멀티에이전트 강화학습을 위한 통신 기술 동향

Survey on Communication Algorithms for Multiagent Reinforcement Learning

서승우 (S.W. Seo, seosw@etri.re.kr) 정보전략부 Post-Doc.
신영환 (Y.H. Shin, shinyh1115@etri.re.kr) 정보전략부 Post-Doc.
유병현 (B.H. Yoo, bhyoo@etri.re.kr) 복합지능연구실 선임연구원
김현우 (H.W. Kim, kimhw@etri.re.kr) 복합지능연구실 책임연구원
송화전 (H.J. Song, songhj@etri.re.kr) 복합지능연구실 책임연구원/실장
이성원 (S. Yi, sungyi@etri.re.kr) 정보전략부 책임연구원/부장

ABSTRACT

Communication for multiagent reinforcement learning (MARL) has emerged to promote understanding of an entire environment. Through communication for MARL, agents can cooperate by choosing the best action considering not only their surrounding environment but also the entire environment and other agents. Hence, MARL with communication may outperform conventional MARL. Many communication algorithms have been proposed to support MARL, but current analyses remain insufficient. This paper presents existing communication algorithms for MARL according to various criteria such as communication methods, contents, and restrictions. In addition, we consider several experimental environments that are primarily used to demonstrate the MARL performance enhanced by communication.

KEYWORDS MARL 실험 환경, 멀티에이전트 강화학습(MARL), 에이전트 간 통신, 에이전트 간 통신 제한 사항, 통신 알고리즘

Ⅰ. 서론

멀티에이전트 강화학습(MARL: Multi-Agent Reinforcement Learning)은 주어진 환경에서 다수의 에이전트가 서로 협력 또는 경쟁하여 보상을 최대화하는 행동을 학습하는 강화학습 방법의 하나이다. 다수의 에이전트가 운용되는 자율주행, 센서 네트워크, 로봇과 같은 환경에서는 운용되는 기기 간의 특성이 다양한 경우가 일반적이다. 하나의 에이전트만을 최적화하는 기존 싱글에이전트 강화학습은 다양한 특성을 갖는 다수의 기기를 최적화하는 데 적합하지 않다. 그에 반해, 다수의 에이전트에 대한 최적화를 수행하는 멀티에이전트 강화학습은 해당 기기들의 개별 특성을 고려하여 전체 성능을 극대화하는 것이 가능하다.

하나의 에이전트가 모든 환경을 관리하는 것이

* DOI: https://doi.org/10.22648/ETRI.2023.J.380410
* 본 연구는 한국전자통신연구원 내부연구과제의 일환으로 수행되었음[멀티에이전트 강화학습 탐색, 통신, 학습전략 기술 연구, 22YE1210, 자율성장형 복합인공지능 원천기술 연구, 23ZS1100].

아닌 다수의 에이전트를 고려하는 멀티에이전트 강화학습의 특성상, 에이전트는 현재 관측하고 있는 자신의 정보만을 바탕으로 최선의 행동을 수행하게 된다. 개별 에이전트의 관점에서는 자신이 현재 선택한 행동이 최선일지라도, 전체 에이전트의 관점에서는 해당 행동이 전체 보상을 낮추는 행동일 수 있다. 따라서 에이전트는 정보 공유를 통해 전체 상황을 파악하고, 전체 보상을 높이는 행동을 수행해야 한다.

전체 에이전트의 정보를 효율적으로 취득하기 위한 방안으로 멀티에이전트 강화학습을 위한 통신 알고리즘들이 제안되었다[1]. 멀티에이전트 강화학습을 위한 통신 알고리즘은 훈련(Training) 및 실행(Execution) 과정에서 에이전트들이 습득한 자신의 부분 관측 정보(Partial Observation)를 다른 에이전트들과 공유하여 합리적인 행동을 선택하도록 한다.

실행 과정에서도 정보를 공유한다는 점은 기존에 제안된 중앙집중형 훈련-분산형 실행(CTDE: Centralized Training and Decentralized Execution) 방식의 멀티에이전트 강화학습 알고리즘과 차이를 보인다. 중앙집중형 훈련-분산형 실행 방식의 훈련 과정에서 에이전트는 모든 에이전트의 관측 정보를 이용할 수 있다. 따라서 에이전트들이 전체 상황을 파악할 수 있으므로 빠르게 자신들의 최적 행동을 습득할 수 있다. 그러나 실행 과정에서는 훈련 과정에서 습득한 행동 규칙과 자신의 관측 정보만을 가지고 행동하게 된다. 이 과정에서 각 에이전트는 다른 에이전트들의 행동을 전혀 고려하지 않거나, 다른 에이전트의 예상된 행동을 기반으로 자신의 행동을 선택한다. 하지만 다른 에이전트들의 추정 행동과 실제 행동이 정확히 일치하지 않은 경우가 많아 에이전트들의 성능 최적화에 어려움이 있다. 멀티에이전트 강화학습 통신 알고리즘은 실행 과정에서도 부분 관측 정보를 공유함으로써 에이전트가 전체 환경 및 다른 에이전트들의 행동을 파악할 수 있으므로 전체 성능을 최적화시킬 수 있다.

최근까지 멀티에이전트 강화학습을 위한 통신 알고리즘들을 분석하기 위한 몇몇 문헌조사가 발표되었다[2,3]. 참고문헌 [2]에서는 멀티에이전트 강화학습 통신 알고리즘이 주목하는 문제점을 제시하고, 이 문제점을 해결하기 위해 발표되었던 알고리즘의 방식을 소개하였다. 참고문헌 [3]은 멀티에이전트 강화학습 통신 알고리즘들이 해결하고자 하는 문제에 따라 9가지 항목을 제안하였고, 이 항목에 맞추어 멀티에이전트 강화학습 통신 알고리즘을 분류하고, 항목별 분류 기준들을 소개하였다.

