효율적인 관계형 강화학습을 위한 사전 영역 지식의 활용

강민교, 김인철
경기대학교 컴퓨터과학과
email:alsry5786@kyonggi.ac.kr, kic@kyonggi.ac.kr

Using Prior Domain Knowledge for Efficient Relational Reinforcement Learning

Minkyo Kang, Incheol Kim
Department of Computer Science, Kyonggi University

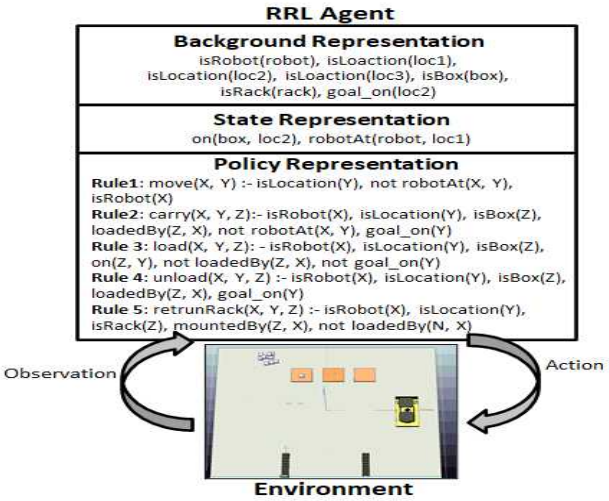
요 약

기존의 심층 강화학습은 상태, 행동, 정책 등을 모두 벡터 형태로 표현하는 강화학습으로서, 학습된 정책의 일반성과 해석 가능성에 제한이 있고 영역 지식을 학습에 효과적으로 활용하기도 어렵다는 한계성이 있다. 이러한 문제점들을 해결하기 위해 제안된 새로운 관계형 강화학습 프레임워크인 dNL-RRL은 상태, 행동, 그리고 학습된 정책을 모두 논리 서술자와 규칙들로 표현할 수 있다. 본 논문에서는 dNL-RRL을 기초로 공장 내 운송용 모바일 로봇의 제어를 위한 행동 정책 학습을 수행하였으며, 학습의 효율성 향상을 위해 인간 전문가의 사전 영역 지식을 활용하는 방안들을 제안한다. 다양한 실험들을 통해, 본 논문에서 제안하는 영역 지식을 활용한 관계형 강화학습 방법의 학습 성능 개선 효과를 입증한다.

1. 서론

최근 들어 신경망(neural network)을 이용하는 심층 강화학습(deep reinforcement learning) 기술은 Atari, Starcraft 등과 같은 컴퓨터 게임 분야뿐만 아니라, 자율주행 자동차, 지능형 서비스 로봇 분야에 이르기까지 폭넓게 활용되고 있다. 하지만 기존의 심층 강화학습은 상태(state), 행동(action), 정책(policy) 등을 모두 벡터 형태로 표현하는 강화학습으로서, 학습된 정책의 일반성(generality)과 해석 가능성(interpretability)에 제한이 있고 영역 지식(domain knowledge)을 학습에 효과적으로 활용하기도 어렵다는 한계성이 있다. 이러한 문제점들을 극복하고자, 상태, 행동, 정책 등을 논리 서술자(logic predicate) 기반의 논리식(logic expression)으로 표현하는 NLM(Neural Logic Machine)[1], RDRL(Relational Deep Reinforcement Learning)[2], NLRL(Neural Logic Reinforcement Learning)[3], dNL-RRL(differentiable Neural Logic-Relational Reinforcement Learning)[4] 등과 같은 다양한 관계형 강화학습(relational reinforcement learning) 방법들이 제안되었다. 이들 중 특히 dNL-RRL 관계형 강화학습 프레임워크는 미분 가능한 귀납적 논리 프로그래밍(differentiable Inductive Logic Programming)[5] 엔진을 채용함으로써, 강화학습 프레임워크 전체에 대해 종단간 학습(end-to-end training)이 가능하고, 비-기호 형태인 입력 영상과 센서 데이터, 그리고 출력 제어 신호도 처리 가능하다는 장점이 있다. 따라서 다양한 응용 분야들에서 활용 가능성이 매우 높다. <그림 1>은 dNL-RRL 관계형 강화학습 프레임워크에서 표현할 수 있는 배경 지식(background knowledge), 상태(state), 학습되는 행동 정책(action policy) 등의 예시를 나타낸다. <그림 1>은 본 논문에서 다루는 응용 분야인 제조 공장 환경에서 동작하는 운송용 모바일 로봇의 행동 정책을 학습하는 예를 보여준다. <그림 1>의 예에서 보듯이, dNL-RRL과 같은 관계형 강화학습 프레임워크에서는 배경 지식과 변화하는 상태는 모두 논리 서술자(logic predicate)들의 집합으로 표현되고, 학습되는 행동 정책은 각 행동에 관한 논리 규칙(logic rule)들로 표현된다. 예컨대, 행동 move(X, Y)에 관한 정책은 isLocation(Y), isRobot(X), not robotAt(X, Y)와 같이 변수들을 포함하는 논리 서술자들을 조건부로 갖는 하나의 규칙으로 표현된다. 따라서 이와 같은 논리 규칙 형태의 정책들은 매우 높은 일반성과 해석 가능성을 갖는다.

