강화학습을 이용한 회전식 도립진자 시스템 설계

김주봉*, 권도형*, 홍용근**, 김민석**, 한연희*+
*한국기술교육대학교 컴퓨터공학과
**한국전자통신연구원
e-mail: jubong1992@gmail.com

Design of Rotary Inverted Pendulum System Using Reinforcement Learning

Ju-Bong Kim*, Do-Hyung Kwon*, Yong-Geun Hong**, Min-Suk Kim**, Youn-Hee Han*
*Dept of Computer Science, Koreatech University
**Electronics and Telecommunications Research Institute

요 약

Rotary Inverted Pendulum 은 제어분야에서 비선형 제어 시스템을 설명하기 위해 자주 사용되어왔다. 본 논문은 강화학습 에이전트의 환경으로써 Rotary Inverted Pendulum 을 도입하였다. 이를 통해서 강화학습이 실제 세계에서의 복합적인 문제를 해결할 수 있음을 보인다. 강화학습 에이전트의 가상 환경과 실제 환경을 맵핑시키기 위해서 Ethernet 연결 위에 MQTT 프로토콜을 사용하였으며 이를 통해서 경량화된 IoT 분야에서의 강화학습의 활용도를 조명한다.

1. 서론

최근 ICT 분야의 발전이 4차 산업혁명을 일으켰고, 그 중심에는 머신러닝이 있었다. 머신러닝 내에서도 강화학습은 동물의 행동학적 요소에 관점을 두어 에이전트의 행동 양식을 발전시켜 나감으로써 환경에서의 움직임을 최적화 시킨다. 그러나 강화학습의 이러한 특성에도 불구하고 현실 시스템에서의 활용보다는 아케이드 게임 시뮬레이션에서의 활용이 압도적으로 많다. 그 이유는 현실세계의 비선형적 복잡함 때문에 에이전트가 어려운 작업에 직면하기 때문이다. 에이전트는 고차원적 환경 상태를 파악하여 이해해야만 효율적 행동이 가능해진다. 이를 위해서 강화학습 에이전트는 과거의 경험을 새로운 경험으로 일반화시켜 입력과 행동사이의 연결을 끊어내는 작업을 통해서 어떠한 복잡한 환경에서라도 학습이 가능하도록 만든다[1, 2].

본 논문은 강화학습을 적용한 Rotary Inverted Pendulum 시스템을 설계 하였다. Rotary Inverted Pendulum은 초기 상태가 매우 불안정한 시스템이고 비선형적 특성 때문에 비선형 제어 분야를 설명하는데 사용되어왔다[3]. 매우 불안정한 초기 상태를 안정적으로 제어하기 위해서 복잡한 수학적 제어 모델인 LQR[4], PID[4] 등이 필수적으로 사용되었고, 성공적인 제어를 위해서는 초기 상태와 상수 값들의 설정이 상당히 중요하기 때문에 제어 분야에 깊은 이해가 없다면 매우 어려운 상황에 직면할 수밖에 없다[5]. 하지만 앞서 언급한 복잡한 제어 모델을 강화학습 에이전트가 대체한다면 설계자의 역량에 상관없이 목표하고자 하는 바를 효과적으로 이루어 낼 수 있다.

실제 Physical 시스템에 해당하는 Rotary Inverted Pendulum 은 $Quenser^{TM}$ 사의 QUBE-Servo2 제품을 사용하였다. Pendulum과 Motor의 위치 및 속도에 대한 피드백을 제공하는 광학 엔코더가 내장되어 있고 Pendulum과 Motor의 각도에 대한 Resolution 은 512 counts/revolution 이므로 강화학습 에이전트의 입력 값에 해당하는 상태 정보를 도출하기에 적합하다[6]. 그리고 QUBE-Servo2 제품과의 통신은 Serial Peripheral Interface(SPI) 방식을 사용했고 라즈베리파이(Raspberry Pi 3 Model B)와 연결하여 시스템을 구성하였다.

