그 효과를 측정하는 데 있어 중요한 역할을 하였다. 이를 통해 실험 결과의 신뢰성과 유효성을 확인하고, 제안된 강화학습 알고리즘의 성능을 적절하게 평가하였다.

표 1. 학습 매개변수
Table 1. Training parameter

Parameter	Value
Number of Station	8
Learning rate	0.0000001
Discount factor	0.99
Episode	200
PPO clip rate	0.2
PPO epoch	10

표 2. 실험 시나리오
Table 2. Experiment scenario

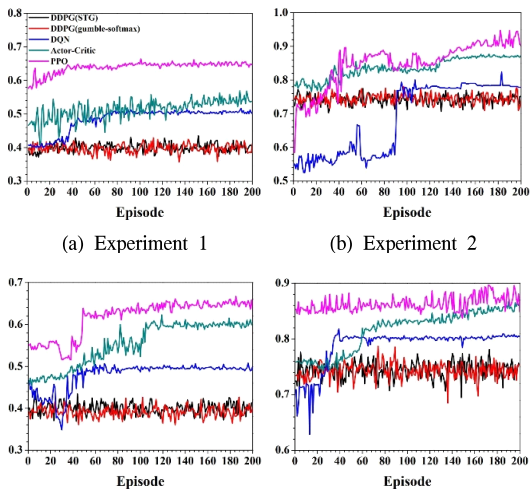
Experiment	실험1	실험2	실험3	실험4
Frequency channel	0, 1, 2, 3	0, 1, 2, 3	0, 1, 2, 3, 4, 5	0, 1, 2, 3, 4, 5
Max number packet	2	2	4	4
BSS2 frequency channel	0, 1	0, 1	0, 1, 2	0, 1, 2
BSS3 frequency channel	2, 3	2, 3	3, 4, 5	3, 4, 5
Collision Reward	-2	-1	-2	-1

3.2 학습 및 실험 결과

본 연구에서는 다양한 강화학습 알고리즘의 성능을 비교 분석하기 위해 4가지 시나리오 환경에서 여러 알고리즘을 동일한 학습 매개변수를 사용하여 실험을 수행하였다. 실험은 총 200회의 에피소드가 종료된 시점의 성능 지표를 기준으로 각 알고리즘의 성능을 평가하였다.

그림 5과 그림 6은 전체 채널 대비 성공률과 패킷의 평균 지연을 시각화한 것으로, 각 알고리즘의 성능을 직관적으로 이해할 수 있다. 충돌 reward 변화에 의한 성공률의 변화는 미비하지만, 공유하는 다중 스펙트럼 채널수를 증가시키면 성공률이 증가하였다. 이 결과는 공유 스펙트럼 자원을 더욱 효율적으로 활용하면서 성능 향상을 이끌어낼 수 있음을 시사한다.

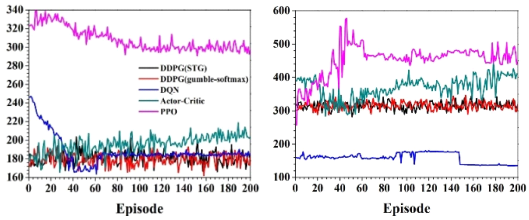
그림 7은 reward를 시각화한 것으로, 충돌 reward가 -2에서 -1로 변했을 때 reward의 변화가 소폭 상승하였다. 이는 충돌 reward가 더 작은 값으로 변하면 에이전트가 충돌을 피하는 방향으로 보다 더 효과적으로 학습함을 의미한다.



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

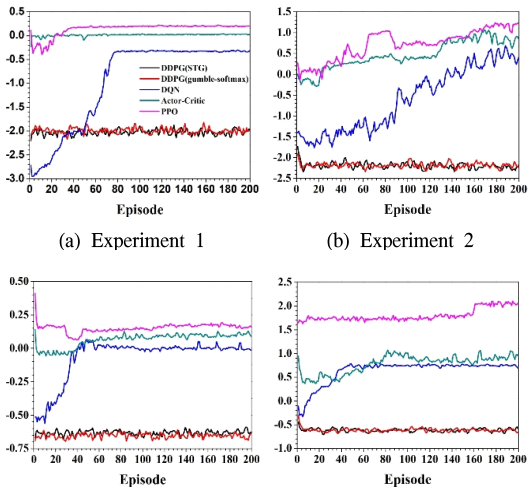
그림 5. 전체 채널 대비 전송 성공 비교
Fig. 5. Comparison of transfer success versus total channel