* 본 연구는 정보통신기획평가원의 재원으로 정보통신방송 기술개발사업의 지원을 받아 수행한 연구 과제(클라우드에 연결된 개별 로봇 및 로봇그룹의 작업 계획 기술 개발, 2020-0-00096)입니다.



<그림 1> 관계형 강화학습의 예시

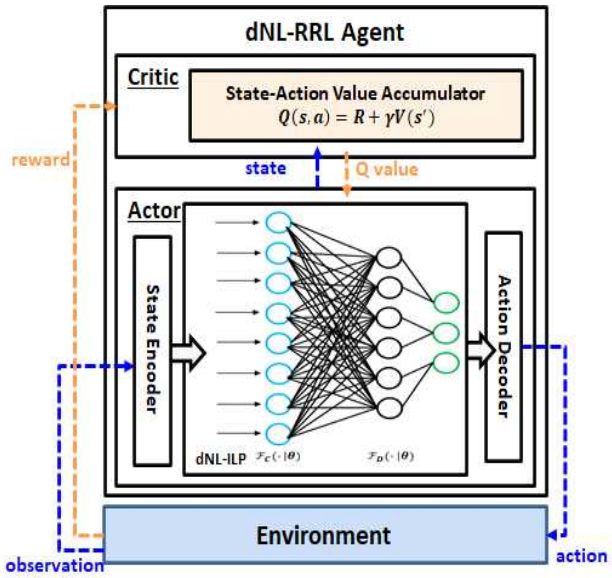
본 논문에서는 대표적인 관계형 강화학습 프레임워크인 dNL-RRL을 기초로 공장 내 물류 로봇의 제어를 위한 행동 정책 학습을 수행하였으며, 학습의 효율성 향상을 위해 인간 전문가(human expert)의 사전 영역 지식(prior domain knowledge)을 활용하는 방안들을 제안한다. 관계형 강화학습 프레임워크에서는 다양한 방법으로 인간의 영역 지식을 활용할 수 있다. 본 논문에서는 (1) 학습하고자 하는 각 행동 규칙(action rule)에 반드시 포함되어야 하는 상태 조건들(included conditions)과 반드시 배제되어야 하는 상태 조건들(excluded conditions)을 미리 정의해 주는 방식, (2) 학습하고자 하는 행동 규칙들의 일부를 미

리 정의해주는 방식(predefined rules) 등의 2 가지 사전 영역 지식 활용 방법들을 제안한다. CopelliaSim 로봇 시뮬레이터로 구현한 가상의 공장 물류 환경과 운송용 모바일 로봇을 이용한 다양한 실험들을 통해, 본 논문에서 제안하는 영역 지식 기반의 관계형 강화학습 방법의 학습 성능 개선 효과 및 정책의 일반성(generality), 해석 가능성(interpretability) 등을 분석해본다.

