Journal of Korea Robotics Society (2022) 17(2):221-229
https://doi.org/10.7746/jkros.2022.17.2.221 ISSN: 1975-6291 / eISSN: 2287-3961


# Q-Learning을 사용한 로봇팔의 SMCSPO 게인 튜닝

# Gain Tuning for SMCSPO of Robot Arm with Q-Learning

이 진 혁[1] · 김 재 형[2] · 이 민 철†
JinHyeok Lee[1], JaeHyung Kim[2], MinCheol Lee†

**Abstract:** Sliding mode control (SMC) is a robust control method to control a robot arm with nonlinear properties. A high switching gain of SMC causes chattering problems, although the SMC allows the adequate control performance by giving high switching gain, without the exact robot model containing nonlinear and uncertainty terms. In order to solve this problem, SMC with sliding perturbation observer (SMCSPO) has been researched, where the method can reduce the chattering by compensating the perturbation, which is estimated by the observer, and then choosing a lower switching control gain of SMC. However, optimal gain tuning is necessary to get a better tracking performance and reducing a chattering. This paper proposes a method that the Q-learning automatically tunes the control gains of SMCSPO with an iterative operation. In this tuning method, the rewards of reinforcement learning (RL) are set minus tracking errors of states, and the action of RL is a change of control gain to maximize rewards whenever the iteration number of movements increases. The simple motion test for a 7-DOF robot arm was simulated in MATLAB program to prove this RL tuning algorithm. The simulation showed that this method can automatically tune the control gains for SMCSPO.

**Keywords:** Robust Control, Reinforcement Learning, Q-Learning, Sliding Mode Control, Auto-Tuning

## 1. 서 론

일반적으로 로봇을 제어할 때 사용되는 선형제어는 로봇을 선형 시스템으로 가정하여 나머지 요소를 고려하지 않고 이루어지기 때문에 비선형성이 강한 로봇은 선형제어로 제어하기에 만족할 만한 성능을 얻기 힘들다[1]. 강인 제어는 시스템의 불확실성을 보완하는 비선형 제어 기술로 로봇을 제어할 때 높은 성능을 보인다. 슬라이딩 모드 제어는 간단한 구조로 인해 자주 사용되는 강인제어방법 중 하나이다[2]. 이 제어 방법은 게인 값을 크게 증가시키는 것으로 제어 대상의 동특성을 정확히 알지 못해도 사용할 수 있지만 부호 함수와 높은 제어 게인으로 인해 떨림 현상인 채터링(chattering)이 발생한다[3]. 이를 해결하기 위해 1997년 Moura 와 Olgac가 제안한 슬라이딩 섭동 관측기를 결합한 슬라이딩 모드 제어기(Sliding Mode Control with Sliding Perturbation Observer, SMCSPO)는 시스템의 비선형성과 외란을 추정하고 보상하여 작은 제어 게인으로 안정적인 제어를 가능하게 한다[4]. 이 강인제어기는 기존의 채터링 문제를 해결할 뿐만 아니라 추정된 상태들을 다양한 방면으로 활용할 수 있다는 장점도 있다[5,6]. 여기서 포화함수의 경계 값과 시스템의 보정 값 등을 제외한 제어시스템을 구성하는데 필요한 여러가지 제어게인들은 시스템 오차방정식을 0으로 수렴하기 위해 주어지는 목표 고유값인 극점을 파라미터로 사용하여 한 번에 결정될 수 있다. 보통 이 파라미터는 낮으면 섭동의 추정성능이 감소하고 높으면 채터링을 유발하는 등 관측기와 제어기의 성능에 직접적인 영향을 주며, 시행착오적인 방법을 통해 결정하기 때문에 최적으로 선정하기에