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

그림 6. 패킷의 평균 지연 비교
Fig. 6. Average Delay Comparison of Packets



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

그림 7. Reward 비교
Fig. 7. Reward Comparison

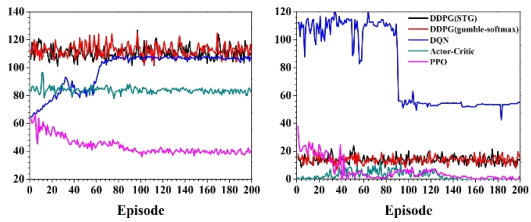
또한, 학습에는 데이터 트래픽 처리를 위한 큐의 정보가 중요하게 작용하였다. 큐의 길이 Q_l가 길거나 HOL delay D_l가 긴 큐를 선택하여 Transmit한 경우 전체 채널 대비 성공률과 reward가 증가하였다.

이러한 결과들은 충돌 reward의 조절과 데이터 트래픽 처리에 따른 큐의 영향이 강화학습 알고리즘의 성능에 영향을 미침을 의미한다. 따라서 이러한 요소들을 고려하여 알고리즘의 성능을 최적화하는 것이 중요하며, 이로 인해 에이전트가 환경과 더 잘 상호작용하여 보다 안정적인 학습 결과를 얻을 수 있다.

그림 8와 그림 9는 패킷의 평균 손실과 평균 충돌을 그래프로 나타내었다.

먼저, 그림 8를 살펴보면 패킷의 평균 손실이 감소할수록 에이전트가 전송하는 패킷의 손실이 줄어들어서 통신의 효율성이 증가하는 것으로 확인하였다. 이는 패킷 손실이 감소하면 데이터 전송이 더 원활하게 이루어져 네트워크 성능이 향상될 수 있다는 것을 의미한다.

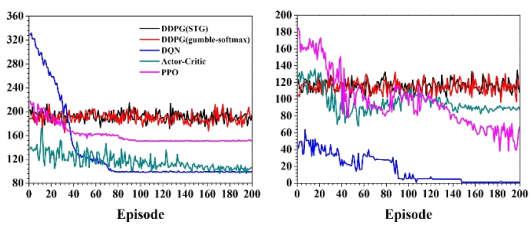
또한, 그림 9에서는 평균 충돌이 감소하면 에이전트 간의 충돌이 줄어들어서 네트워크의 효율성과 안정성이 향상되는 것으로 해석할 수 있다. 충돌이 감소하면



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

그림 8. 패킷의 평균 손실 비교
Fig. 8. Average Drop Comparison of Packets



(a) Experiment 1 (b) Experiment 2

(c) Experiment 3 (d) Experiment 4

그림 9. 패킷의 평균 충돌 비교
Fig. 9. Average Collision Comparison of Packets

전송 시 충돌로 인한 데이터 손실이 줄어들어, 더 많은 패킷이 성공적으로 전송된다.

그림 5, 그림 8, 그리고 그림 9의 모든 실험의 결과를 분석한 결과 DDPG 알고리즘은 패킷의 평균 손실과 평균 충돌이 미미하게 변할 경우에 전체 채널 대비 전송 성공률의 변화가 미미했다. 그러나, 실험2의 경우 PPO, Actor-Critic, DQN 알고리즘은 패킷의 평균 손실과 평균 충돌이 감소할 때 전체 채널 대비 전송 성공률이 상당히 증가하는 것을 확인하였다. 이는 PPO, Actor-Critic, DQN 알고리즘의 에이전트가 자원을 더 효율적으로 활용하고 충돌을 피하며 데이터를 전송한다는 의미이다.

이러한 결과를 종합적으로 고려할 때, 패킷의 평균 손실과 평균 충돌이 감소하는 경향을 보이는 경우에는 전체 채널 대비 전송 성공이 증가함을 확인하였다. 이는 제안된 강화학습 알고리즘이 네트워크 통신의 성능을 개선하는데 효과적으로 기여하였음을 시사한다.

실험 결과, PPO 알고리즘은 다른 알고리즘들에 비해 높은 성공률과 높은 보상을 얻는 것으로 나타났다. PPO, Actor-Critic, DQN 알고리즘은 시간이 지남에 따라 성공률과 평균 보상이 증가하는 경향을 보였는데, 이는 에이전트가 학습 과정을 거치며 성능을 향상시키고 있음을 시사한다.