2 사전 영역 지식 기반 관계형 강화학습

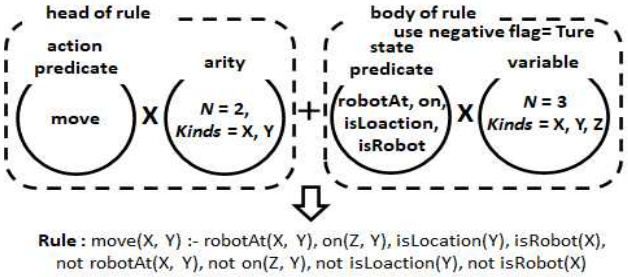
2.1 관계형 강화학습 프레임워크

본 논문에서는 대표적인 관계형 강화학습 프레임워크인 dNL-RRL을 기초로, 사전 영역 지식 활용 방법들을 제안한다. dNL-RRL 관계형 강화학습 프레임워크는 <그림 2>와 같이, 행동자-비평가(actor-critic) 구조를 따른다. 행동자(actor)는 환경에서의 경험 데이터들을 토대로 행동 정책을 구성하는 규칙들을 학습(policy learning)하고, 이 학습된 행동 규칙들에 따라 추론함(policy inference)으로써 매 순간 에이전트가 실행해야 할 행동을 결정(action decision)하는 역할을 수행한다. 반면에, 비평자(critic)는 경험 데이터와 보상(reward)을 토대로 행동 가치 함수 Q를 학습(value function learning)하고, 이를 토대로 행동자가 결정한 행동들을 평가해줌으로써, 행동자가 신속하게 정책을 새롭게 갱신하는데 도움을 주는 역할을 수행한다.



<그림 2> 관계형 강화학습 프레임워크의 구조

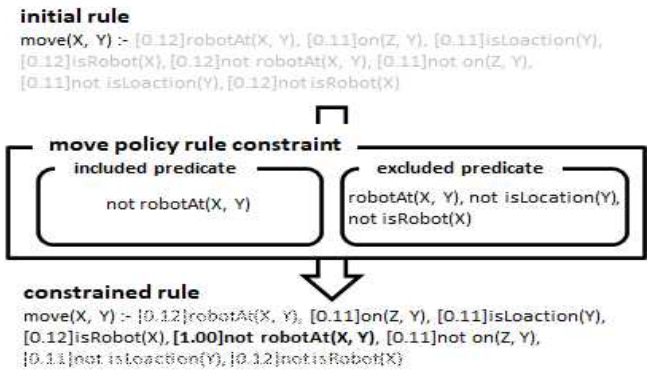
한편, 정책 학습과 행동 결정을 담당하는 행동자는 다시 상태 인코더(state encoder), 미분 가능한 뉴로-로직 귀납적 논리 프로그래밍 엔진(differentiable Neural-Logic Inductive Logic Programming, dNL-ILP), 행동 디코더(action decoder)들로 구성된다. 상태 인코더는 환경으로부터 관측 데이터(observation)를 입력받아, 해당 상태(state)에서 만족되는 논리 서술자들(logic predicates)을 생성함으로써 하나의 상태를 일차-술어논리(first-order logic) 형태로 표현하는 역할을 담당한다. 미분 가능한 뉴로-로직 귀납적 논리 프로그래밍 엔진(dNL-ILP)은 논리 서술자 기반의 상태 표현과 이전에 실행한 행동에 관한 평가자(critic)의 가치 평가를 토대로 행동 규칙들을 학습하기도 하고, 학습된 행동 규칙들을 토대로 n 단계의 전향 추론(n-step forward chaining)을 통해 다음에 실행할 행동을 결정하기도 한다. 행동 디코더는 실행이 결정된 서술자 형태의 행동을 실제 환경 안에서 실현되도록 하는 제어 함수의 역할을 수행한다. 한편, 뉴로-로직 귀납적 논리 프로그래밍 엔진(dNL-ILP)에서는 모든 일차-술어논리식을 곱의 합 표준형(Disjunctive Normal Form, DNF)으로 변환하여 표현한다. 따라서 하나의 행동 결론부(action head)와 여러 상태 조건들(state conditions)로 구성된 각각의 행동 규칙도 모두 곱의 합 표준형(DNF)으로 표현한다. 그리고 이러한 임의의 곱의 합 표준형(DNF) 논리식을 논리곱(logical conjunction, F_C) 계층들과 논리합(logical disjunction, F_D) 계층들로 구성된 하나의 인공 신경망으로 학습한다. 따라서 행동 규칙들을 논리곱 계층들과 논리합 계층들로 구성된 하나의 신경망으로 학습하기 위해서, 먼저 학습할 행동 규칙의 구성을 나타내는 행동 규칙 템플릿(policy rule template)을 요구한다. <그림 3>은 행동 move(X, Y)에 관한 행동 규칙 생성 템플릿을 예시하고 있다. 이 템플릿에 의하면 각 행동 규칙은 크게 행동 결론부(head)와 상태 조건부(body)로 구성되며, 규칙의 결론부는 다시 move 행동 서술자(action predicate)와 2개의 인자들로 구성되고 인자의 종류(kind)는 X와 Y이다. 또 규칙의 조건부는 robotAt, on, isLocation, isRobot 등의 상태 서술자(state predicate)들이 사용될 수 있으며, 종류가 X, Y, Z인 3 개의 변수(variable)들을 포함할 수 있다. 따라서 이러한 규칙 생성 템플릿에 따라, <그림 3>의 하단에 표시된 행동 move(X, Y)에 관한 행동 규칙이 학습될 수 있다.



