Journal of the KIMST, Vol. 27, No. 4, pp. 474-484, 2024
ISSN 1598-9127(Print) · 2636-0640(Online)
DOI https://doi.org/10.9766/KIMST.2024.27.4.474

Research Paper 제어·구동 부문

멀티에이전트 강화학습에서 견고한 지식 전이를 위한 확률적 초기 상태 랜덤화 기법 연구

김도현[1] · 배정호[*,1]

[1] 국방과학연구소 국방AI센터

Stochastic Initial States Randomization Method for Robust Knowledge Transfer in Multi-Agent Reinforcement Learning

Dohyun Kim[1] · Jungho Bae[*,1]

[1] *Defense AI Center, Agency for Defense Development, Korea*

(Received 9 May 2024 / Revised 28 June 2024 / Accepted 9 July 2024)

Abstract

Reinforcement learning, which are also studied in the field of defense, face the problem of sample efficiency, which requires a large amount of data to train. Transfer learning has been introduced to address this problem, but its effectiveness is sometimes marginal because the model does not effectively leverage prior knowledge. In this study, we propose a stochastic initial state randomization(SISR) method to enable robust knowledge transfer that promote generalized and sufficient knowledge transfer. We developed a simulation environment involving a cooperative robot transportation task. Experimental results show that successful tasks are achieved when SISR is applied, while tasks fail when SISR is not applied. We also analyzed how the amount of state information collected by the agents changes with the application of SISR.

Key Words : Multi-Agent Reinforcement Learning(멀티에이전트 강화학습), Transfer Learning(전이학습), Deep Reinforcement Learning(심층 강화학습), Robust Knowledge Transfer(견고한 지식 전이)

1. Introduction

현재 우리 군은 무인 로봇에 대해 원격으로 제어하는 방식을 취하고 있으나, 미래에는 자율 에이전트가 통제하는 이종/다중 로봇들의 비중이 증가할 것으로 기대된다. 기존에 규칙 기반으로 제어하던 다양한 분야에 강화학습을 접목하는 연구가 이뤄지고 있으며, 군사 분야에서도 괄목할 만한 성과를 이뤄내고 있다. 무인 전투기에 대한 공중전 기동[1], 군집제어[2]가 그 예시이다.

강화학습은 에이전트가 환경과 상호작용하며 학습하는 패러다임이다. 강화학습 모델을 학습시키기 위해서는 대량의 샘플 데이터가 필요하며, 이를 샘플 효율

* Corresponding author, E-mail: jhbae@add.re.kr

성 문제라 한다. 강화학습 모델이 새로운 정보를 수집하는 것과, 알려진 정보에 기반해 최적의 행동을 선택하는 것 사이의 균형을 유지해야하는데, 이를 탐색-이용 딜레마라 부른다. 강화학습의 탐색-이용 딜레마를 해결하기 위해 반복적인 환경과의 상호작용으로 충분한 양의 데이터를 수집하여야 한다.

샘플 효율성 문제로 인해 많은 강화학습 연구는 시뮬레이션 환경을 기반으로 한다. 멀티에이전트 강화학습을 위한 대표적인 환경으로 실시간 전략 게임인 스타크래프트를 기반으로 한 StarCraft Multi-Agent Challenge (SMAC)[3], 협업 및 경쟁 상태에서 여러 에이전트가 상호작용하는 Multi Particle Environments(MPE)[4], 11대 11의 축구 경기를 모사한 Google Research Football[5] 등이 연구되었다.

강화학습 분야에서 학습 성능을 높이기 위한 방법으로 전이학습(Transfer Learning)이 존재한다. 강화학습에서 말하는 전이학습이란 외부에서 얻은 지식을 바탕으로 학습 과정에 도움을 주는 접근법을 말한다. 외부의 지식을 학습에 활용하는 것으로 강화학습이 직면한 샘플 효율성 문제를 완화한다. 전이학습을 적용할 경우, 기존 대비 적은 양의 샘플로도 모델이 학습할 수 있으며 기존보다 더 나은 성능을 보이기도 한다.

멀티에이전트 강화학습에도 전이학습을 접목하기 위한 다양한 시도가 이어졌다. 우선, Wang[16]은 Graph Neural Network를 멀티에이전트 강화학습에 도입하여 입출력 크기가 바뀌더라도 전이가능한 아키텍처를 제안하였고, Hu[17]와 Zhou[18]는 attention 매커니즘을 통해 모델의 입출력 크기를 완화한 바 있다. 멀티에이전트 강화학습 관점에서 전이학습의 성능을 향상시킨 다른 시도로는 Zeng[19]의 연구가 있다, 해당 연구는 이미지의 표현 학습에 좋은 성능을 보인 contrastive learning을 가져와, 에이전트들이 달성해야 할 하위 목표를 생성하는 접근으로 샘플 효율성을 향상시킨 바 있다.

멀티에이전트 강화학습에 전이학습을 적용시키려는 다양한 노력에도 불구하고, 전이학습에서 중요한 요소인 외부 지식과 관련된 논의는 활발히 이뤄지지 않았다. 전이학습 방법론에 따르면, 상대적으로 쉽고 간단한 원본 도메인에서 학습한 후 본래 풀고자 하는 대상 도메인으로 지식을 전이하게 된다. 이때, 강화학습 모델이 원본 도메인의 문제에 과적합되거나 학습한 지식의 다양성이 부족한 경우에는 에이전트들이 대상 도메인에서 우수한 성능을 달성하지 못할 수 있다. 또한, 원본 도메인과 대상 도메인 간의 간극이 큰 경우, 원본 도메인의 지식이 대상 도메인에서의 학습에 악영향을 미치는 부정적 전이 현상 또한 발생할 수 있다.[20] 우리는 이러한 문제를 해결하고 전이학습으로 인한 샘플 효율성을 향상시키기 위해 확률적 초기 상태 랜덤화(Stochastic Initial States Randomization, SISR) 기법을 제안한다. SISR 기법은 초기 상태 분포를 확률적으로 랜덤화하여, 에이전트들이 환경에 대해 더욱 다양한 정보를 수집하도록 촉진하는 기법이다. 에이전트들이 학습할 데이터의 정보량을 증가시켜, 두 도메인 간의 차이가 존재해도 견고하게 지식을 전이할 수 있도록 한다. 우리의 기법에 대한 유효성을 검증하기 위해 무인 로봇들의 전투 상황을 모사한 시뮬레이션 환경을 개발하였다. 시뮬레이션 환경을 통해, 우리의 기법을 적용하는 것으로 지식 전이를 통한 학습 속도와 성능 향상이 있었음을 실험적으로 보였다. 또한, SISR 기법으로 인해 수집한 학습 데이터의 정보량이 상승한 것을 확인하였다.