Received : Nov. 29. 2021; Revised : Jan. 23. 2022; Accepted : Mar. 9. 2022
※ This paper was supported by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008473, HRD Program for Industrial Innovation) and funded under 『the Competency Development Program for Industry Specialists』 of the Korean Ministry of Trade, Industry and Energy (MOTIE), operated by Korea Institute for Advancement of Technology (KIAT). (No. P0008473, The development of high skilled and innovative manpower to lead the Innovation based on Robot)
1. MS Course, School of Mechanical Engineering, Pusan National University, Busan, Korea (kkjs365@naver.com)
2. Ph.D Course, School of Mechanical Engineering, Pusan National University, Busan, Korea (11045kjh@naver.com)
† Professor, Corresponding author: School of Mechanical Engineering, Pusan National University, Busan, Korea (mclee@pusan.ac.kr)

어려움을 겪는다. 제어게인 선정의 어려움을 해결하기 위해 다양한 연구가 진행되었다. 2008년 Kuo 등이 Self-tuning 알고리즘을 사용하여 SMC의 제어게인을 최적화하는 연구를 제안하였다[7]. 이 방법은 추정된 상태를 이용하기 위해 더 복잡한 구조를 가지는 SMCSPO에 적용하기에 한계가 있다. 2004년 You가 유전알고리즘을 사용하여 SMCSPO의 제어게인을 최적화하는 연구를 제안하였다[8]. 유전 알고리즘은 최적의 파라미터를 찾는데 적합한 방법이지만, 많은 실험 데이터를 필요로 하며, 프로그래밍을 통해 실험적으로 구현하기가 쉽지 않다.

강화학습(reinforcement learning, RL)은 에이전트의 경험을 통해 기대보상이 최대가 되도록 하는 정책을 찾는 학습 방법이다[9]. 이 방법은 로봇이 작업을 수행할 때 인공적인 지능을 부여하여 비전데이터와 반력을 받아 부품을 조립할 수 있도록 움직임을 판단하거나[10,11] 계산하기 어렵고 복잡한 형상을 가진 로봇암의 역기구학 등을 계산할 수 있게 한다[12]. 제어 분야에도 강화학습을 접목한 연구가 진행되고 있다. 위치와 속도 등의 로봇의 상태를 입력으로 받아서 직접적으로 제어신호를 출력하는 강화학습 모델이 연구되었다[13]. 설계자가 직접 제어대상과 환경에 따라 모델링하고 제어알고리즘을 선택하여 적용하는 다른 방법에 비해 이 방법은 강화학습 알고리즘을 적용시켜 학습이 완료되면 로봇이 자동으로 최적의 추종성능을 가질 수 있게 된다. 또 다른 알고리즘으로 기존의 제어기를 사용하면서 강화학습 모델이 간접적으로 간섭하여 자동으로 제어기의 제어게인을 설정하는 알고리즘도 존재한다[14].

강화학습 알고리즘 중 Q-learning은 기초적인 강화학습알고리즘으로 배열형태의 Q-table을 통해 가치에 따른 정책을 결정하며, 간단한 구조로 인해 구현하기 쉽다는 장점이 있다[15]. 본 연구에서는 Q-learning을 사용해 로봇팔을 제어하기위한 SMCSPO의 제어게인을 튜닝하는 알고리즘을 제안한다. 로봇이 반복적으로 움직일 때 마다 추종오차를 최소화할 수 있도록 Q-learning이 학습하여 자동으로 SMCSPO의 제어게인을 튜닝할 수 있게 하였고, 시뮬레이션을 통해 알고리즘이 로봇제어에 적합한지를 검증하였다.

본 논문은 다음과 같이 구성된다. 2장에서는 기존의 SMCSPO에 대한 내용을 간단하게 요약하였으며, 3장에서는 반복동작과 강화학습을 이용한 제어게인 튜닝 알고리즘 이론을 설명한다. 가상 7축 로봇의 MATLAB과 Simulink를 사용한 시뮬레이션 환경을 4장에서 구성하고, 시뮬레이션 결과를 5장에서 분석하였다. 마지막으로 6장에서 결론을 정리한다.