<그림 3> 행동 규칙 생성 템플릿의 예시

2.2 행동 규칙 조건 제약

앞 절에서 설명한대로, 관계형 강화학습 프레임워크인 dNL-RRL에서는 행동 규칙들을 학습하기 위해 미리 정의해둔 규칙 생성 템플릿에 따라 가능한 모든 규칙들을 생성한 후, 행동가-비평가(actor-critic) 기반의 심층 강화 학습을 통해 이들을 표현하는 신경망의 가중치(weight)들을 학습한다. 일반적으로 이러한 템플릿에 의해 생성 가능한 후보 행동 규칙들(candidate action rules)의 수는 후속 학습 과정에 큰 부담을 줄 정도로 매우 많다. 그중에는 현실 환경에서는 동시에 만족하기 어려운 상태 조건식들이 포함된 규칙들이나 무의미한 규칙들도 다수 포함되어 있다. 충분한 학습이 이루어진 후에는 불가능한 상태 조건식들의 조합이 해소될 수도 있으나, 그러기 위해서는 많은 계산 자원이 소모될 수 있다. 미리 정의된 규칙 템플릿을 이용하는 것외에, 해당 영역에 관한 인간 전문가의 사전 지식(prior knowledge)을 활용해 의미있는 후보 행동 규칙들만 생성될 수 있도록 미리 제한할 수 있다면, 이러한 문제들을 해결하고 학습 효율성을 크게 향상시킬 수 있을 것이다.