2. 강화학습 에이전트

강화학습 에이전트의 최종목표는 환경과 상호작용하며 목표인 보상(Reward) 값을 최대로 높이는 데 있다. 강화학습 알고리즘의 대표적인 예인 Q-Learning 의 에이전트는 보상 R_t의 최대화를 위해서 현재 상태 값에 해당하는 상태 s_t를 관찰하고 현재 정책 π에 맵핑된 일련의 행동 중 하나인 a_t를 얻어내는데 이는 곧 최적의 $Q^*(s,a)$의 값을 얻어 내는 것과 같다[7, 8].

$$Q^*(s,a) = argmax(E[R_t | s_t = s, a_t = a, \pi]) \quad (1)$$

+: 교신저자 한연희(한국기술교육대학교)

"이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2018R1A6A1A03025526)"

Q-learning 의 핵심은 스텝 t에서의 상태를 관찰하여 정책에 의한 행동의 가치 판단에 있다. 이를 Bellman 방정식 기반의 연속된 업데이트를 통하여 발전시켜나간다. 여기에 비선형적 특성을 갖는 Neural Network가 추가되며 Q-Network라 불린다. Q-Network 의 파라미터 θ를 업데이트 하는 것이 곧 Q-learning 의 목적이며 Loss Function $L(\theta)$을 기반으로 훈련이 진행된다[7, 8].

$$Q^*(s,a) = E[r + \gamma max_a Q(s',a')|s,a] \quad (2)$$

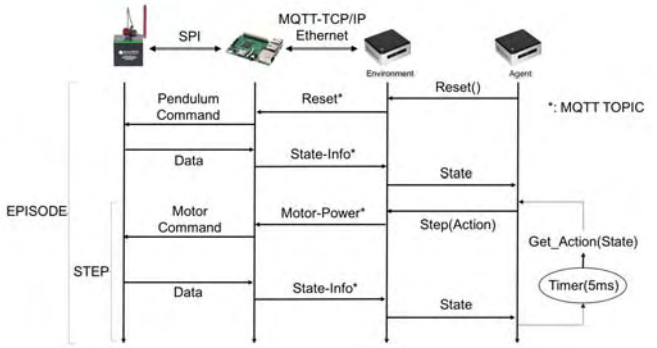
$$L(\theta) = E[(Q^*(s,a) - Q(s,a;\theta))^2] \quad (3)$$

우리는 Q-learning 에서 진보된 Deep Q-learning(DQN) 을 비롯하여 Time-Differences 기반의 Advantage Actor-Critic(A2C)[9], 비동기 병렬 다중 에이전트를 갖는 Asynchronous Advantage Actor-Critic(A3C)[9] 을 모두 고려한 시스템을 구축하였다. 세 가지 모델은 정책의 사용 방향이 다르고 어떠한 네트워크 토폴로지를 사용하느냐에 따라서 성능이 크게 차이가 날 수 있으므로 현재까지 연구된 기록만을 놓고 특정 알고리즘 모델을 선택하지는 않았다. 하지만 세 가지 모델을 선정한 이유는 다음과 같다. 먼저 DQN 은 오프폴리시(Off-Policy) 기반의 알고리즘으로써 Deep Input Network의 복합적 신경요소를 활용하여 환경을 관찰함으로써 복잡한 초기 상태를 갖는 Rotary Inverted Pendulum에 적합하다. 하지만 DQN은 학습 메모리를 사용하여 과거의 상태를 마치 현재의 상태로 일반화시켜 사용한다는 점에서 Time-Differences 기반의 온폴리시(On-Policy) 알고리즘인 A2C 및 A3C와는 큰 차이를 보인다. A2C와 A3C의 공통점은 네트워크가 정책망과 가치망으로 분류된다는 것이다. 정책을 관할하는 정책망과 가치를 관할하는 가치망으로 나뉘는데 가치망에서 결정된 가치 값에 Cross Entropy 수식을 접목하여 미래의 결과를 알 수 없는 현재에서의 불확실성을 낮추면서 현재로써 최적의 가치를 얻을 수 있는 Advantage 값을 정책망에서 활용한다[1, 9]. 다른 점은 A2C와 달리 A3C는 비동기적 특성을 가지기 때문에 여러 Local 에이전트가 상호작용을 하여 Global 에이전트의 안정된 가치판단을 유도한다. 하지만 실제 Physical 시스템에 해당하는 Rotary Inverted Pendulum은 가상으로 동일한 여러 개의 시스템으로써 동시에 구동시킬 수 없으며 가상 환경과 달리 외란에 취약하다. 그러므로 비동기적인 다중 에이전트의 학습 환경 구성에 차질을 빚을 우려도 존재한다. 세 강화학습 모델은 각기 장단점이 있으며 본 논문이 제시하려는 시스템의 에이전트의 강화학습 알고리즘 모델로써 다수의 실험으로 성능을 평가해 보아야 할 필요가 있다.