## 2. SMCSPO

SMCSPO는 슬라이딩 표면을 따라 부호함수로 제어되는 기존의 Sliding Mode Control (SMC)와 불확실성 및 외란과 같은 섭동 및 상태를 추정하는 Sliding Perturbation Observer (SPO)를 결합하여 SMC의 채터링 문제를 보완하는 방법이다[4]. SMCSPO를 설명하기 위해 Lagrangian-dynamics[5,16]를 사용하여 식 (1)과 같이 $n$ 자유도 다관절로봇암의 동역학 모델식을 정의하고, $j$축의 회전각도를 식 (2)와 같이 정의하면 식 (3), 식 (4)와 같이 유도된다,

$$u_j = \sum_{k=1}^{n} M_{jk}(\theta_1)\ddot{\theta}_{1k} + \sum_{k=1}^{n}\sum_{m=1}^{n} C_{jkm}(\theta_1, \dot{\theta}_1) + G_j(\theta_1) \quad (1)$$

$$\dot{\theta}_{1j} = \theta_{2j} \quad (2)$$

$$\dot{\theta}_{2j} = (M_{oj}(\theta_1) + \Delta M_j(\theta_1))^{-1}(-C_{oj}(\theta_1, \theta_2) - G_{oj}(\theta_1) - \Delta C_j(\theta_1, \theta_2) - \Delta G_j(\theta_1) + u_j) + d_j \quad (3)$$

$$\dot{\theta}_{2j} = f_j(\theta_1, \theta_2) + \Delta f_j(\theta_1, \theta_2) + \sum_{i=1}^{n}(b_{ji}(\theta_1) + \Delta b_{ji}(\theta_1))u_i + d_j \quad (4)$$

$$y_j = \theta_{1j} \quad (5)$$

여기서, $j = 1, \cdots, n$으로 상태 $\theta_{1j}$는 엔코더로 측정한 로봇의 $j$축의 각도이다. $M_{oj}(\theta_1)$, $C_{oj}(\theta_1, \theta_2)$, $G_{oj}(\theta_1)$는 $j$축의 관성항, 코리올리 힘과 원심력, 중력을 나타내는 선형성분을 나타내며, $\Delta M_j(\theta_1)$, $\Delta C_j(\theta_1, \theta_2)$, $\Delta G_j(\theta_1)$는 각각의 비선형 항요소를 분리한 항과 파라미터 추정오차로 발생하는 항의 합으로 나타낸다. 제어입력 $u_j$는 구동부에서 발생하는 구동토크, $d_j$는 외란을 의미한다. 본 논문에서는 $\dot{\theta}_{2j}$에 대한 링크의 동역학 식 (3)을 SMCSPO를 사용하기 위한 식 (4)로 적용 가능하며, 일반적으로 구하기 어려운 비선형, 불확실성 요소들을 제외하면 $M_{oj}^{-1}(\theta_1)(-C_{oj}(\theta_1, \theta_2) - G_{oj}(\theta_1))$는 시스템 항 $f_j(\theta_1, \theta_2)$으로, $M_{oj}^{-1}(\theta_1)u_j$는 제어 입력 항 $\sum_{i=1}^{n}(b_{ji}(\theta_1)u_i)$로 정의할 수 있다. 직접 동역학을 계산하지 않고 신호압축법 등의 시스템 규명 방법을 통해 대략적인 선형요소 $f_j(\theta_1, \theta_2)$와 $(b_{ji}(\theta_1))$를 얻을 수 있다[5]. 여기서 각 요소에 대한 비선형 항과 불확실성, 외란 등의 항을 섭동항으로 표현하면 식 (6)과 같이 정의된다.

$$\Psi_j = \Delta f_j(\theta_1, \theta_2) + \sum_{i=1}^{n}\Delta b_{ji}(\theta_1)u_i + d_j \quad (6)$$

정의한 섭동을 추정해 낼 수 있으면 보상되지 못한 불확실성과 외란을 한 번에 제어기에 보상할 수 있다[4].