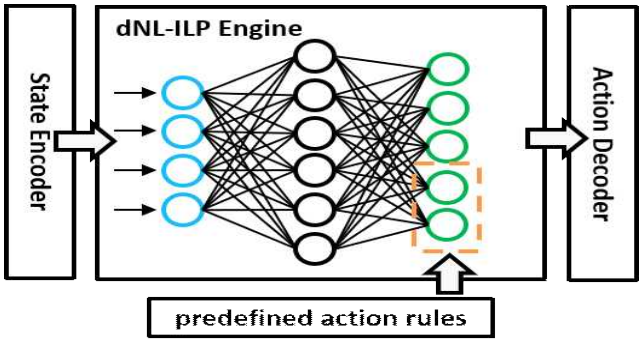


<그림 4> 행동 규칙 조건 제약의 예시

관계형 강화학습 프레임워크인 dNL-RRL에서 후보 행동 규칙들을 생성할 때 영역 지식을 활용할 수 있는 방법 중의 하나는 행동 규칙의 조건 제약(condition constraint)들을 이용하는 것이다. 행동 규칙 조건 제약에는 두 가지 종류가 있다. 첫째는 조건 포함 제약(condition inclusion constraint)으로서, 행동 규칙의 조건부에 반드시 특정 상태 조건식(state condition)들을 포함하도록 하는 제약이다. 둘째는 조건 배제 제약(condition exclusion constraint)으로서, 행동 규칙의 조건부에 반드시 특정 상태 조건식들은 포함되지 않도록 배제하는 제약이다. <그림 4>는 행동 move(X, Y)를 위한 행동 규칙 학습에 조건 포함 제약과 조건 배제 제약을 적용하는 예를 보여주고 있다. <그림 4>의 상단에는 <그림 3>과 같이 규칙 생성 템플릿에 따라 생성된 행동 move(X, Y)의 초기 행동 규칙(initial action rule)을 나타내며, 여기에 조건 포함 제약인 not robotAt(X, Y)와 조건 배제 제약들인 robotAt(X, Y), not isLocation(Y), not isRobot(X) 등이 적용되어 <그림 4>의 하단과 같이 정제된 행동 규칙(constrained action rule)이 생성된다.

2.3 사전 정의된 행동 규칙의 활용

인간 전문가의 사전 영역 지식을 활용해 관계형 강화학습 프레임워크인 dNL-RRL의 학습 효율성을 향상시킬 수 있는 또 다른 방법은 이미 잘 알려져 있는 행동 행동 규칙들의 일부를 정책 학습이 본격적으로 시작되기 이전에 미리 제공하는 방법이다. 예를 들어, <그림 1>과 같은 공장 내 운송용 모바일 로봇의 행동 정책은 move, carry, load, unload, returnRack 등 총 5가지 행동 각각에 관한 규칙들로 구성된다. 따라서 이 로봇의 경우 완전한 행동 정책을 습득하려면 이 5가지 행동 행동 규칙들을 모두 학습해야 한다. 하지만 이 규칙들 중 이미 잘 알려져 있는 것들이 존재한다면, 이들까지 처음부터 학습할 필요가 없이 다른 행동들에 관한 행동 규칙 학습이 시작되기 전에 미리 해동 정책을 구성하는 행동 규칙 집합에 포함시켜줄 수 있을 것이다. 예컨대, 위의 5가지 행동들 중 사전 영역 지식을 토대로 행동 load(X, Y, Z)에 관한 행동 규칙을 load(X, Y, Z):- isRobot(X), isLocation(Y), isBox(Z), robotAt(X, Y), on(Z, Y)와 같이 미리 정의해줄 수 있다면, 학습은 나머지 4가지 행동들에 관한 행동 규칙들에만 집중할 수 있을 것이다. 이와 같이 사전에 정의된 행동 규칙(predefined action rule)들을 활용한다면 꼭 필요한 행동 규칙들에만 학습을 집중함으로써 행동 정책을 학습하는데 소요되는 시간과 자원을 크게 절약할 수 있으며, 보다 실효성이 있는 행동 정책을 습득하기에도 도움이 될 것이다.



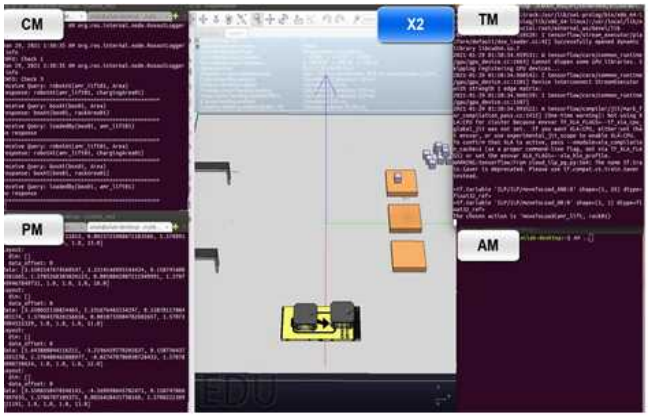
<그림 5> 사전 정의된 행위 규칙들의 활용

사전에 정의된 행동 규칙들은 학습을 통해 습득되는 나머지 행동 규칙들과 함께 통합적으로 하나의 행동 정책(policy)을 형성하여, 입력 상태에 대처할 수 있는 로봇의 행동을 결정하는데 이용된다. <그림 5>는 이와 같이 사전에 정의된 행동 규칙들과 학습된 행동 규칙들이 뉴로-로직 귀납적 논리 프로그래밍 엔진(dNL-ILP) 내부에서 행동 결정에 이용되는 과정을 나타낸다.