본고에서는 최근까지 발표되었던 멀티에이전트 강화학습 알고리즘들이 해결하고자 했던 문제 및 해결 방안 위주로 분류하여 기술하고자 한다. 먼저 Ⅱ장에서는 멀티에이전트 강화학습 통신 알고리즘들을 몇 가지 항목으로 분류하고, 항목별로 대표적인 멀티에이전트 강화학습 통신 알고리즘을 소개한다. 1절에서는 통신 알고리즘들이 전송하는 메시지의 내용, 2절에서는 통신 알고리즘에서 에이전트들이 메시지를 전송하는 대상, 3절에서는 통신 알고리즘들이 통신 정책을 학습하는 방법, 4절에서는 통신 제약 사항들의 고려 여부와 고려한 통신 제약 사항을 소개한다. Ⅲ장에서는 멀티에이전트 강화학습 통신 알고리즘들의 비교 및 검증을 위해 주로 사용되는 실험 환경에 관해서 서술한다.

Ⅱ. 멀티에이전트 강화학습을 위한 통신 알고리즘

이 장에서는 최근까지 발표되었던 멀티에이전트 강화학습 통신 알고리즘을 몇 가지 기준에 맞추어 분류하였고, 그 중 대표적인 통신 알고리즘 몇 가지

를 소개하고자 한다. 통신 알고리즘들은 각자 해결하고자 하는 문제점과 해결하는 방식이 모두 조금씩 다르다. 따라서 본고에서 제안하는 분류가 모든 알고리즘에 정확히 일치하는 것은 아니다. 그러나 최대한 통신 알고리즘들의 공통점을 찾아 분류하였으며, 각 통신 알고리즘의 특성도 잘 표현할 수 있도록 노력하였다.

멀티에이전트 강화학습 통신 알고리즘은 여러 가지 기준에 따라 분류할 수 있으며, 이 장에서는 메시지 전송 내용에 따른 분류, 통신 대상에 따른 분류, 통신 정책의 학습에 따른 분류, 고려한 통신 제약 사항에 따른 분류의 총 네 가지 사항을 제시하였다.

1. 메시지 전송 내용에 따른 분류

통신 메시지는 실행 과정에서 에이전트가 다른 에이전트의 현재 상황을 파악할 수 있도록 정보를 전달하는 임무를 수행한다. 주로 보내는 메시지의 내용은 에이전트가 관측한 부분 관측 정보(Partial Observation), 에이전트가 선택한 행동(Action), 현재 상태(State), 에이전트가 사용하는 파라미터(Parameter) 또는 가중치(Weight)가 포함되어 있을 수 있다.

표 1은 에이전트가 보내는 메시지의 내용에 따라 멀티에이전트 강화학습 통신 알고리즘들을 분류한 것이다.

표 1 메시지 전송 내용에 따른 멀티에이전트 강화학습 통신 알고리즘 분류

유형	알고리즘
관측 정보 및 현재 상태	CommNet, MADDPG-M, SchedNet, TarMAC, VBC, Diff Discrete, G2ANET, Gated-ACML, DGN, Neurcomm, NDQ, IMAC, ETCNet, variable length coding, I2C, TMC, R-MACRL, MASIA, AME
행동, 정책 및 기타 정보	RIAL, DIAL, BiCNet, ATOC, TarMAC, VBC, AME

출처 Reproduced from [3].

대다수의 멀티에이전트 강화학습 통신 알고리즘은 에이전트의 부분 관측 정보 또는 현재 상태를 메시지로 공유한다. CommNet[4]의 컨트롤러 모형은 여러 개의 층(Layer)으로 구성되어 있다. 첫 번째 층에서 자신의 현재 상태 벡터를 이용하여 메시지로 생성한 뒤 모든 에이전트가 메시지를 공유한다. 이후 전체 메시지(상태 벡터)의 평균값과 이전 층에서 생성한 메시지가 다음 층의 입력값이 된다. 생성된 출력값은 다시 모든 에이전트에게 공유되어 평균값을 계산하고, 자신이 가지고 있던 출력값과 함께 다음 층의 입력값이 된다. 이 과정을 몇 번 거친 후 최종적으로 행동을 선택하게 된다. 에이전트의 현재 상태를 모두에게 공유하고 평균값으로 반영하는 매우 단순한 형태이지만, 공유 과정에서 통신 정책을 학습할 수 있어 뛰어난 성능을 보여주었다.

에이전트의 관측 정보나 현재 상태가 아닌 행동, 정책이나 다른 정보를 공유하는 알고리즘도 있다. RIAL[5]은 에이전트의 현재 행동을 메시지로 생성하여 다른 에이전트에게 전달한다. 메시지를 전달받은 에이전트는 현재 자신의 관측 정보와 메시지를 이용하여 행동 가치를 계산하여 행동을 선택한다. 선택한 행동은 다시 다음 메시지가 되어 전달된다. RIAL의 방식은 통신 프로토콜을 학습하는 데 좋은 방식이다. 그러나 에이전트의 행동을 제외한 다른 값들은 공유되지 않고, 에이전트 간 피드백 과정이 없어 에이전트가 정책을 학습하는데 매우 부정확하고 오랜 시간이 걸린다는 단점이 있다.

DIAL[5]은 RIAL의 단점을 보완하기 위해, 경사값(Gradient)을 공유하는 과정을 추가하였다. 메시지와 피드백을 통해 경사값을 공유함으로써 더 많은 정보가 에이전트 사이에 공유되게 된다. 이를 통해 학습 과정에서 시행착오를 줄일 수 있으므로 빠르게 정책을 학습할 수 있다.