본 논문의 구성은 다음과 같다. 2장에서는 강화학습과 전이학습에 대한 기본 개념을 소개하며, 3장에서는 본 연구에서 제안한 SISR 기법에 관해 설명한다. 그리고 4장에서는 협업 시뮬레이션 환경과 실험에 관해 설명하며, SISR 기법의 적용에 따른 정보량 변화를 분석하였다.

2. Background

2.1 Markov Decision Process(MDP)

MDP는 의사결정 과정을 확률적 그래프로 나타낸 것으로, 현재 상태와 행동에 따라 다음 상태가 결정된다고 가정한다. MDP는 튜플 $(\mu_0, S, A, T, \gamma, R)$로 나타낸다. 시간 t에서의 상태 $s_t \in S$에서 특정 행동 $a_t \in A$를 수행했을 때, 전환(transition) 함수 $T(s_{t+1}|s_t, a)$에 따라 다음 상태 s_{t+1}로 변환되며 그에 따른 보상 $R(s_t, a_t, s_{t+1})$이 주어지게 된다. 그리고 누적 보상은 감쇠율(discount factor) $\gamma \in (0,1]$에 따라 감쇠한다. 또한 초기 상태 s_0는 μ_0의 분포를 따른다. 이러한 MDP 프로세스에서 에이전트는 현재 상태에서 보상이 최대가 되는 행동을 수행하여 전체 보상의 합이 최대가 되는 정책을 학습하는 것이 강화학습의 최종 목표이다.

2.2 멀티에이전트 강화학습 알고리즘

최근 멀티에이전트 강화학습(MARL) 연구는 중앙집중형 방식과 분산형 방식을 혼합한 Centralized Training with Decentralized Execution(CTDE) 패러다임을 따르고 있다. 각 에이전트가 선택할 행동은 각자의 관측 정보만을 이용하여 판단하며, 학습 단계에서는 모든 에이전트의 관측 정보와 선택한 행동 정보를 취합하여 학습의 방향을 판단하게 된다. CTDE 패러다임을 적용하는 것으로 에이전트 수에 따른 확장성과 에이전트 간의 협력을 모두 보장할 수 있다. CTDE 패러다임 아래에서 Deepmind는 Value Decomposition Network (VDN)[6] 알고리즘을 제안하였다. VDN 알고리즘은 식 (1)과 같이, 개별 에이전트의 가치 함수 Q_i의 합을 공동 가치 함수 Q_{jt}로 정의하였다. 여기서 s^i, a^i는 각각 i번째 에이전트의 상태와 행동을 나타낸다.

$$Q_{jt}(s,a) = \sum_{i=1}^{N} Q_i(s^i, a^i) \tag{1}$$

VDN 알고리즘은 개별 가치 함수의 선형적 결합에 대해서만 표현할 수 있다. 이를 확장하고자 Rashid의 QMIX[7] 알고리즘은 개별 가치 함수를 입력으로 하는 인공 신경망을 두었고, 이를 mixing network라고 하였다. 이때, 실행 단계에서 사용되는 개별 가치 함수와 학습 단계에서 사용되는 공동 가치 함수간의 행동을 일치시키는 일관성(consistency)를 식 (2)와 같이 만족해야 한다. QMIX 알고리즘의 경우 식 (3)의 단조성(monotonicity)을 부과하여 일관성을 충족하였다.

$$\underset{\boldsymbol{a}}{argmax}\, Q_{jt}(s,\boldsymbol{a}) = \begin{pmatrix} \underset{a_1}{argmax}\, Q_1(s,a^1) \\ \cdots \\ \underset{a_n}{argmax}\, Q_n(s,a^n) \end{pmatrix} \tag{2}$$

$$\frac{\partial Q_{jt}}{\partial Q_i} \geq 0, \forall i \in N \tag{3}$$

QMIX를 구성하는 mixing network와 개별 에이전트의 네트워크는 deep Q-learning을 통해 학습한다. 각 에이전트의 행동-가치(action-value) 함수는 DQN과 같이 인공 신경망으로 표현하며, 리플레이 메모리를 이용해 학습한다. QMIX는 식 (4)의 손실 함수를 최소화하는 것으로 모든 네트워크가 학습된다. 이때 $y^{jt} = r + \gamma\, max_{\boldsymbol{u}'}\, Q_{jt}(\boldsymbol{\tau}', \boldsymbol{u}', s'; \theta^-)$이며 θ^-는 DQN[8]에서와 같이 타겟 네트워크의 파라미터이다.

$$L(\theta) = \sum_{i=1}^{batch} [(y_i^{jt} - Q_{jt}(\tau, a, s; \theta))^2] \tag{4}$$

QMIX는 다양한 멀티에이전트 문제에 대해 우수한 결과를 보였지만, 단조성을 만족하지 않는 정책을 학습할 수 없다는 제약사항이 존재했다. 이러한 제약을 완화하고 더 일반적인 정책을 학습하기 위해 QTRAN 알고리즘이 발표되었다. QTRAN 알고리즘은 제약사항을 식 (5)와 같이 완화하였고, 이 제약사항을 만족할 때 일관성을 가진다는 것을 이론적으로 증명하였다.

$$\sum_{i=1}^{n} Q_i(\tau_i, u_i) - Q_{jt}(\boldsymbol{\tau}, \boldsymbol{u}) + V_{jt}(\boldsymbol{\tau}) = \begin{cases} 0 & \boldsymbol{u} = \bar{\boldsymbol{u}} \\ \geq 0 & \boldsymbol{u} \neq \bar{\boldsymbol{u}} \end{cases} \tag{5}$$
$$where\ V_{jt}(\boldsymbol{\tau}) = \max_{\boldsymbol{u}} Q_{jt}(\boldsymbol{\tau}, \boldsymbol{u}) - \sum_{i=1}^{n} Q_i(\tau_i, \bar{u}_i)$$

QMIX와 QTRAN 모두 멀티에이전트 강화학습 알고리즘의 주요 벤치마크인 SMAC에서 우수한 결과를 보였다. 따라서, 본 연구에서는 충분히 검증된 두 알고리즘에 대해 우리의 SISR 기법을 적용하는 것으로 SISR 기법의 우수성을 실험적으로 평가하였다.