$$\dot{\theta}_{2j} = \alpha_{3j}\bar{u}_j + \Psi_j \quad (7)$$

$$\theta_{3j} = \alpha_{3j}\theta_{2j} - \Psi_j/\alpha_{3j} \quad (8)$$

식 (4)에서 선형요소인 $f_j(\theta_1,\theta_2)+\sum_{i=1}^{n} b_{ji}(\theta_1)u_i$는 SPO의 구조를 단순화하기 위해 새로운 제어 변수 $\bar{u}_j$와 임의의 상수 $\alpha_{3j}$의 곱으로 치환하여 식 (7)과 같이 변환할 수 있다. 그리고 세 번째 상태 $\theta_{3j}$를 정의한다. $\theta_{3j}$는 시스템의 섭동을 추정하기 위한 상태로 물리적인 의미를 나타내지는 않는다. $\theta_{1j}$, $\theta_{2j}$, $\theta_{3j}$와 섭동 $\Psi_j$는 SPO를 사용하여 추정할 수 있으며 다음과 같다[4].

$$\dot{\hat{\theta}}_{1j}=\hat{\theta}_{2j}-k_{1j}sat(\tilde{\theta}_{1j})-\alpha_{1j}\tilde{\theta}_{1j} \tag{9}$$

$$\dot{\hat{\theta}}_{2j}=\alpha_{3j}\tilde{u}_j-k_{2j}sat(\tilde{\theta}_{1j})-\alpha_{2j}\tilde{\theta}_{1j}+\hat{\Psi}_j \tag{10}$$

$$\dot{\hat{\theta}}_{3j}=\alpha_{3j}{}^2(-\hat{\theta}_{3j}+\alpha_{3j}\hat{\theta}_{2j}+\bar{u}_j) \tag{11}$$

$$\hat{\Psi}_j=\alpha_{3j}(-\hat{\theta}_{3j}+\alpha_{3j}\hat{\theta}_{2j}) \tag{12}$$

여기서, '^'와 '~'는 각각 추정값과 추정오차를 의미한다. 양의 상수 $k_{1j}$, $k_{2j}$, $\alpha_{1j}$, $\alpha_{2j}$는 $\alpha_{3j}$와 함께 사용자가 설정할 파라미터들로 $\theta_{1j}$의 추정오차 $\tilde{\theta}_{1j}=\hat{\theta}_{1j}-\theta_{1j}$가 0이 되도록 유도하며 SPO의 성능에 직접적인 영향을 준다. 식 (9), 식 (10)에서의 $sat$은 포화함수로 식 (13)과 같다.

$$sat(\tilde{\theta}_{1j})=\begin{cases}\tilde{\theta}_{1j}/\left|\tilde{\theta}_{1j}\right|, if\left|\tilde{\theta}_{1j}\right|\geq\epsilon_{oj}\\ \tilde{\theta}_{1j}/\epsilon_{oj}, if\left|\tilde{\theta}_{1j}\right|\leq\epsilon_{oj}\end{cases} \tag{13}$$

식 (13)의 포화함수 $sat$는 기존의 Sliding Observer (SO)[17]에서 부호함수를 대체하는 포화함수이다. SO의 부호함수는 추종오차 $\tilde{\theta}_{1j}$가 0에 근접할 때 게인이 크다면 추정상태에 큰 변화량을 유발해 채터링을 발생시키게 된다. 하지만 포화함수는 $\tilde{\theta}_{1j}$가 0에 근접할 때 적은 변화량을 유도하기 때문에 이 문제를 해결할 수 있다. 여기서, $\epsilon_{oj}$는 SPO에서 사용되는 포화함수의 경계층이다. 이 SPO를 기존의 SMC에 적용하기 위해 실제 상태를 추정된 상태로 대체함으로써 다음과 같은 슬라이딩 평면을 얻는다[4].

$$\hat{s}_j=c_{j1}(\hat{\theta}_{1j}-\theta_{1dj})+(\dot{\hat{\theta}}_{1j}-\dot{\theta}_{1dj}) \tag{14}$$

$$\dot{\hat{s}}_j=-K_jsat(\hat{s}_j) \tag{15}$$

$$sat(\hat{s}_j)=\begin{cases}\hat{s}_j/\left|\hat{s}_j\right|, if\left|\hat{s}_j\right|\geq\epsilon_{cj}\\ \hat{s}_j/\epsilon_{cj}, if\left|\hat{s}_j\right|\leq\epsilon_{cj}\end{cases} \tag{16}$$