BiCNet[6]은 행동 가치 계산을 위해서 양방향 RNN(Bidirectional Recurrent Neural Network)을 이용한다. 행동 가치를 계산하는 과정에서 에이전트 간의 협력을 위해 통신을 수행하게 된다. 양방향 RNN의 입력값은 각 에이전트가 관측한 관측 정보이지만, 공유되는 정보는 양방향 RNN의 은닉층(Hidden Layer) 정보이다. 통신 프로토콜도 양방향 RNN에 합류한 순서를 고려하여 설정된다. 공유하는 메시지에 에이전트 간의 의존도가 포함되어 행동에 전달되기 때문에, 경사값이 전체 네트워크에 효율적으로 전파되어 완전한 의존을 허용하게 된다. 결과적으로 에이전트 간에 파라미터가 완전히 공유된다. 실험 결과, 협력에 필요한 에이전트의 수가 많은 시나리오에서 다른 알고리즘에 비해 더 좋은 성능을 보여주었고, 에이전트의 수가 늘어날수록 성능 격차가 더 커지는 것을 보여주었다.

TarMAC[7]은 메시지가 특정 에이전트에게만 효과적으로 반영되는 대상 지정 통신(Targeted Communication)을 수행한다. 자신의 관측 정보와 이전 과정에 생성한 집합 메시지(Aggregated Message)를 사용하여 에이전트의 행동 가치와 메시지를 생성한다. 이 과정에서 soft-attention을 사용하는데, soft-attention의 결과 중 signature와 value가 메시지에 포함된다. 메시지는 모든 에이전트에게 브로드캐스팅으로 전파되고, 전송받은 에이전트들은 받은 메시지와 메시지 생성 과정에서 사용하지 않은 Query를 사용하여 집합 메시지를 생성한다. Attention의 특징에 의해 집합 메시지는 특정 에이전트가 아니면 생성되지 않는다.

2. 통신 대상에 따른 분류

통신 대상은 1절에서 설명한 메시지를 에이전트들이 어떤 에이전트에게 공유하는지에 대한 내용이다. 크게 분류하면 에이전트가 다른 에이전트 모두에게 메시지를 전송하는 방법, 통신 범위 내 주변 에이전트에게 전송하는 방법, 그리고 중계 역할을 하는 에이전트에게 전송하는 방법으로 나눌 수 있다.

표 2 통신 대상에 따른 멀티에이전트 강화학습 통신 알고리즘 분류

유형	알고리즘
모든 에이전트	RIAL, DIAL, CommNet, BiCNet, SchedNet, TarMAC, VBC, Diff Discrete, ETCNet, variable length coding, TMC, R-MACRL, AME
통신 범위 내 주변 에이전트	DGN, Neurcomm, NDQ, I2C
중계 에이전트	ATOC, MADDPG-M, G2ANET, Gated-ACML, IMAC, MASIA

출처 Reproduced from [3].

표 2는 통신 대상에 따라 멀티에이전트 강화학습 통신 알고리즘들을 분류한 것이다.

대부분의 멀티에이전트 강화학습 통신 알고리즘은 통신 모델 및 프로토콜에 대한 설명 위주로 되어 있다. 그리고 네트워크에 포함된 모든 에이전트는 전부 통신이 가능하다고 가정하고 시작한다. 중앙집중형 훈련의 형태로 강화학습이 진행되기 때문에 모든 에이전트가 통신 정책을 학습하기 위해서는 이러한 가정들이 필수적이다. 그러나 실제 실행 환경에서는 통신거리로 인해 모든 에이전트의 통신이 불가능한 경우가 발생한다. 이 점을 통신 알고리즘을 설계할 때부터 고려하여, 주변 에이전트에게만 전송할 수 있도록 설정한 몇몇 알고리즘이 존재한다.

DGN[8]은 그래프 신경망(GNN: Graph Neural Network) 기반의 통신 알고리즘이다. 에이전트들은 관측 정보를 벡터화하여 메시지로 만든 뒤 1홉 거리의 에이전트와 공유한다. 전달받은 메시지와 자신의 메시지를 이용해 합성곱 층(Convolutional Layer)에서 잠재 벡터(Latent Vector)를 생성하게 된다. 합성곱 층

은 여러 층이 존재할 수 있으며, 합성곱 층이 많을수록 여러 홉 거리에 있는 에이전트들의 정보까지 포함할 수 있다. 행동 가치는 에이전트가 생성한 모든 층의 잠재 벡터를 사용하여 계산한다.

Neurcomm[9]은 모든 메시지가 어느 에이전트한테서 왔는지 판별 가능하다고 가정한다. 훈련 과정에서 에이전트는 자신의 주변에 있는 에이전트에게만 현재 상태, 은닉 상태(Hidden State), 정책을 메시지로 만들어 전송한다. 전송받은 메시지와 자신의 정보를 입력으로 하여 행동 가치를 계산하고, 결정된 행동을 바탕으로 평가한다.

I2C[10]는 에이전트가 현재 관측할 수 있는 범위 내에 있는 에이전트와만 통신할 수 있다. 이때, 통신 범위 내의 모든 에이전트와 통신하는 것이 아니라, 에이전트들의 공동 행동 가치(Joint-action Value)를 softmax로 비교하여 가장 값이 큰 에이전트와 통신한다.

몇몇 통신 알고리즘은 중계 에이전트를 사용해 통신한다. 중계 에이전트는 훈련이나 실행 과정에서 강화학습에 참여하는 것이 아니라, 에이전트들이 보낸 메시지를 병합 또는 가공하여 네트워크에 속한 에이전트들에게 재전송하는 역할만 수행한다. 중계 에이전트를 사용하는 방법은 중계 에이전트와만 통신을 수행한다는 점에서 통신 프로토콜 설계가 매우 간단하고, 대부분 전송받은 메시지를 압축하여 에이전트에게 전송함으로써 통신 오버헤드가 다른 알고리즘들에 비해 작다는 장점이 있다.

MADDPG-M[11]에서는 에이전트가 메시지를 공유하기 위해 Communication Medium에 메시지를 전송한다. 일정 시간 동안 전송된 메시지는 모두 병합되며, 병합된 메시지들은 다시 에이전트들로 전송된다. 현재와 과거 정보가 모두 포함된 메시지를 가지고 에이전트는 행동 결정 정책을 갱신하고, 정책에 따라 행동을 결정, 실행한다.