2.3 전이학습

강화학습에서 말하는 전이학습은 다음과 같다. 원본 도메인(source domain) M_s과 대상 도메인(target domain) M_t이 주어졌을 때, 원본 도메인 M_s에서 얻은 외부 정보 I_s와 대상 도메인 M_t에서 얻은 내부 정보 I_t를 통해 대상 도메인에 대한 최적의 정책 π^*을 학습하는 것을 전이학습이라 한다[9]. 즉, 에이전트가 풀고자 하는 문제와 관련된 사전 지식이 존재하는 경우, 이 지식을 이용하여 풀고자 하는 문제를 효율적으로 학습하는 접근법이다. 전이학습은 전이하고자 하는 지식의 종류에 따라 분류할 수 있다. 우선 전문가 정책의 시연 궤적(demonstrated trajectories)을 지식으로 전이하는 방법이다. 학습 에이전트는 효율적인 탐색을 위해 외부의 시연 정보를 바탕으로 문제에 대한 이해를 높일 수 있다. 대표적인 알고리즘으로 GAIL[10], DDPGfD[11] 등이 연구되었다. 다음으로 정책 전이(policy transfer)를 통한 방법이다. 정책 전이 기법은 이미 학습된 정

책에 포함된 중요한 정보들을 간접적으로 학습하는 방식과, 학습된 정책을 직접 사용하는 방식으로 구분된다. 각 방식에 대표적인 알고리즘으로 각각 policy distillation[12], policy reuse[13] 알고리즘이 존재한다. 마지막으로 표현 전이(representation transfer) 방법도 존재한다. 심층 신경망에서 입력 값이 레이어를 거쳐 처리된 값을 표현(representation)의 형태로 지식을 전이하는 방법을 말한다. 대표적으로 Progressive Net[14], PathNet[15]이 존재한다.

3. Method

3.1 Stochastic Initial States Randomization

전이학습은 원본 도메인에서 적절히 지식을 추출한 뒤 대상 도메인으로 지식을 전달하는 방법으로, 전이학습을 통해 대상 도메인의 학습 성능(학습 속도, 최고 성능 등)의 향상을 기대할 수 있다. 전이학습을 통해 에이전트가 더 적은 데이터만으로 학습할 수 있거나, 같은 양의 데이터를 환경으로부터 수집했을 때 더 높은 성능을 보일 수 있다는 것이다. 전이학습을 통한 성능의 향상을 보장하고, 좋은 지식을 전이하기 위해서는 원본 도메인의 정보를 적절히 추출하는 것이 필요하다. 즉, 에이전트들이 원본 도메인에서 학습할 때에는 대상 도메인에서 잘 활용될 수 있는 지식들을 학습하여야 한다는 것이다. 에이전트들이 대상 도메인에서 잘 활용될 수 있는 지식을 학습하기 위해서는 원본 도메인에만 과적합된 지식이 아닌, 다양하고 일반화된 데이터를 수집하여 견고한 지식을 학습해야 한다.

본 연구에서는 에이전트가 환경에 대해 다양하고 일반화된 데이터를 수집하도록 촉진하는 방법으로, 초기 상태 분포 μ_0를 확률적으로 랜덤화하는 확률적 초기 상태 랜덤화(Stochastic Initial States Randomization, SISR) 기법을 제안한다. SISR 기법은 풀고자하는 문제에 대해 MDP의 초기 상태 분포를 균등 분포로 대체하여, 에이전트가 존재할 수 있는 모든 위치에서 랜덤으로 초기화하는 방법이다. 작은 확률(대략 ~1 %)로 초기 상태를 무작위로 설정할 경우, 그렇지 않을 때보다 도메인에 대해 더욱 다양한 정보를 수집할 수 있다. 이를 통해 모델이 더 일반화되도록 학습을 유도하며, 이는 곧 원본 도메인 지식에 대한 견고한 지식 전이가 가능하다는 것을 의미한다. SISR 기법을 통해 견고한 지식을 받은 경우, SISR 기법을 적용하지 않았을 때보다 대상 도메인에서의 학습 성능이 더 우수하게 나타났다. 학습 결과에 대한 구체적인 내용은 4장에서 설명하였다.

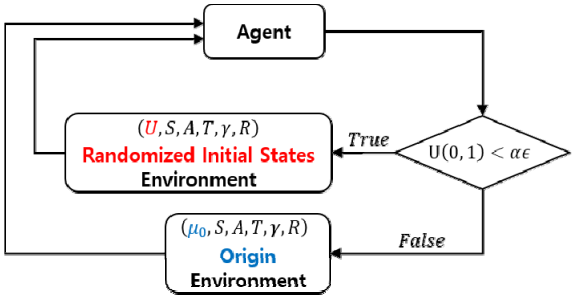

Fig. 1. Stochastic initial states randomization(SISR) method diagram

Algorithm 1 Stochastic Initial States Randomization

Initialize network Q, *memory*, *env*
for i in 1 to *max_episodes* **do**
  $rand \leftarrow$ random number from uniform distribution $U(0,1)$
  if $rand < \alpha \cdot \epsilon$ **then**
    $s \leftarrow env.\text{reset}(random = True)$
  else
    $s \leftarrow env.\text{reset}(random = False)$
  end if
  while not *done* **do**
    $a \leftarrow$ Get action using epsilon-greedy
    $s', r, done \leftarrow env.\text{step}(a)$
    Store (s, a, r, s') in *memory*
  end while
  Decay ϵ
  Update θ with *memory*
  Periodically update target network θ^-
end for

Fig. 2. Algorithm of Stochastic initial states randomization(SISR)