3. 구현 및 실험

3.1 실험 환경 및 모델 학습

본 논문에서 제안하는 관계형 강화학습의 사전 영역 지식 활용 방법의 효과를 평가를 위해, CopelliaSim 로봇 시뮬레이터를 이용해 <그림 6>과 같은 공장 내 운송용 모바일 로봇을 포함한 실험 환경을 구축하였다. 모바일 로봇의 전체적인 인식과 행동 제어는 로봇 지능 구조를 구성하는 PM(Perception Manager), CM(Context Manager), TM(Task Manager), AM(Action Manager) 모듈들 간의 실시간 메시지 기반의 상호작용에 의해 구현된다. 이 모듈들 중 PM은 환경으로부터 인식 정보를 취득하여 CM에게 전달하고, CM은 인식 정보들을 바탕으로 상태 서술자들로 구성된 논리적 상태 묘사를 생성하여 TM에게 전달한다. 이 상태 묘사를 토대로 TM은 관계형 강화학습 프레임워크가 학습한 행동 정책에 의해 실행할 행동을 결정하여 AM에게 전달한다. AM은 해당 로봇의 행동을 환경 내에서 실제로 실행하는 역할을 수행한다. 특히 본 논문에서 다루는 관계형 강화학습 프레임워크는 로봇의 실시간 행동 결정을 담당하는 TM 내부에서 이용된다.

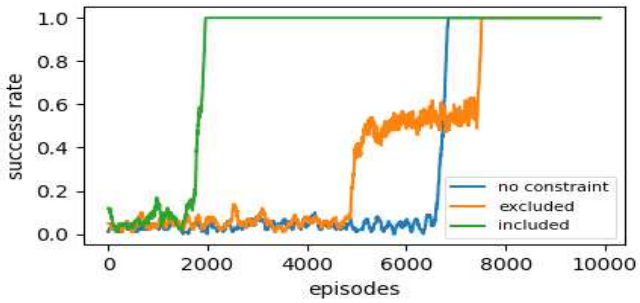


<그림 6> 운송용 모바일 로봇을 포함한 실험 환경

정책 학습과 테스트에 이용된 로봇 작업은 로봇이 상품을 정해진 목표 지점까지 운송하는 작업이다. 초기 상태는 로봇이 충전 위치에 있고 상품 박스는 선반 위에 올려진 상태로서 on(box, loc1), robotAt(robot, loc3)와 같이 표현되고, 목표 상태는 상품 박스는 컨베이어 벨트로 옮겨지고 로봇은 다시 충전 위치로 복귀하는 상태로서 on(box, loc2), robotAt(robot, loc3)와 같이 술어 논리식으로 표현된다. 관계형 강화학습 프레임워크의 학습을 위해 최적화 알고리즘(optimizer)은 Adam Optimizer를 사용하였으며. 학습률(learning rate)은 0,02, 비평가의 감가율(γ)은 0.9로 설정하였다. 관계형 강화학습 프레임워크의 학습과 성능 실험들은 Geforce RTX 1080ti GPU, 128GB RAM 하드웨어 환경, Ubuntu 18.04, Python 3.6, Tensorflow 1.15의 소프트웨어 환경에서 수행하였다.