Gated-ACML[12]에서 에이전트들은 자신의 관측 정보를 이용하여 생성한 메시지를 Message CoordinatorNet(MCN)에 전송한다. MCN은 전송받은 메시지 중 중요 정보만 추출하여 하나의 메시지 형태로 가공한 후, 다시 에이전트들에 전송한다.

IMAC[13]은 제한된 대역폭(Bandwidth)에서 통신하는 상황을 고려하여, 전송하는 메시지들의 엔트로피(Entropy) 최소화를 위해 중앙 스케줄러에서 메시지 전송을 관장한다. 스케줄러는 에이전트들이 전송하는 메시지 중 서로 겹치는 정보를 최대한 줄이는 클리핑 과정을 수행하여, 메시지의 엔트로피를 최대한 낮추는 방향으로 학습한다.

MASIA[14]는 Attention을 사용하여 생성된 메시지를 중계 에이전트에서 받아 잠재 벡터를 생성한 후, 다시 에이전트들에게 전송한다. 이때 잠재 벡터의 크기를 최소화하기 위하여 MASIA는 과거와 현재 잠재 벡터를 이용하여 다음에 생성될 잠재 벡터를 예측한 뒤, 겹치는 정보를 최대한 줄이는 방법으로 잠재 벡터 메시지를 압축한다. 에이전트는 글로벌 관측 정보가 포함된 잠재 벡터에서 자신에게 가장 필요한 정보만 추출하여 행동 가치 함수에 사용한다. 정보 추출 과정은 MLP(Multi Layer Perceptron)에 잠재 벡터와 에이전트가 가장 필요한 정보를 추출하기 위한 가중치 정보를 입력으로 사용하고, 훈련 과정에서는 정보를 효율적으로 추출하기 위한 가중치 정보를 학습한다.

3. 통신 정책의 학습에 따른 분류

멀티에이전트 강화학습 통신 알고리즘에서 통신 정책은 훈련 시작부터 지정되어 수행되는 형태와 통신을 위한 정책을 따로 수립하여 훈련 과정에서 함께 훈련하는 두 가지 형태로 분류할 수 있다.

표 3은 통신 방법의 학습 여부에 따라 멀티에이전

표 3 통신 정책의 학습 여부에 따른 멀티에이전트 강화학습 통신 알고리즘 분류

유형		알고리즘
직접 지정	전체 에이전트	RIAL, DIAL, CommNet, BiCNet, TarMAC, Diff Discrete, NDQ, IMAC, variable length coding, MASIA, AME
	하나 및 일부 에이전트	VBC, DGN, Neurcomm, TMC
훈련 중 학습	에이전트별	ATOC, MADDPG-M, Gated-ACML, ETCNet, I2C
	전체 에이전트	SchedNet, G2ANET, R-MACRL

출처 Reproduced from [3].

트 강화학습 통신 알고리즘들을 분류한 것이다.

통신 정책을 직접 지정하는 방법 중 전체 에이전트와 통신은 아무런 제약 없이 모든 에이전트와 통신할 수 있는 것을 의미한다. 이 방식은 대부분의 멀티에이전트 강화학습 통신 알고리즘에 사용되며, 통신 과정에 특별한 가정이 없다면 이 방식에 속한다.

하나 및 일부 에이전트와 통신하는 방법은 에이전트의 통신거리를 제한하거나, 통신할 에이전트를 선택하는 방식이 수식 등으로 정해져 있는 방식을 의미한다.

VBC[15]는 에이전트 자신의 관측 정보, 이전 은닉 상태와 다른 에이전트로부터 전송받은 메시지를 이용해 행동 가치 함수를 계산한다. 에이전트가 가진 행동 가치의 편차가 일정 기준값을 넘지 못하면, 다른 에이전트에게 정보를 갱신하기 위한 값을 브로드캐스팅을 통해 요청하게 된다. 요청받은 에이전트들도 자신의 정보가 일정 기준값을 초과하였을 때만 메시지를 전송한다. 이러한 방식은 상황에 크게 영향을 주지 못하는 메시지들을 제거할 수 있어 통신 오버헤드를 줄이는 효과를 가져온다.

다른 방법은 훈련 중에 통신 정책을 학습하는 방법이 있다. 이 방식은 통신을 위한 정책을 따로 수립하여 훈련 과정에서 함께 훈련하고, 실행 과정에서는 훈련 과정에서 습득한 통신 정책을 바탕으로 에이전트 간 통신을 수행한다.

ATOC[16]는 행동 가치 함수를 계산하기 위해 2개의 actor net을 사용한다. 첫 번째 actor net은 에이전트 자신이 이전에 선택한 행동과 현재 관측 정보 등을 바탕으로 thought라는 핵심 정보만을 요약한 메시지를 생성한다. 생성된 thought는 통신 채널을 통해 에이전트끼리 공유되며, 이 과정에서 하나의 메시지로 병합한다. 두 번째 actor net은 thought와 메시지를 사용하여 에이전트의 행동을 결정한다. 에이전트 간 정보를 전달하는 과정에서 probabilistic gate mechanism을 사용하는데, 훈련 과정에서 정보를 전달할 에이전트를 선택하는 과정과 전달받은 thought를 하나의 메시지로 합치는 과정을 에이전트들이 개별로 학습한다.

위와 다르게 학습 과정에서 전체 에이전트를 같은 정책을 사용하여 학습시키는 경우도 존재한다. SchedNet[17]은 제한된 대역폭을 만족시키기 위해 WG(Weight Generator)를 사용하여 전체 에이전트들의 통신 스케줄을 학습한다. WG는 각 에이전트가 전송한 관측 정보의 중요도에 따라 가중치를 결정하는데, 이 값은 훈련 과정에서 학습되어 결정된다. 통신을 수행할 에이전트를 결정하는 방법은 가중치가 가장 높은 에이전트들로 선정하거나 (top(k)), softmax 함수를 통해 결정하게 된다. 설정된 결정 방법에 따라 선택된 에이전트들은 자신의 메시지를 브로드캐스팅하는 형태로 메시지를 전송한다.