SISR 기법을 도입하여 도메인의 더 다양한 탐색 공간을 탐색하여 다양한 정보를 수집할 수 있다. 이와 동시에, 도메인에서 본래 풀고자하는 문제에 대해 학습하여야 하기에, 본래 정해진 위치에서 초기화하는 원본의 초기 상태 분포 μ_0를 선택하는 것도 필요하다. 즉, 탐색-이용 사이에 적절한 균형을 이루는 것이 중요하다. 본 기법을 도입하는 데에 있어 최적의 탐색-이용 균형을 맞추기 위해, 학습이 진행됨에 따라 랜

덤화된 초기 상태 분포의 선택 비율을 서서히 감쇠하였다. 또다른 탐색-이용 기법 중 하나인 epsilon-greedy 기법과 본 연구에서 제안한 SISR 기법의 감쇠 속도를 연동하였다. epsilon-greedy 기법에서 탐색의 선택 확률을 의미하는 ϵ 변수에 대해, SISR에서 $\alpha\epsilon$의 확률로 랜덤화된 초기 상태 분포를 선택하도록 하였다. 여기서 α는 하이퍼파라미터이다.

SISR 기법을 구현하는 방법으로 SMAC 환경의 경우 초기 위치, 체력 등을 랜덤으로 초기화할 수 있겠다. 본 연구에서 구축한 협업 시뮬레이션 환경에 대해서는 위치 정보를 랜덤으로 초기화하였다. 시뮬레이션 환경의 구체적인 설명과 SISR의 구현에 대해서는 4.1 절에서 자세히 소개하였다.

3.2 지식 전이

앞서 2.3 절에서 설명한 것과 같이, 전이학습을 통해 전달할 지식으로는 다양한 형태가 존재한다. 본 연구에서는 원본 도메인에서 대상 도메인으로 전이되는 지식을 정책의 형태로 전이하였다. 인공신경망으로 근사된 정책 함수를 직접적으로 전이함으로써, 견고하게 학습된 지식을 곧바로 대상 도메인 학습에 활용할 수 있다. 여기에 추가적으로, Wang의 buffer reuse[16] 기법에 영감을 받아 원본 도메인 학습에 쓰였던 리플레이 메모리를 재사용하였다. 본 연구에서는 buffer reuse 기법을 변형하여, 원본 도메인에서의 메모리를 대상 도메인에서의 리플레이 메모리에 채운 후 학습을 시작하였다. 즉, 학습이 진행될수록 원본 도메인에서의 오래된 정보들은 서서히 비워지게 된다. 또한 원본 도메인에서 배운 지식이 epsilon-greedy 탐색 과정에서 사라지는 것을 방지하기 위해, 대상 도메인에서 학습을 시작할 때의 초기 ϵ을 0.5로 설정하였다. 이 경우 epsilon-greedy의 랜덤 행동과 학습된 정책 간의 탐색-이용 조화를 이뤄 전이받은 지식을 크게 잃지 않을 수 있다. 버퍼를 재사용하고 ϵ를 0.5로 초기화하는 것으로, 과거의 지식을 보유함과 동시에 새로운 도메인을 탐색하여 효율적으로 학습할 수 있다.

4. Experiments

4.1 환경의 구성

3장에서 설명한 SISR 기법의 유효성을 검증하고 다중 에이전트의 협업 작전을 나타내기 위해 시뮬레이션 환경을 구성하였다. 본 연구에서는 로봇들이 물자를 수송하는 과정에서 적과 조우하는 상황을 시뮬레이션으로 표현하였다. 2차원의 지도상에 위치하는 도착 지점까지 아군 수송기가 파괴되지 않으며 이동하는 것이 작전의 목표이다. 수송 경로상에 적 포탑과 지뢰가 길을 가로막고 있으며, 적 병력에 대응하기 위해 아군 호위 로봇은 적을 공격하고 지뢰를 개척하여 아군 수송기를 보호한다. 시뮬레이터는 Fig. 3에 나타나 있으며, 수송기는 청색, 아군 호위 로봇은 녹색, 적 포탑은 황색, 적 지뢰는 적색, 도착지점은 회색으로 표현하였다. 아군 호위 로봇과 적 포탑의 음영 구역은 5°의 Weapon Engagement Zone(WEZ)을 나타내며, WEZ 내에 적이 존재할 경우 공격자의 공격력만큼 피격자의 체력을 감소시킨다. 체력이 0 이하로 내려가거나 지뢰에 피격된 경우 해당 객체는 파괴되어 이동, 공격 등 일체의 행동을 수행하지 못한다.

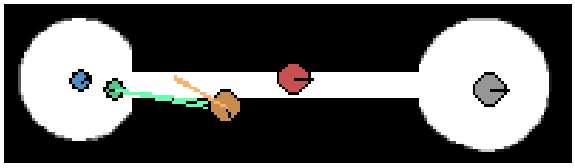

Fig. 3. Simulation environment

위에서 소개한 시뮬레이션 환경은 MDP 구조를 따른다. 따라서 에이전트는 환경에 대한 관측 정보를 바탕으로 행동을 결정하며, 환경은 행동에 따른 다음 상태를 반환하는 절차를 따른다. 위 환경에서 보상 함수는 Table 2와 같이 정의하였으며, 관측 공간과 행동 공간은 각각 Table 3, Table 4와 같다. 이때, 적 지뢰는 땅에 매설된 상황을 상정하여 에이전트의 관측에도 적 지뢰의 정보는 나타나지 않도록 설정하였다.

Table 1. Unit attributes

Type	Attack damage	Life	Range	Team
Convoy	-	50	0	Ally
Defender	2	100	17	Ally
Tank	5	100	10	Enemy
Mine	100	-	0	Enemy

Table 2. Reward function

Condition	Reward
Convoy Dead	−10
Convoy reaches landmark	+10
Defender clears a mine	+1
Defender eliminates a turret	+1
Convoy moves to landmark	$\exp(\Delta d_{convoy}/4)-1$
Defender moves to landmark	$\exp(\Delta d_{defender}/4)-1$
Defender attacks an enemy	$\sum damage$

Table 3. Observation features for one agent

Feature	Description
global map	(3, 64, 64) color image
local map	range measurement for every 5°
role	one-hot vector, convoy or defender
ego life	0~100 integer
destination info	relative distance, relative angle
ally agent info	relative distance, relative angle, life
enemy info	relative distance, relative angle, life

Table 4. Action space for one agent

Forward	Backward	Turn Left	Turn Right
v = +2.0	v = −0.5	w = +0.5	w = −0.5

SISR 기법을 구현하기 위해, 초기 상태 분포를 랜덤화할지의 여부를 선택할 수 있다. 에피소드를 시작하기 전, 환경을 초기화할 때 주어지는 랜덤 변수에 따라 고정 위치에 객체들을 초기화할지, 랜덤 위치에 객체들을 초기화할지가 결정된다. 고정 위치로 초기화하는 경우, Fig. 5과 같이 객체별 초기 구역 내에서 시작 위치가 국소적 랜덤으로 설정된다. 랜덤 위치에 초기화하는 경우, 아군 객체들은 지도상의 흰색으로 표기된 모든 위치 중 랜덤으로 시작 위치가 결정되며, 적 포탑의 경우 확장된 구역 내에서 랜덤으로 시작 위치가 결정된다.