여기서, $\theta_{1dj}$는 $\theta_{1j}$에 대한 목표 각도이다. 식 (14)에서 추정 슬라이딩 표면 $\hat{s}_j$는 각도를 목표 각도로 이동시키기 위해 0으로 수렴해야 한다. 식 (15)는 $\hat{s}_j$가 0에 구속될 수 있도록 하는 switching function이다. 식 (16)은 식 (13)과는 반대로 슬라이딩 평면의 추정값에 대한 포화함수로, $\epsilon_{cj}$는 SMC에 대한 포화함수의 경계층이다. 다음으로 제어입력 $u_j$를 얻기 위해 $\bar{u}_j$를 계산한다[4].

$$\bar{u}_j=\frac{1}{\alpha_{3j}}\{-K_jsat(\hat{s}_j)+[\frac{k_{2j}}{\epsilon_{oj}}+c_{j1}\left(\frac{k_{1j}}{\epsilon_{oj}}\right)-\left(\frac{k_{1j}}{\epsilon_{oj}}\right)^2]\tilde{\theta}_{1j}+\ddot{\theta}_{1dj}-c_{j1}\left(\hat{\theta}_{2j}-\dot{\theta}_{1dj}\right)-\hat{\Psi}_j\} \tag{17}$$

$$u=B^{-1}\ \ Col\ [\alpha_{3j}\bar{u}_j-f_j(\hat{\theta}_1,\hat{\theta}_2)] \tag{18}$$

식 (17)을 통해 $\bar{u}_j$를 구하고, $\alpha_{3j}\bar{u}_j$를 정리한 식 (18)을 통해 제어 입력 벡터 $u=\left[u_1\cdots u_n\right]$를 얻을 수 있다. 여기서 $Col$은 열벡터, $B=\left[b_{ji}(\hat{\theta}_1)\right]_{n\times n}$를 의미한다. 관측기와 제어기의 안정도를 확인하기 위해 식 (9)에서 식 (11)까지의 추정값과 식 (2), 식 (7), 식 (8)의 참값으로 상태의 추정오차에 대한 식을 얻고 슬라이딩 표면 $s_j$와 함께 다음과 같은 상태 방정식을 나타낼 수 있다[4].

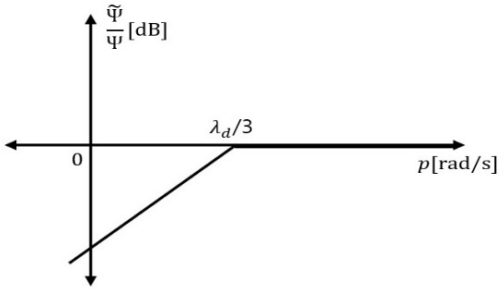
$$\begin{bmatrix}\dot{\tilde{\theta}}_{1j}\\ \dot{\tilde{\theta}}_{2j}\\ \dot{\tilde{\theta}}_{3j}\\ \dot{s}_j\end{bmatrix}=\begin{bmatrix}-k_{1j}/\epsilon_{oj} & 1 & 0 & 0\\ -k_{2j}/\epsilon_{oj} & \alpha_{3j}{}^2 & -\alpha_{3j} & 0\\ 0 & \alpha_{3j}{}^3 & -\alpha_{3j}{}^2 & 0\\ k_{2j}/\epsilon_{cj}+(c_{j1}-k_{1j}/\epsilon_{oj})^2 & -(2c_{j1}+\alpha_3^2) & \alpha_{3j} & -c_{j1}\end{bmatrix}\times\begin{bmatrix}\tilde{\theta}_{1j}\\ \tilde{\theta}_{2j}\\ \tilde{\theta}_{3j}\\ s_j\end{bmatrix}+\begin{bmatrix}0\\0\\1\\0\end{bmatrix}\dot{\Psi}_j/\alpha_{3j} \tag{19}$$