G2ANET[18]은 soft-attention과 hard-attention을 사용하여 행동 가치를 계산한다. Soft-attention은 강화학습의 전체 가중치를 계산하기 위한 용도로 사용되는데, 관계없는 에이전트의 가중치까지 계산하여 계산복잡도가 증가한다는 단점이 있다. 이를

해결하기 위해 hard-attention을 통해 먼저 에이전트 간의 연관성을 탐색하여 계산 범위를 좁히기 위해 사용한다. Hard-attention과 soft-attention은 훈련 과정에서 행동 가치를 계산하는 GNN과 함께 학습된다.

4. 통신 제약 사항에 따른 분류

멀티에이전트 강화학습 통신 알고리즘은 에이전트의 상황을 정확히 전달하여 강화학습의 성능을 높이는 방법 위주로 서술된 경우가 많다. 그러나 잡음(Noise), 통신거리, 간섭(Interference), 적대적 공격(Adversarial attack) 등으로 인해 실제 환경에서는 메시지 전달이 완벽하게 이루어지지 않을 가능성이 높다. 이를 극복하기 위해 실제 통신 환경에서의 제약 사항을 고려한 통신 알고리즘들도 많이 제안되었다. 멀티에이전트 강화학습 통신 알고리즘에서 주로 고려되는 통신 제약 사항은 대역폭 제한, 적대적 공격 및 잡음으로 인한 메시지 위변조 등이 있다.

표 4는 멀티에이전트 강화학습 통신 알고리즘들이 고려한 통신 제약 사항에 따라 알고리즘들을 분류한 것이다.

앞서 설명한 몇몇 통신 알고리즘들은 대역폭 제한을 염두에 두고 통신 오버헤드 및 중복 메시지를 최소화하는 것을 목표로 하였다[5,12-15,17].

표 4 통신 제약 사항에 따른 멀티에이전트 강화학습 통신 알고리즘 분류

유형	알고리즘
대역폭 제한	RIAL, DIAL, SchedNet, VBC, Gated-ACML, NDQ, IMAC, ETCNet, variable length coding, TMC, MASIA
통신 잡음 및 적대적 공격	DIAL, Diff Discrete, R-MACRL, AME

출처 Reproduced from [3].

ETCNet[19]도 대역폭 제한 상황을 염두에 둔 통신 알고리즘이다. 에이전트가 gate mechanism을 사용하여 메시지 전송 여부를 결정한 후, 전송 확률을 계산한다. 전송 확률에 따라 에이전트가 메시지 전송 기회를 부여받았을 때만 메시지 전송이 가능하다. 전송 확률은 최댓값이 지정되어 있으며, 전송을 최근에 했거나 메시지 크기가 클수록 전송 확률은 감소한다.

Variable length coding[20]은 메시지의 크기를 변경하여 대역폭 제한을 해결하려고 하는 알고리즘이다. 기존 통신 분야에서 사용하는 variable length coding을 멀티에이전트 환경에 맞게 수정하여 사용하였다. 메시지 중 등장하는 빈도가 높은 메시지는 짧은 메시지로 보내고, 빈도가 낮은 메시지는 긴 메시지로 생성하여 전송한다. 이 기준으로 메시지 구성을 훈련 과정에서 학습하여 실행 과정에서 사용한다.

TMC[21]는 에이전트마다 메시지 버퍼를 사용하여 자신이 보낸 메시지와 다른 에이전트로부터 받은 메시지를 저장한다. 이때, 현재 자신이 저장한 메시지와 새로 생성한 메시지의 유클리드 거리(Euclidean Distance)를 계산하여 일정 기준값을 넘지 못한다면, 생성한 메시지는 새로운 정보가 없는 것으로 판단하여 전송하지 않는다.

NDQ[22]는 메시지를 전송받은 에이전트의 행동 결정의 불확실성을 낮추고, 메시지는 최대한 짧게 보내야 한다는 두 가지 목표를 만족시키는 방법을 제안하였다. 메시지의 상호의존정보(Mutual Information) 값을 최대화하고 엔트로피를 최소화하는 것을 훈련 과정에서 학습하도록 하였으며, 메시지 전송 시 최대한 필요 없는 부분의 비트(Bit)를 보내지 않는 형태로 메시지 길이를 최소화한다.

이 외에도 통신 과정에서 발생하는 잡음 또는 외부자 침입 및 에이전트 감염으로 인한 적대적 공격

을 고려한 통신 알고리즘도 존재한다. 잡음, 적대적 공격은 에이전트가 전송하는 메시지를 변경하여, 에이전트가 정확한 관측 정보 및 상황을 파악할 수 없도록 한다.

Diff Discrete[23]는 메시지를 전송하는 통신 채널의 구간을 나누어 전송 과정 중 잡음이 발생하더라도, 해당 구간 내에 신호가 도달하면 메시지를 완벽하게 복원하는 방법을 제안하였다. Diff Discrete는 멀티에이전트 강화학습보다는 기존의 통신 분야에서 제안하는 알고리즘 쪽에 가깝게 서술되어 있다.

ℜ-MACRL[24]은 적대적 공격으로 인한 메시지 변조를 복원하기 위한 방어 정책을 만드는 알고리즘이다. 훈련 과정에서 에이전트들은 공격자와 방어자로 나뉘어 학습이 진행된다. 공격자는 전체 알고리즘의 보상을 낮추는 정책을 학습하고, 방어자는 적대적 공격이 존재하는 상황에서 보상을 최대화하는 정책을 학습한다. 훈련을 진행하면서 공격자와 방어자의 갱신된 정책을 공유하고, 내쉬 균형 이론(Nash Equilibrium)을 사용하여 가장 좋은 정책을 선택하게 한 후 학습을 다시 진행한다. 훈련이 종료되고 나면 방어자가 학습했던 정책을 모든 에이전트가 수행하도록 한다.