3. 시스템 설계 및 구현

Rotary Inverted Pendulum은 하나의 모터를 가지고 자유도가 상당히 높은 도립진자를 제어해야하는 시스템으로써 Underactuated-System 이라고도 불린다. 비선형 제어법칙을 설명하기에 매력적인만큼 강화학습 모델의 효과를 입증시키기에도 적합한 시스템이라 할만하다[10].

QUBE-Servo2 의 제어를 위해서는 뒷면의 패널을 이용한 통신이 이루어져야 한다. 뒷면의 패널은 USB 혹은 SPI 연결이 가능하도록 옵션이 주어지는데 우리는 SPI 패널을 선택했으며 이와 함께 라즈베리파이를 이용하였다. 그리고 베어본 컴퓨터(Barebone PC)를 에이전트(Agent)와 환경(Environment) 구성에 사용하였고 라즈베리파이와 Ethernet 연결을 시켰다. 전체 시스템 구성도는 아래 그림에서 보이고 있다.



(그림 1) 시스템 구성도

(그림 1)에서는 QUBE-Servo2와 라즈베리파이 사이의 통신 규격인 SPI, 라즈베리파이와 베어본 컴퓨터 간의 Ethernet 연결을 표현했으며 각 컴포넌트 사이의 네트워킹 시퀀스를 나타냈다. 한 대의 베어본 컴퓨터에서는 환경과 강화학습 에이전트 간의 상호작용을 구조적으로 분리시켰다. 여기서 환경은 가상의 에이전트와 실제 시스템인 QUBE-Servo2 간의 다리 역할을 해주며 통신을 위해서, IoT 장비를 위해 경량화 된 프로토콜인 MQTT 를 이용하였다. MQTT-Brocker 는 그림 상 존재하지는 않지만 환경과 라즈베리파이 간 개설된 Topic 에 메시지를 담아 발행하는 것과 구독하는 행위를 중계해준다. 간단하게 예를 들면 'Reset*' 과 'Motor-Power*' Topic은 환경이 발행하고 라즈베리파이가 구독을 하는 것이고, 반대로 'State-Info*' Topic 은 라즈베리파이가 발행하고 환경이 구독을 하는 것이다.

<표 1> 강화학습 에이전트의 Action 과 Step

Action										
0	1	2	3	4	5	6	7	8	9	10
-300	-240	-180	-120	-60	0	60	120	180	240	300
State										
Pendulum Angle(radian)		Pendulum Speed(radian/s)		Motor Angle (radian)		Motor Speed (radian/s)				
θ_{k1}		θ'_{k1}		θ_{k2}		θ'_{k2}				

위 <표 1>에서 행동은 QUBE-Servo2 의 Motor에게

얼마만큼의 전력(Voltage)을 인가해줄지 에이전트의 행동으로 표현한 것이다. 행동은 11개의 Index로 구성되어 각 Index 마다 Motor 전력 값을 맵핑(Mapping)시켰다. 또한 상태는 에이전트가 이해할 수 있는 형태로 QUBE-Servo2의 Pendulum과 Motor의 현재 상태를 표현한 것이다. 행동과 상태에 대한 자세한 내용은 <표 1>에서 보인다.