Fig. 4. Fixed(left) and randomized(right) initial states regions

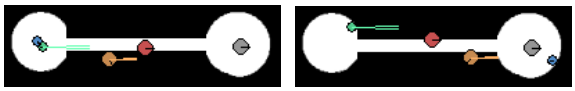

Fig. 5. Fixed(left) and randomized(right) spawn position units

시뮬레이션 환경을 구성하기 위해 Multi Particle Environments(MPE)[4] 환경을 참조하였다. OpenAI에서 제안한 MPE 환경은 다수의 입자들이 상호작용하며 협력, 경쟁 등의 미션을 달성하는 환경이다. 이번 연구에서는 Farama 재단에서 공개한 MPE 환경의 소스 코드를 기반으로 시뮬레이션 환경을 구현하였다.

4.2 실험의 구성

SISR 기법의 목적은 원본 도메인에서 수집한 정보를 최대화하여 대상 도메인으로의 지식 전이가 견고하게 일어나도록 유도하는 것이다. 앞서 소개한 협업 시뮬레이션 환경에서 SISR 기법의 유용성을 검증하기 위해 다음과 같은 실험을 설계하였다. 실험을 통해 지식이 견고하게 전이되었는지를 확인하기 위해, 원본 도메인과 대상 도메인 간의 난이도 격차가 크도록 설계하였다. 원본 도메인에서는 적 개체를 포탑만 두고, 대상 도메인에서는 적 개체에 포탑과 지뢰가 모두 존재하도록 설정하였다. 원본 도메인에서는 방어로봇이 적 포탑과의 교전에서 승리하고 수송로봇이 목적지까지 잘 도착하는 과정을 학습한다. 대상 도메인에서는 새롭게 나타난 지뢰를 개척하는 과정을 학습하는데, 이때 적 포탑과의 교전에서 승리하여야만 살아남은 방어로봇이 지뢰를 개척할 수 있다. 또한, 지뢰는 관측불가능한 개체로서 오로지 경험에 따른 보상 함수의 피드백으로만 지뢰에 대한 정보를 알 수 있다. 즉, 관측할 수 없는 개체를 다룬다는 점에서 지금까지 강화학습 분야에서 제안된 다른 환경들보다 더욱 어렵고, 지뢰에 닿지 않기 위해 시작 지점에 머무르는 국소 최적 지점에 빠질 가능성이 높다. 이러한 어려운 문제 상황에서 SISR 기법의 견고한 지식 전이를 바탕으로 문제 환경을 학습할 수 있는지를 검증하는 것이

실험의 목적이다. 원본 도메인에서 적 포탑 하나만 있는 환경을 1t로, 대상 도메인에서 적 포탑과 지뢰가 모두 있는 환경을 1t1m으로 명명하였다. 또한 이 실험은 1t 환경에서 1t1m 환경으로 지식을 전이했다는 점에서 1t-1t1m 실험으로 명명하였다.

실험에 사용한 하이퍼파라미터 값은 Table 5과 같다. 실험의 승리 조건은 수송기가 목적지에 시간 내에 도달한 경우를 승리한 것으로 간주하였다. 또한 현재 에피소드에서 최근 10000 에피소드 동안의 승리 비율을 모델의 학습 성능으로 정의하였다. 학습에 사용한 장비는 Intel Xeon Silver 4214 CPU, NVIDIA A100-PCIE-40 GB이며 사용된 메모리는 30 GB 내외이다. 학습 시간은 24시간에서 48시간 수준이다.

Table 5. Hyperparameter

Hyperparameter	Value
discount factor	0.99
learning rate	0.001
target update period	200
replay buffer size	3500
minibatch size	32
epsilon annealing	1.0 to 0.1
epsilon-greedy length	20000

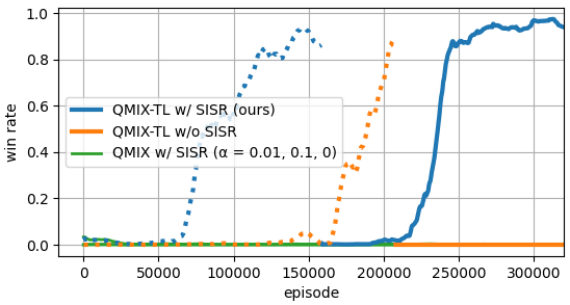

Fig. 6. Win rate for 1t–1t1m experiment

4.3 실험 결과

1t-1t1m 실험에서 SISR 기법을 적용한 전이학습(QMIX-TL w/ SISR)과 비교하기 위해 SISR 기법을 적용하지 않은 전이학습(QMIX-TL w/o SISR)과 SISR 기법을 적용하되 지식 전이 없이 학습한 경우(QMIX w/ SISR)를 비교하였다. QMIX-TL w/ SISR 실험에서 원본 도메인에서 α = 0.1, 대상 도메인에서 α = 0.01이며, QMIX w/ SISR 실험에서는 α 값이 0.1, 0.01일 때와 SISR를 적용하지 않은(즉, α = 0) 경우를 모두 나타내었다. 1t-1t1m 실험의 승률 그래프는 Fig. 6에 나타냈으며, 점선은 원본 도메인에서의 결과를 의미한다. 실험 결과 SISR를 적용한 전이학습은 0.9 이상의 승률을 보였으며, SISR를 적용하지 않은 경우(QMIX-TL w/o SISR)에는 전이 후 환경을 충분히 학습하지 못한 결과를 보였다. 전이학습 없이 SISR 기법만 적용한 경우(QMIX w/ SISR)에는 원본 도메인인 1t 환경에 대해 학습이 이뤄지지 않아 전이조차 일어나지 않은 결과를 보였다.