AME[25]도 적대적 공격으로부터 강화학습의 성능을 방어하기 위한 멀티에이전트 강화학습 통신 알고리즘이다. AME에서는 적대적 공격을 수행하는 에이전트의 수가 전체의 절반을 넘지 않는다고 가정한다. 훈련 과정에서 에피소드마다 일부 에이전트들의 메시지만 학습하면, 적대적 공격을 수행하는 에이전트의 메시지보다 정상적인 메시지가 더 많이 들어 있는 경우가 많으므로, 결과적으로는 정상적인 통신 정책을 수립할 수 있게 된다고 주장한다. 표본 메시지의 개수가 많을수록 정책이 더 정확하지만, 처리 속도는 느리다. 표본 메시지의 개수가 적을수록 처리 속도는 빠르지만, 적대적 에이전트의 메시지가 주 의견이 될 가능성도 커지므로, 정책의 정확도는 낮아지게 된다.

Ⅲ. MARL 통신 실험 환경

멀티에이전트 강화학습을 위한 통신 알고리즘은 기존의 정책이나 가치 기반의 멀티에이전트 강화학습 알고리즘들과 결합하여 사용된다. 따라서 기존 멀티에이전트 강화학습에서 사용하는 실험 환경에도 적용할 수 있다. 그러나 에이전트 간 통신을 사용하여 획득할 수 있는 이득을 명확히 보여주기 위해, 같은 실험 환경에서도 에이전트 간 정보 교환이 필수적으로 포함된 시나리오를 많이 사용한다. 이 장에서는 소개한 멀티에이전트 강화학습 통신 알고리즘들이 사용하는 주요 검증 환경에 관해 기술하고자 한다.

1. Traffic Junction

Traffic Junction(TJ)은 멀티에이전트 강화학습에서 통신 알고리즘의 성능을 보여주기 위해 주로 사용되는 검증 환경이다. Commnet[4]에서 처음 등장하였고, 참고문헌 [26]에서 다양한 난이도를 설정할 수 있도록 발전되어 다양한 멀티에이전트 강화학습 통신 알고리즘 검증에 사용되었다.

Easy나 Medium 난이도 기준으로, 십자 모양의 교차로 하나를 격자로 구성한 환경에서 각 에이전트는 이 교차로상에서 움직이는 입자(자동차) 하나를 담당하며, 주어진 목적지로 빠르게 이동하도록 설정되어 있다. Hard 난이도의 경우 교차로의 개수와 에이전트의 수가 증가한다. 에이전트는 관측 범위가 지정되어 있고, 관측 범위보다 조금 큰 범위의 통신 범위를 갖고 있다. 에이전트 각각은 가장 빠른 시간 내로 목적지로 이동해야 하는 목표를 가짐과 동시에, 충돌을 일으키지 않으면서 이동해야 하

는 협력 목표를 갖는다. 충돌이 일어날수록 보상에서 더 큰 손해를 얻게 되어 있으므로, 통신을 통해 교차로를 통과하는 순서를 에이전트들이 결정해야 한다.

2. Multi-agent Particle Environment

Multi-agent Particle Environment(MPE)[27]는 Open AI Gym 기반으로 여러 형태의 멀티에이전트 시나리오를 검증할 수 있도록 만들어진 검증 환경이다. 에이전트는 입자 형태로 구성되어 입자 형태로 존재하는 목표(Prey, Landmark 등)로 빠른 시간 내에 정확히 이동하는 것이 이 환경의 목적이다. MPE에는 에이전트를 방해하기 위한 방해물(Obstacle) 및 적대적 에이전트(Adversarial Agent)가 존재한다. MPE는 다양한 시나리오를 제공함으로써 이전의 실험 환경에 비해 고차원적인 검증이 가능하다.

3. StarCraft Multi-Agent Challenge

StarCraft Multi-Agent Challenge(SMAC)[28]은 블리자드사의 StarCraft II 게임을 멀티에이전트 환경에 맞게 구축한 실험 환경이다. 같은 게임을 싱글 에이전트 강화학습에 사용하기 위한 DeepMind사의 PySC2를 기반으로 생성되었으나, PySC2와 달리 각각의 유닛 하나하나가 에이전트의 역할을 수행한다. 에이전트는 유닛의 종류에 따라 공격 범위, 체력, 이동속도, 특수능력 등이 모두 다르다. 실험의 제1목표는 승리이며, 이때 발생한 피해량, 유닛 손실 등을 토대로 전체 보상이 결정된다. 알고리즘의 성능 검증은 난이도 구분이 되어 있는 여러 시나리오의 승률에 따라 결정된다.

멀티에이전트 강화학습 통신 알고리즘은 협력이 강조되는 시나리오로 검증이 이루어진다. 예를 들어, '1o_2r_vs_4r' 시나리오의 경우, 1기의 대군주(Overseer)가 전달하는 적군의 위치를 바탕으로, 2기의 바퀴(Roach)가 4기의 적군(Reaper)을 물리쳐야 승리하는 시나리오이다. 1기씩 공격하면 각개 격파되어 패배한다. 정확한 타이밍에 동시에 공격해야만 승리할 수 있으므로, 에이전트 간 통신을 통한 협업이 SMAC의 다른 시나리오들에 비해 강조되는 시나리오라고 할 수 있다.