전체 Flows는 강화학습이 진행되는 동안의 여러 Episodes 중 하나의 Episode를 그린 것이며 에이전트의 'Step(Action)' 행위를 기점으로 강화학습 에이전트가 QUBE-Servo2 에 행동을 전달하고 상태와 보상을 넘겨받는 일련의 과정을 'Step(Action)' 으로 표현하였다. 여기서 보상의 수식은 아래와 같다.

$$-(\theta_{k1}^2 + 0.1*(\theta'^2_{k1}) + 1e-6*action^2) \qquad (4)$$

에이전트는 5ms 마다 Interrupt 되는 Timer를 지니고 있는데 이 Timer가 하나의 Episode 내에서 'Step(Action)' 이 5ms 마다 반복되는 것을 관할한다. 즉 동일한 시간 간격으로 QUBE-Servo2 의 제어를 에이전트가 할 수 있도록 돕는다.

4. 결론

우리는 Rotary Inverted Pendulum이라는 비선형적 제어의 실험을 진행하기 위해서 적합한 실험환경을 구성하여 사용하였다. 가상의 환경을 MQTT 통신을 통해서 실제 환경과 네트워킹을 할 수 있도록 구축함으로써 에이전트가 직접 실제 환경과 상호작용을 하도록 전체 시스템을 구성하였다.

강화학습은 현재까지 아직도 실제 환경에 접목되어 쓰이는 경우가 드물고 쓰인다 하더라도 복잡하고 어려운 문제에 적용시키는 경우가 적었다. 전통적인 제어분야의 전유물로 비선형제어 실험에 많이 사용되던 시스템을 강화학습 에이전트가 제어할 수 있다는 것은 앞으로 현실세계에 접목되어 활용될 여지가 충분하다는 것을 입증할 수 있다. 우리는 현재 전체 시스템 구축을 끝내고 실험 중에 있으며 계속적으로 피드백 중이다.

참고문헌

[1] Mnih, Volodymyr, "Human-level control through deep reinforcementlearning", in Nature, 518(7540):529 - 533, 02 2015.

[2] Mnih, Volodymyr "Playing atari with deep reinforcementlearning" In NIPS Deep Learning Workshop, 2013.

[3] Viroch S. "Real-Time Optimal Control for Rotary Inverted Pendulum" American Journal of Applied Sciences 6, 2009.

[4] Lal B. P. "Optimal Control of Nonlinear Inverted Pendulum Dynamical System with Disturbance Input using PID Controller & LQR" IEEE International Conference on Control System, Computing and Engineering, 2011.

[5] Md. Akhtaruzzaman, A. A. Shafie "Modeling and Control of a Rotary Inverted Pendulum Using Various Methods, Comparative Assessment and Result Analysis" IEEE International Conference on Mechatronics and Automation, 2010.

[6] "https://www.quanser.com/products/qube-servo-2/", QUENSER INOVATE&EDUCATE, QUBE_Servo_2_Product_Info_Sheet_v1.0, accessed Sep 20. 2018.

[7] Kai Arulkumaran "A Brief Survey of Deep Reinforcement Learning" in IEEE Signal Processing Magazine Special Issue On Deep Learning For Image Understanding, 2017.

[8] Christopher JCH Watkins and Peter Dayan. "Q-Learning", Machine Learning, 8(3-4):279-292, 1992.

[9]Volodymyr Mnih, "Asynchronous Methods for Deep Reinforcement Learning", Proceedings of the 33rd International Conference on MachineLearning, New York, NY, USA, 2016. JMLR: W&CP volume48.

[10] Alexander B. "Optimal Control of a Double Inverted Pendulum on a Cart" OGI School of Science & Engineering, OHSU, Technical Report CSE-04-006, 2004.