3.2 성능 평가 실험

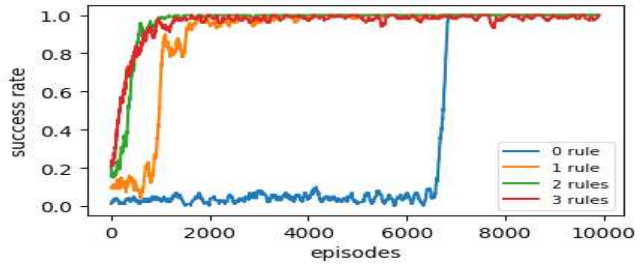
첫 번째 실험은 행동 규칙 조건 제약으로 표현되는 사전 영역 지식이 학습 성능에 미치는 효과를 분석하기 위한 실험이다. 이 실험에서는 (1) 행동 규칙 조건 제약이 적용되지 않은 경우(no constraint), (2) 조건 배제 제약이 적용된 경우(excluded constraint) (load와 unload의 행동 규칙 조건부에 mountedBy와 isRack 조건들을 배제), (3) 조건 포함 제약이 적용된 경우(included constraint) (move와 carry의 행동 규칙 조건부에는 not robotAt(X, Y) 조건을, load, unload, retrunRack의 행동 규칙 조건부에는 robotAt(X, Y) 조건을 반드시 포함) 등 총 3가지 서로 다른 경우들을 비교하였다. 이 실험에서는 총 100 에피소드의 평균 작업 성공률을 성능 척도로 사용하였다. 실험 결과는 <그림 7>과 같다.



<그림 7> 규칙 조건 제약에 따른 성능

<그림 7>에서 볼 수 있듯이, 행동 규칙 조건 제약이 적용된 경우들이 그렇지 않은 경우에 비해 더 빠른 성능 향상을 보였다. 특히 본 실험에서는 조건 배제 제약을 적용한 경우에 비해 조건 포함 제약을 적용한 경우가 더 빠른 성능 향상을 보였다. 본 실험 영역에서는 템플릿에 따라 생성되는 행동 규칙들에 불필요한 조건들이 많이 포함되지 않아서 그런 결과가 나타난 것으로 판단된다. <그림 7>의 실험 결과를 토대로, 본 논문에서 제안한 바와 같이 행동 규칙 조건 제약으로 주어지는 사전 영역 지식이 관계형 강화학습의 학습 효율성과 작업 성공률을 향상시키는데 큰 효과가 있음을 확인할 수 있었다.

두 번째 실험은 사전 정의된 행동 규칙들로 표현되는 사전 영역 지식이 학습 성능에 미치는 효과를 분석하기 위한 실험이다. 이 실험에서는 (1) 사전 정의된 행동 규칙을 활용하지 않는 경우(0 rule), (2) 사전 정의된 carry 행동 규칙을 활용하는 경우(1 rule), (3) 사전 정의된 carry, load 행동 규칙들을 활용하는 경우(2 rules), (4) 사전 정의된 carry, load, unload 행동 규칙들을 활용하는 경우(3 rules)들을 서로 비교하였다. 총 100 에피소드의 평균 작업 성공률을 성능 척도로 사용하였으며, 실험 결과는 <그림 8>과 같다.



<그림 8> 사전 정의된 규칙 활용에 따른 성능

<그림 8>에서 볼 수 있듯이, 사전 정의된 규칙들을 많이 활용할수록, 더 빠른 성능 향상을 보였다. 반면에, 사전 정의된 규칙들을 활용하지 않은 경우가 가장 동일한 성공률에 도달하기 위해서는 가장 기 학습 시간을 요구했다. 이와 같은 결과를 토대로, 본 논문에서 제안한 바와 같이 사전 정의된 행동 규칙 형태로 영역 지식을 활용하는 것이 관계형 강화학습의 학습 효율성과 작업 성공률 향상에 도움을 줄 수 있음을 확인할 수 있었다.

세 번째 실험은 관계형 강화학습 프레임워크를 통해 학습되는 행동 정책의 해석 가능성(interpretability)과 일반성(generality)을 확인해보기 위한 실험이다. 관계형 강화학습 프레임워크가 운송용 모바일 로봇을 위해 학습한 행동 정책은 <표 1>과 같다. 학습된 행동 규칙들은 논리 서술자들로 표현되어 의미 해석이 용이하며, <그림 1>에 기술된 인간 전문가에 의해 정의된 행동 규칙들과 높은 일치성을 보였다. 다음은 관계형 강화학습 프레임워크를 통해 학습되는 행동 정책의 일반성을 확인해보기 위한 실험을 수행하였다. 이 실험에서는 행동 정책 학습을 위한 작업들의 초기 상태들과는 다른 테스트 작업들의 초기 상태들을 가정하고, 이와 같이 학습용 작업(training task)들과는 다른 테스트 작업(test task)들에 대한 행동 정책의 작업 성공률을 비교해보았다. 학습용 작업들의 초기 상태(initial state)는 <표 2>와 같이 로봇의 초기 위치의 X 좌표는 [2.8, 3.2] 범위 안에서, Y 좌표는 [1.8, 2.2] 범위 안에서 각각 임의의 실수값으로 선택하여 결정하였다. 그리고 <표 1>과 같은 행동 정책들이 학습된 후에는, 행동 정책의 성능 테스트를 위해 학습용 작업들의 로봇 초기 위치와는 다른 범위의 로봇 초기 위치들을 임의로 선택하여 테스트 작업군들(test set-1, 2, 3, 4)을 결정하였다.