SMACv2[29]는 SMAC에 무작위 특성을 강조한 검증 환경이다. SMAC에서는 하나의 시나리오에 대해서 아군과 적군의 구성 및 위치가 훈련을 수행하는 동안 변경되지 않는다. 이러한 점에서 SMAC으로 검증하는 멀티에이전트 강화학습 알고리즘은 해당 시나리오 하나에 대해서는 성능 검증이 되지만, 다른 시나리오에 대해서는 검증이 되지 않았기 때문에 여러 시나리오에 대해 검증해야 한다는 불편함이 발생한다. 또한 SMAC을 적용하여 검증하는 멀티에이전트 강화학습 알고리즘들이 많아지면서, 모든 시나리오에 대해 90% 이상의 높은 승률을 보여주는 알고리즘들이 등장하고 있다. 이 때문에 기존보다 좀 더 어려운 시나리오가 필요하다는 의견들도 나타났다.

SMACv2에서는 전체 아군/적군의 유닛 수, 유닛 구성 비율을 지정하면, 생성 위치, 유닛 구성 등이 에피소드가 실행될 때마다 변경되어 좀 더 다양한 환경에서 멀티에이전트 강화학습 성능을 검증할 수 있도록 하였다. 에피소드마다 환경이 변경되므로, 에이전트 간 통신을 활용해 현재 상황을 빠르게 파악하고, 다른 에이전트들에게도 전달하는 것이 중요해졌기 때문에 멀티에이전트 강화학습 통신 알고리즘을 검증하는 데 좋은 환경이 될 수 있다.

4. Level Based Foraging

Level Based Foraging(LBF)[30]는 격자로 구성된 정사각형의 환경에 에이전트와 습득해야 하는 목표(열매)가 배치되어 시작된다. 각 에이전트 및 목표에는 레벨에 부여되어 있으며, 배치된 목표에 도달하여 자신의 레벨을 높이는 것이 목적이다. 목표에 부여된 레벨보다 낮은 경우, 에이전트 혼자서는 목표를 습득할 수 없다. 그러나 목표에 도달한 모든 에이전트 레벨의 총합이 목표보다 높다면 목표에 있는 레벨을 나누어 가질 수 있다. 알고리즘의 성능은 목표를 전부 습득한 시간과 에이전트 중 가장 높은 레벨, 전체 에이전트의 평균 레벨로 성능을 검증한다. LBF는 에이전트가 자신의 레벨을 높이는 경쟁 과정과 자신보다 높은 목표의 레벨을 습득하기 위해 다른 에이전트와 협력하는 협력 과정을 동시에 살펴볼 수 있다.

Ⅳ. 결론

본고에서는 멀티에이전트 강화학습을 위한 통신 알고리즘 연구에 대해 메시지 내용, 통신 대상, 통신 정책 학습, 알고리즘에서 고려한 통신 제약 사항들로 구분하여 간단히 소개한 후, 성능을 검증하기 위한 실험 환경에 대해 살펴보았다. 현재에도 많은 멀티에이전트 강화학습 통신 알고리즘이 제안되고 있지만, 해결하기 위한 문제와 해결 방법이 달라 모든 알고리즘을 전부 이해하기는 어렵다. 따라서 본고와 같이 일정 기준을 통해 통신 알고리즘을 분석하는 과정은 연구에 있어서 꼭 필요하다. 멀티에이전트 강화학습 통신 알고리즘은 현재 진행 중인 여러 멀티에이전트 강화학습 알고리즘의 성능을 높여주고 있다. 시뮬레이션이 아닌 실제 상황에 적용했을 때의 어려움을 미리 살펴보고 해결할 수 있다는 점에서 강화학습을 연구하는 연구자뿐만 아니라 기존의 통신, 네트워크를 연구하는 연구자에게도 매력적인 분야가 될 수 있을 것으로 생각한다.

용어해설

MARL(Multi-Agent Reinforcement Learning) 동일한 환경에서 다수의 에이전트가 경쟁/협력하는 강화학습 알고리즘

부분 관측 정보(Partial Observation) 하나의 에이전트가 관측할 수 있는 범위 내의 정보. 싱글 에이전트 강화학습에서는 모든 환경을 살펴볼 수 있는 것과 달리, MARL의 에이전트는 일부 환경만 관측 가능함

통신 정책(Communication Policy) 에이전트가 자신의 메시지를 전송하거나, 전송받을 에이전트를 선택하는 방법

통신 대역폭(Communication Bandwidth) 메시지 교환 과정에서 한 번에 통과시킬 수 있는 메시지의 최대량

약어 정리

ACML	Actor-Critic Message Learner
AME	Ablated Message Ensemble
ATOC	ATtentional Communication model
BicNet	multiagent Bidirectionally-Coordinated Network
CTDE	Centralized Training and Decentralized Execution
DIAL	Differentiable Inter-Agent Learning
ETCNet	Event-Triggered Communication Network
G2ANET	Game abstraction mechanism based on TWO-stage Attention NETwork
GNN	Graph Neural Network
I2C	Individually Inferred Communication
IMAC	Informative Multi-Agent Communication
LBF	Level Based Foraging
MACRL	Multi-Agent Communicative Reinforcement Learning
MADDPG-M	Multi-Agent Deep Deterministic Policy Gradient algorithm enhanced by a communication Medium
MARL	Multi-Agent Reinforcement Learning

MASIA	Multi-Agent communication via Self-supervised Information Aggregation
MLP	Multi Layer Perceptron
MPE	Multi-agent Particle Environment
NDQ	Nearly Decomposable Q-function
RIAL	Reinforced Inter-Agent Learning
RNN	Recurrent Neural Network
SMAC	StarCraft Multi-Agent Challenge
TarMAC	Targeted Multi-Agent Communication
TJ	Traffic Junction
TMC	Temporal Message Control
VBC	Variance Based Control
WG	Weight Generator

참고문헌

[1] C.V. Goldman and S. Zilberstein, "Decentralized control of cooperative systems: Categorization and complexity analysis," J. Artif. Intell. Res., vol. 22, 2004, pp. 143-174.

[2] 유병현 외, "멀티 에이전트 강화학습 기술 동향," 전자통신동향분석, 제35권 제6호, 2020, pp. 137-149.