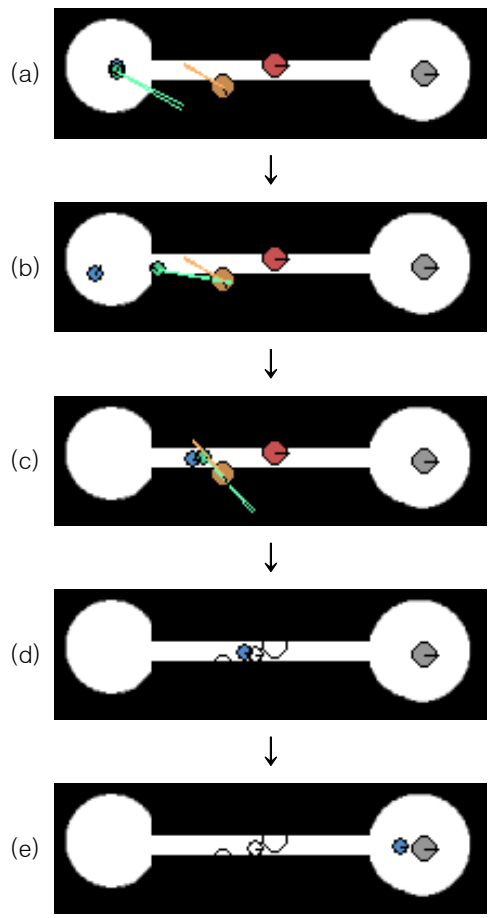

Fig. 7. Behaviors of trained agents by QMIX-TL w/ SISR

다음으로 학습이 완료된 에이전트들의 행동 양상을 살펴보았다. 우수한 성능을 보인 QMIX-TL w/ SISR에 대해 Fig. 7에서 에이전트들의 행동 양상을 시간 순서대로 나타냈다. Fig. 7(a) 시작과 동시에 아군 방어로봇은 포탑을 제거하기 위해 다가가며, 아군 수송기는 제자리에 대기한다. Fig. 7(b) 방어로봇의 타격에 적 포탑이 제거되는 순간에 맞춰 아군 수송기는 이동을 시작하며, Fig. 7(c) 포탑이 파괴되는 것과 동시에 아군 방어로봇과 함께 지뢰로 나아간다. Fig. 7(d) 방어로봇이 적 지뢰를 제거하는 것과 동시에 지뢰 구역을 통과하며, Fig. 7(e) 목적지로 신속하게 나아가 목표를 달성한다. 그에 반해 목표에 도달하지 못했던 QMIX-TL w/o SISR와 QMIX w/ SISR의 경우, 벽에 부딪힌 후 움직이지 않거나 적 포탑에 피격되지 않기 위해 적 포탑의 사거리 경계 주위를 진동하는 양상만 보일 뿐, 적 포탑을 파괴하거나 지뢰를 개척하는 등의 양상을 전혀 보이지 못했다.

Table 6. Experimental results for different algorithms

Experiments	Win Rate (%)
QMIX-TL w/ SISR (ours)	**98.59 ± 1.37**
QMIX-TL w/ policy reuse	0.99 ± 0.12
QMIX-TL	54.28 ± 46.27
QTRAN-TL w/ SISR (ours)	**91.11 ± 21.58**
QTRAN-TL	0.0 ± 0.0

SISR 기법에 대한 포괄적인 검증을 위해 1t 환경에서 1t 환경으로 전이하는 시나리오인 1t-1t 실험을 설계하였다. 해당 시나리오에 대해 QMIX, QTRAN 알고리즘과 SISR 기법을 결합하여 성능을 측정했다. 두 알고리즘 모두 SISR 기법을 적용했을 때에는 90 % 이상의 우수한 성능을 보였지만, SISR 기법을 적용하지 않았을 때는 1 % 미만의 승률을 보였다. 또한, 2.3절에서 설명한 기존의 전이학습 기법과 SISR를 비교하였다. 발표된 전이학습 기법 중에서 본 연구에 적용가능한 policy reuse를 채택했다. Table 6과 같이, policy reuse를 시뮬레이션 환경에 도입한 결과는 54.28 %로, SISR 기법의 성능과는 큰 차이를 보였다. 이러한 결과로 미루어 판단하였을 때, 본 연구에서 제안한 SISR 기법은 멀티에이전트 강화학습에서 전이학습의 성능을 향상시키는 데에 우수한 효과를 가져온다고 판단하였다.

추가로, 본 연구에서 새롭게 제안한 시뮬레이션의 신뢰성을 검증하였다. 4.1절에서 설명한 시뮬레이션 환경이 다양한 실험 설정에 대해서도 일관적인 결과를 도출하는지 검증할 필요가 있다. 시뮬레이션 환경은 강화학습 알고리즘의 성능을 잘 평가하고, 하이퍼파라미터의 변화에도 일관적인 학습 결과를 나타내는 것이 필요하다. 다양한 값의 하이퍼 파라미터와 환경 설정들에 대해서 SISR 기법을 적용한 학습 결과를 비교하는 것으로 시뮬레이션 환경의 우수성을 입증하였다. open space map의 경우, Fig. 8에 나타난 것과 같이 장애물이 없어 모든 영역이 주행가능한 환경 설정이다. slow dynamics mode의 경우 에이전트들의 속도와 각속도를 기존 대비 절반으로 감소시킨 환경 설정이다. Table 7과 같이, 하이퍼파라미터와 환경 설정들을 변경하였을 때도 충분한 수준의 학습 성능을 달성하는 것으로 나타났다. 따라서, 본 연구에서 제안한 시뮬레이션 환경은 충분한 일관성을 갖춘 것으로 판단하였다.