<표 1> 학습된 행동 규칙들

head of rule	body of rule
move(X, Y)	isLocation(Y), not robotAt(X, Y), not loadedBy(Z, X), not goal_on(Y)
	not on(Z, Y), not goal_on(Y), not robotAt(X, Y), isLocation(Y), not loadedBy(Z, X)
load(X, Y, Z)	on(Z, Y), isLocation(Y), not goal_on(Y), not loadedBy(Z, X), robotAt(X, Y)
unload(X, Y, Z)	loadedBy(Z, X), not on(Z, Y), goal_on(Y), isLocation(Y)
carry(X, Y, Z)	loadedBy(Z, X), not robotat(X, Y), goal_on(Y), isLocation(Y)
returnRack(X, Y, Z)	robotAt(X. Y), mountedBy(Z, X), not loadedBy(N, X), not goal_on(Y)

<표 2> 정책의 일반성 실험 결과

initial state			success rate
robot location	X-axis	Y-axis	
train set	[2.8, 3.2]	[1.8, 2.2]]	1.00
test set-1	[1.5, 4.5]	[0.5, 3.5]	0.53
test set-2	[1.0, 5.0]	[0, 4.0]	0.27
test set-3	[0.5, 5.5]	[-0.5, 4.5]	0.17
test set-4	[4.0, 8.0]	[3.0, 7.0]	0.00

<표 2>의 실험 결과를 살펴보면, 학습용 작업들의 초기 위치와 동일한 작업들(train set)에 대해서는 학습된 행동 정책은 100%의 작업 성공률을 보였다. 그러나 테스트 작업군들의 초기 위치의 범위가 학습용 작업들의 초기 위치 범위와의 차이가 커질수록, 작업 성공률은 점차 낮아지는 결과를 보이고 있다. 하지만 test set-1과 test set-2와 같이 학습용 작업들과는 비교적 차이가 있는 테스트 작업군들의 경우에도 각각 53%, 27% 정도의 작업 성공률을 보여주었다. 이러한 실험 결과를 토대로, 관계형 강화학습 프레임워크를 통해 학습되는 행동 정책의 높은 일반성을 확인할 수 있었다.

4. 결 론

본 논문에서는 대표적인 관계형 강화학습 프레임워크인 dNL-RRL을 기초로 공장 내 운송용 모바일 로봇의 제어를 위한 행동 정책 학습을 수행하였으며, 학습의 효율성 향상을 위해 인간 전문가의 사전 영역 지식을 활용하는 방안들을 제안하였다. 다양한 실험들을 통해, 본 논문에서 제안하는 영역 지식을 활용한 관계형 강화학습 방법의 학습 성능 개선 효과를 입증하였다.

참고 문헌

[1] H. Dong et al., "Neural Logic Machines," *arXiv preprint arXiv:1904.11694*, 2019.
[2] V. Zambaldi, et al., "Relational Deep Reinforcement Learning," *arXiv preprint arxiv:1806.01830*, 2018.
[3] Z. Jiang and S. Luo, "Neural Logic Reinforcement Learning," *arXiv preprint arXiv:1904.10729*, 2019.
[4] A. Payani and F. Fekri, "Incorporating Relational Background Knowledge into Reinforcement Learning via Differentiable Inductive Logic Programming," *arXiv preprint arXiv:2003.10386*, 2020.
[5] A. Payani and F. Fekri, "Inductive Logic Programming via Differentiable Deep Neural Logic Networks," *arXiv preprint arXiv:1906.03523*, 2019.