[3] C. Zhu, M. Dastani, and S. Wang, "A survey of multi-agent reinforcement learning with communication," arXiv preprint, CoRR, 2022, arXiv: 2203.08975.

[4] S. Sukhbaatar, A. Szlam, and R. Fergus, "Learning multiagent communication with backpropagation," in Proc. Neural Inform. Process. Syst., (Barcelona, Spain), Dec. 2016, pp. 2244-2252.

[5] J.N. Foerster et al., "Learning to communicate with deep multi-agent reinforcement learning," in Proc. Neural Inform. Process. Syst., (Barcelona, Spain), Dec. 2016, pp. 2137-2145.

[6] P. Peng et al., "Multiagent bidirectionally-coordinated nets for learning to play starcraft combat games," arXiv preprint, CoRR, 2017, arXiv: 1703.10069.

[7] A. Das et al., "Tarmac: Targeted multi-agent communication," in Proc. Int. Conf. Mach. Learn., (Long Beach, CA, USA), June 2019, pp. 1538-1546.

[8] J. Jiang et al., "Graph convolutional reinforcement learning," in Proc. Int. Conf. Learn. Representations, (Addis Ababa, Ethiopia), Apr. 2020.

[9] T. Chu, S. Chinchali, and S. Katti. "Multi-agent reinforcement learning for networked system control," in Proc. Int. Conf. Learn. Representations, (Addis Ababa, Ethiopia), Apr. 2020.

[10] Z. Ding, T. Huang, and Z. Lu. "Learning individually inferred communication for multi-agent cooperation," in Proc. Neural Inform. Process. Syst., (Vancouver, Canada), Dec. 2020, pp. 22069-22079.

[11] O. Kilinc and G. Montana. "Multi-agent deep reinforcement learning with extremely noisy observations," arXiv preprint, CoRR, 2018, arXiv: 1812.00922.

[12] H. Mao et al., "Learning agent communication under limited bandwidth by message pruning," Proc. AAAI Conf. Artif. Intell., vol. 34, no. 4, 2020, pp. 5142-5149.

[13] R. Wang et al., "Learning efficient multi-agent communication: An information bottleneck approach," in Proc. Int. Conf. Mach. Learn., (Vienna, Austria), July 2020, pp. 9908-9918.

[14] C. Guan et al., "Efficient multi-agent communication via self-supervised information aggregation," in Proc. Neural Inform. Process. Syst., (New Orleans, LA, USA), Nov. 2022, pp. 1020-1033.

[15] S.Q. Zhang, Q. Zhang, and J. Lin. "Efficient communication in multi-agent reinforcement learning via variance based control," in Proc. Neural Inform. Process. Syst., (Vancouver, Canada), Dec. 2019, pp. 3230-3239.

[16] J. Jiang and Z. Lu. "Learning attentional communication for multiagent cooperation," in Proc. Neural Inform. Process. Syst., (Montreal, Canada), Dec. 2018, pp. 7265-7275.

[17] D. Kim et al., "Learning to schedule communication in multiagent reinforcement learning," in Proc. Int. Conf. Learn. Representations, (New Orleans, LA, USA), May 2019.

[18] Y. Liu et al., "Multi-agent game abstraction via graph attention neural network," Proc. AAAI Conf. Artif. Intell., 2020, vol. 34, no. 5, pp. 7211-7218.

[19] G. Hu et al., "Event-triggered multi-agent reinforcement learning with communication under limited-bandwidth constraint," arXiv preprint, CoRR, 2020, arXiv: 2010.04978.

[20] B. Freed et al., "Sparse discrete communication learning for multi-agent cooperation through backpropagation," in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., (Las Vegas, NV, USA), Jan. 2020, pp. 7993-7998.

[21] S.Q. Zhang, Q. Zhang, and J. Lin. "Succinct and robust multi-agent communication with temporal message control," in Proc. Neural Inform. Process. Syst., (Vancouver, Canada), Dec. 2020. pp. 17271-17282.

[22] T. Wang et al., “Learning nearly decomposable value functions via communication minimization,” in Proc. Int. Conf. Learn. Representations, (Addis Ababa, Ethiopia), Apr. 2020.

[23] B. Freed et al., “Communication learning via back-propagation in discrete channels with unknown noise,” Proc. AAAI Conf. Artif. Intell., vol. 34, no. 5, 2020, pp. 7160-7168.

[24] W. Xue et al., “Mis-spoke or mis-lead: Achieving Robustness in Multi-Agent Communicative Reinforcement Learning,” in Proc. Int. Conf. Auton. Agents Multiagent Syst., (Richland, SC, USA), 2022, pp. 1418-1426.

[25] Y. Sun et al., “Certifiably robust policy learning against adversarial multi-agent communication,” in Proc. Int. Conf. Learn. Representations, (Kigali, Rwanda), May 2023.

[26] A. Singh, T. Jain, and S. Sukhbaatar, “Learning when to communicate at scale in multiagent cooperative and competitive tasks,” in Proc. Int. Conf. Learn. Representations, (Vancouver, Canada), Dec. 2018.

[27] R. Lowe et al., “Multi-agent actor-critic for mixed cooperative-competitive environments,” in Proc. Neural Inform. Process. Syst., (Long Beach, CA, USA), Dec. 2017.

[28] M. Samvelyan et al., “The StarCraft Multi-Agent Challenge,” arXiv preprint, CoRR, 2019, arXiv: 1902.04043.

[29] B. Ellis et al., “SMACv2: An improved benchmark for cooperative multi-agent reinforcement learning,” arXiv preprint, CoRR, 2022 arXiv: 2212.07489.

[30] F. Christianos, L. Schafer, and S. Albrecht, “Shared experience actor-critic for multi-agent reinforcement learning,” in Proc. Neural Inform. Process. Syst., (Vancouver, Canada), Dec. 2020. pp. 10707-10717.