Table 7. Experimental results for consistency

Experiments	Win Rate (%)
discount factor = 0.97	69.59 ± 27.74
buffer size = 5000	83.10 ± 31.77
epsilon-greedy length = 15000	96.48 ± 3.45
open space map	100.0 ± 0.0
slow dynamics mode	98.62 ± 3.34

Fig. 8. Open space simulation environment

4.4 정보량 분석

본 연구에서 제안한 SISR 기법을 통해 에이전트들이 수집한 정보가 다양해지고 정보량이 증가하였다. 본 절에서는 리플레이 메모리에 담긴 에이전트들의

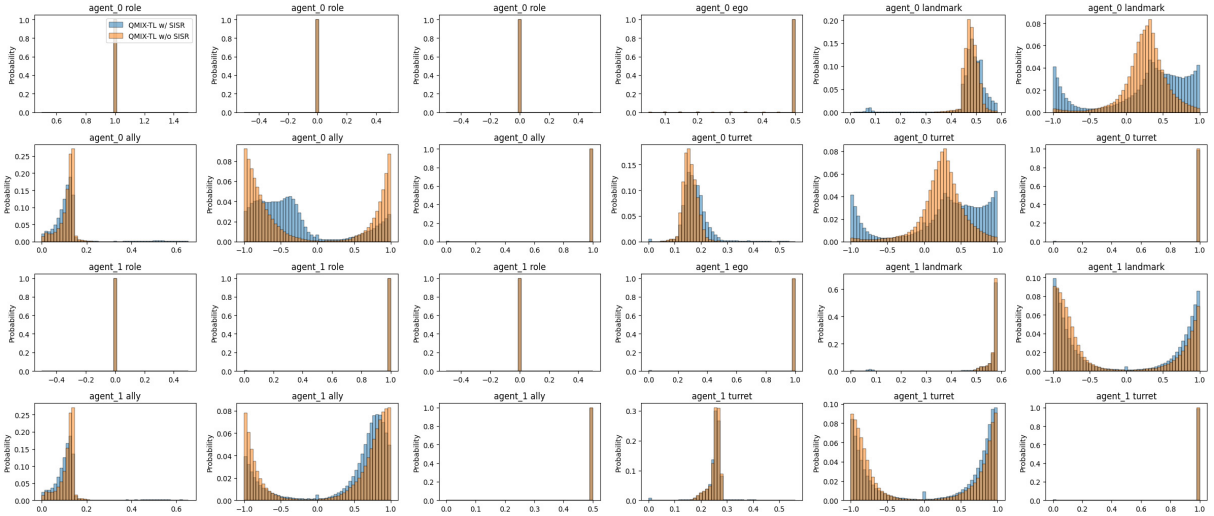

Fig. 9. Histogram of each feature with(blue) and without(orange) SISR

상태 벡터를 분석하는 것으로 실제로 다양성과 정보량이 개선되었는지를 분석하였다. SISR 기법의 목적은 원본 도메인에서 수집할 수 있는 데이터의 다양성을 증가시키는 것이므로, 원본 도메인의 1t 환경에서 초반 3500 에피소드 동안 수집된 65만 개 이상의 상태 벡터에 대해 분석을 진행하였다.

t-SNE는 고차원의 데이터를 낮은 차원에 표현하기 위한 클러스터링 기법을 말한다. 에이전트가 2개일 때를 기준으로 24개의 feature를 가진 상태 벡터들을 2차원으로 압축한 뒤, SISR 기법 적용의 여부에 따라 압축된 분포를 살펴보았다. t-SNE를 적용한 상태 벡터의 분포는 Fig. 10에 나타냈다. SISR를 적용하였을 때, 그렇지 않았을 경우보다 더 넓은 분포를 가진다는 것을 확인할 수 있다. 청색 영역은 황색 영역의 대부분에 분포하고 있는 동시에, 황색 영역이 존재하지 않는 영역까지도 넓게 분포하고 있다. 이로써 SISR 기법을 통해 상태 벡터들이 넓고 다양한 상태 공간을 다루고 있음을 확인할 수 있다.

다음으로, 각 feature 각각의 분포를 확인하기 위해 히스토그램 분석을 진행했다. 각 feature의 최솟값부터 최댓값까지 50개의 구간으로 나눈 후, 각 구간에 포함된 빈도를 확률로써 히스토그램으로 나타냈다. SISR 기법 적용 여부에 대한 상태 벡터 값들의 히스토그램은 Fig. 9에 나타냈다. 여러 feature 중 특히 목표 지점(landmark), 아군 로봇(ally), 적 포탑(turret) 정보의 분포가 더 다양해진 것을 확인할 수 있다.

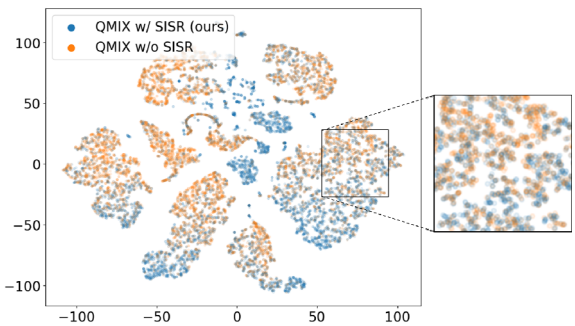

Fig. 10. t-SNE with(blue) and without(orange) SISR

끝으로, SISR 기법의 적용에 따라 각각의 feature가 가지는 평균 정보량이 증가하였는지를 알아보기 위해 엔트로피를 계산하였다. 정보이론에서 말하는 엔트로피란 확률변수에 내재된 평균적인 정보량을 의미하며, 확률변수 X에 대해 식 (6)과 같이 정의된다.

$$H(X) = -\sum_{x} p(x) \log p(x) \tag{6}$$

본 연구에서는 엔트로피를 이용하여 SISR 기법의 유무에 따른 상태 벡터의 평균 정보량 변화를 살펴보았다. 앞서 설명한 히스토그램에서 feature 값이 각 구간에 포함될 확률에 따른 엔트로피를 계산하였으며, 그 값은 Fig. 11과 같다. 또한 전체 feature에 대한 엔트로피의 평균은 QMIX-TL w/ SISR의 경우 1.076이며 QMIX-

TL w/o SISR의 경우 0.974로, SISR 기법의 적용에 따라 약 1.1배의 엔트로피 상승을 보였다. 즉, 초기 상태 분포의 랜덤화를 통해 에이전트들이 수집한 정보들이 유의미하게 늘어났다는 것을 의미한다. 이로써 SISR 기법을 적용하였을 때, 다양하고 일반화된 상태 벡터들을 수집할 수 있으며 이로써 많은 정보량을 수집하는 것으로 견고한 지식 전이를 촉진하였다고 볼 수 있다.