그러나, DDPG 알고리즘은 Episode의 진행과 무관하게 에이전트의 학습이 진전되지 않는 것으로 확인되었다. 이는 DDPG 알고리즘이 이산적인 action공간에서는 최적의 성능을 발휘하지 못함을 보여준다. 이로 인해 DDPG 알고리즘이 본 연구에서 적용된 이산적인 환경에 적합하지 않다는 점을 확인하였다.

실험 결과를 통해 환경의 성격은 강화학습의 핵심 요소로 작용함을 확인하였다. 이러한 환경의 성격은 에이전트가 어떤 행동을 선택하고 보상을 받게 될지에 큰 영향을 미친다. 따라서, 강화학습에서는 환경의 성격을 고려하여 적절한 알고리즘을 선택하고 학습 파라미터를 조정하는 것이 중요하다.

이러한 결과들은 다양한 강화학습 알고리즘을 비교하여 어떤 상황에서 어떤 알고리즘이 더 적합하고 효과적인지를 파악하는데 도움을 줄 것으로 기대된다.

Ⅳ. 결 론

본 연구에서는 802.11ax 기반 무선 통신 환경에서 강화학습 알고리즘을 적용하여 채널 접근 방식에 대한 깊이 있는 연구를 수행하였다. 실험 결과를 통해, 총 에피소드 수가 증가할수록 강화학습 에이전트는 충돌 없이 채널에 접근하는 능력을 향상시켰으며, 이를 통해 채널 상태를 학습하고 자원을 효율적으로 활용하였다. 더불어, 에이전트는 전송 과정에서의 충돌을 사전에 예측하고 이를 회피하는 능력 또한 개선하였다. 그러나, DDPG 알고리즘을 적용한 경우, 환경 변화에 따른 학습 결과의 변동성을 확인하였으며, 이에 따라 실험 결과가 초기 예상과는 다르게 나타나는 경우가 있었다.

본 연구의 결과는 802.11ax 기반 무선 통신 환경에서 통신 속도를 개선하고, 미래의 주파수 활용 기술 개발에 중요한 기반을 마련하는데 기여할 것으로 기대된다. 강화학습 모델을 더욱 발전시키는 것은 무선 통신 기술의 안정화와 연구에서는 다중 에이전트 환경에서의 성능 향상에 대해 추가적인 연구를 진행할 계획이다.

References

[1] Rashid Ali, et al., “Reinforcement-learning-enabled massive internet of things for 6g wireless communications,” *IEEE*, Jun. 2021. (https://doi.org/10.1109/MCOMSTD.001.2000055)

[2] IEEE 802.11, “*Wireless LAN Medium Access Control and Physical Layer specifications*,” IEEE, Sep. 1999. (https://doi.org/10.1109/IEEESTD.2007.373646)

[3] V. Mnih, et al., “Playing atari with deep reinforcement learning,” *Google DeepMind*, 2016. (https://doi.org/10.48550/arXiv.1312.5602)

[4] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017. (https://doi.org/10.48550/arXiv.1707.06347)

[5] T. P. Lillicrap, et al., “Continuous control with deep reinforcement learning,” *arXiv preprint arXiv:1509.02971*, 2015. (https://doi.org/10.48550/arXiv.1509.02971)

[6] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba, “Mastering atari with discrete world models,” *arXiv preprint arXiv:2010.02193*, 2020. (https://doi.org/10.48550/arXiv.2010.02193)

[7] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with gumbel-Softmax,”

arXiv preprint arXiv:1611.01144, 2016.
(https://doi.org/10.48550/arXiv.1611.01144)

채 준 병 (Jun-byung Chae)

2014년 2월 : 인하대학교 컴퓨터정보공학과 학사 졸업
2022년 3월~현재 : 성균관대학교 전자전기컴퓨터공학과 석사과정
<관심분야> 무선통신, 강화학습, 머신러닝
[ORCID:0009-0002-6237-668X]

박 종 인 (Jong-in Park)

2020년 8월 : 한국외국어대학교 전자공학과 학사 졸업
2023년 2월 : 성균관대학교 전자전기컴퓨터공학과 석사 졸업
2023년 3월~현재 : LG전자 연구원
<관심분야> 웹엔진 개발, 강화학습, 머신러닝
[ORCID:0000-0002-4035-6728]

최 계 원 (Kae-won Choi)

2007년 8월 : 서울대학교 전기컴퓨터공학부 박사
2010년 9월~2016년 8월 : 서울과학기술대학교 컴퓨터공학과 조교수
2016년 9월~현재 : 성균관대학교 전자전기컴퓨터공학과 부교수
<관심분야> 무선통신, 머신러닝, WPT
[ORCID:0000-0002-3680-1403]