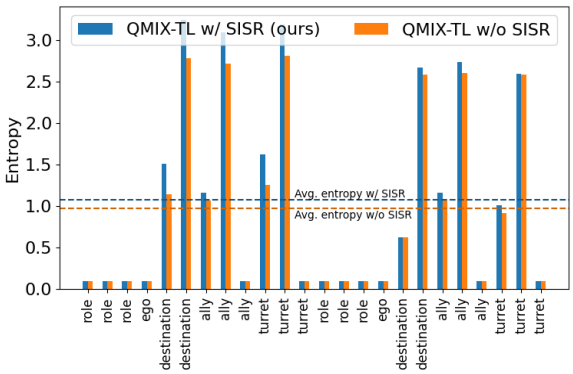

Fig. 11. Entropy of each feature with(blue) and without(orange) SISR

5. Conclusion

본 연구에서는 강화학습에서 전이학습을 적용할 때에 발생할 수 있는 문제를 해결하기 위해 확률적 초기 상태 랜덤화 기법을 제안하였다. 해당 기법을 적용하는 것으로 더 다양하고 일반화된 상태 정보들을 수집할 수 있었으며, 이는 견고한 지식 전이를 가능하게 하였다. 견고한 지식 전이는 기존 전이학습이 풀지 못하던 문제를 풀 수 있었다. 하지만, 다양하고 일반화된 상태 정보와 에이전트의 학습 성능 간의 상관관계를 명시적으로 나타내지 못한 한계점이 존재한다. 추후 학습 데이터의 다양성과 견고한 지식 전이 사이의 상관관계에 관한 연구를 이어나가 이러한 한계점을 보완하고자 한다.

후 기

이 논문은 2024년 정부의 재원으로 수행된 연구 결과임.

References

[1] I. Hwang and J. Bae, “Two Circle-based Aircraft Head-on Reinforcement Learning Technique using Curriculum,” Journal of the Korea Institute of Military Science and Technology, Vol. 26, No. 4, pp. 352-360, 2023.

[2] S. Yi, K. Kim, and S. Yoon, “Study on Enhancing Training Efficiency of MARL for Swarm Using Transfer Learning,” Journal of the Korea Institute of Military Science and Technology, Vol. 26, No. 4, pp. 361-370, 2023.

[3] M. Samvelyan, T. Rashid, C. S. De Witt, G. Farquhar, N. Nardelli, T. GJ Hung, C.-M. Hung, P. HS Torr, J. Foerster, and S. Whiteson, “The StarCraft Multi-Agent Challenge,” arXiv preprint arXiv:1902.04043, 2019.

[4] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, “Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments,” Advances in Neural Information Processing Systems, pp. 6379-6390, 2017.

[5] K. Kurach, A. Raichuk, P. Stańczyk, M. Zając, O. Bachem, L. Espeholt, C. Riquelme, D. Vincent, M. Michalski, O. Bousquet, and S. Gelly, “Google Research Football: A Novel Reinforcement Learning Environment,” Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, No. 4, pp. 4501-4510, 2020.

[6] P. Sunehag, G. Lever, A. Gruslys, W. M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Z. Leibo, K. Tuyls, and T. Graepel, “Value-Decomposition Networks For Cooperative Multi-Agent Learning,” arXiv preprint arXiv:1706.05296, 2017.

[7] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, “Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning,” Journal of Machine Learning Research, Vol. 21, No. 178, pp. 1-51, 2020.

[8] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing Atari with Deep Reinforcement Learning,” arXiv preprint arXiv:1312.5602, 2013.

[9] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, “Transfer Learning in Deep Reinforcement Learning: A Survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 45, No. 11, pp. 13344-13362, 2023.

[10] J. Ho and S. Ermon, “Generative Adversarial Imitation Learning,” Advances in Neural Information Processing Systems, pp. 4565-4573, 2016.

[11] M. Vecerik, T. Hester, J. Scholz, F. Wang, O. Pietquin, B. Piot, N. Heess, T. Rothörl, T. Lampe, and M. Riedmiller, “Leveraging Demonstrations for Deep Reinforcement Learning on Robotics Problems with Sparse Rewards,” arXiv preprint arXiv:1707.08817, 2018.

[12] G. Hinton, O. Vinyals, and J. Dean, “Distilling the Knowledge in a Neural Network,” arXiv preprint arXiv:1503.02531, 2015.

[13] F. Fernández and M. Veloso, “Probabilistic policy reuse in a reinforcement learning agent,” Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 720-727, 2006.

[14] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, “Progressive Neural Networks,” arXiv preprint arXiv:1606.04671, 2022.

[15] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, and D. Wierstra, “PathNet: Evolution Channels Gradient Descent in Super Neural Networks,” arXiv preprint arXiv:1701.08734, 2017.

[16] W. Wang, T. Yang, Y. Liu, J. Hao, X. Hao, Y. Hu, Y. Chen, C. Fan, and Y. Gao, “From Few to More: Large-Scale Dynamic Multiagent Curriculum Learning,” Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, No. 5, pp. 7293-7300, 2020.

[17] S. Hu, F. Zhu, X. Chang, and X. Liang, “Updet: Universal Multi-Agent Reinforcement Learning via Policy Decoupling with Transformers,” arXiv preprint arXiv:2101.08001, 2021.

[18] T. Zhou, F. Zhang, K. Shao, Z. Dai, K. Li, W. Huang, W. Wang, B. Wang, D. Li, W. Liu, and others, “Cooperative Multi-Agent Transfer Learning with Coalition Pattern Decomposition,” IEEE Transactions on Games, Vol. 16, No. 2, pp. 352-364, 2024.

[19] W. Zeng, J. Campbell, S. Stepputtis, and K. Sycara, “Multi-Agent Transfer Learning via Temporal Contrastive Learning,” arXiv preprint arXiv:2406.01377, 2024.

[20] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, “Transfer Learning in Deep Reinforcement Learning: A Survey,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 45, No. 11, pp. 13344-13362, 2023.