추정 오차와 슬라이딩 표면이 안정하고 빠르게 0으로 수렴할 수 있도록 식 (19)에서 시스템 행렬 고유값의 실수부가 음수여야 하며, 이는 극점배치기법을 사용하는 것으로 모든 기대 고유값을 -$\lambda_{dj}$로 설정하여 SMCSPO에 필요한 제어게인들을 식 (20)와 같이 설정할 수 있다[4].

$$K_j=\lambda_{dj}\epsilon_{cj},\ \ k_{1j}=3\lambda_{dj}\epsilon_{oj},\ \ k_{2j}=\lambda_{dj}k_{1j}\qquad \alpha_{3j}=\sqrt{\lambda_{dj}/3},\ \ c_{j1}=\lambda_{dj} \tag{20}$$

$\lambda_{dj}$는 제어 성능에 직접적인 영향을 미치는 필수 제어 파라미터가 된다. 이때 식 (8)과 식 (11)을 정리하고 주파수역으로 변환하면 주파수역의 복소변수 $p$에 대해 식 (21)과 같이 실제 섭동과 섭동의 추정오차의 관계를 나타내는 전달함수를 [Fig. 1]과 같이 얻을 수 있다[4].

$$\frac{\tilde{\Psi}_j(p)}{\Psi_j(p)}=-\frac{p(p^2+3\lambda_{dj}p+3\lambda_{dj}{}^2)}{(p+\lambda_{dj})^3} \tag{21}$$




[Fig. 1] Magnitude plot of the perturbation to the estimated error

이 전달함수는 고주파통과필터의 특성을 가지기 때문에 제어게인 $\lambda_{dj}$의 값을 작게 줄 경우 높은 주파수의 섭동에 대하여 정상적인 추정이 불가능하다. 따라서 시스템이 높은 주파수를 가지는 섭동을 정상적으로 추정하기 위해서 높은 제어게인을 요구한다. 그러나 높은 값의 $\lambda_{dj}$는 $K_j$의 값을 증가시켜 채터링 현상을 유발하기 때문에 제어대상, 주변환경, 모션에 따라 적절한 범위에서 제어게인을 선택하는 것이 중요하다.

## 3. Q-Learning 튜닝 알고리즘

일반적으로 사용되는 인공지능인 신경망은 지도학습으로 학습단계에서 라벨링작업이 필요하며, 최적의 제어게인을 찾아야하는 것이 목적인 튜닝 문제에는 모순이 발생한다. 경험에 의존하는 강화학습은 가치함수 추정을 통한 정책 탐색 방법으로 적절한 제어게인을 찾는다는 결과를 얻어낼 수 있다[9]. 이러한 강화학습의 특징은 구조가 복잡한 SMCSPO의 제어게인을 튜닝하는 것에 적합하다. 3장에서는 로봇이 반복적인 작업을 할 때마다 학습하여 자동으로 SMCSPO의 제어게인 $\lambda_d$의 튜닝을 수행하는 강화학습 알고리즘을 소개한다.

강화학습을 실제 로봇의 제어게인 튜닝에 적용하기 위해 상태, 행동, 및 보상을 사용자가 적절하게 설계해야 한다. 본 연구에서는 상태를 SMCSPO의 제어게인 $\lambda_d$로 설정하고, [Table 1]에서 주어진대로 $\lambda_d$는 1의 간격으로 1에서 100까지 설정 가능하도록 하였다. 행동은 제어게인의 증가, 감소로 설정하였다. 제어게인의 증감의 단위가 작을수록 더 세밀한 탐색이 가능하지만, 더 많은 탐색량이 요구되고, 단위가 클수록 빠른 탐색이 가능하지만 최적의 게인값을 건너뛸 수 있게 된다. 상태와 행동의 개수는 자유롭게 설정할 수 있으나 개수가 많을수록 연산량이 늘어나게 된다.

[Table 1] States and action of reinforcement learning

| | |
|---|---|
| State | $s_t = \lambda_d$ (1-100) |
| Action | $a_t$ = -5 or -1 or +1 or +5 |

로봇이 동작할 때 최소의 추종오차를 가지는 제어게인을 찾기 위해 식 (22)과 같이 반복 횟수 t에 대한 비용함수 $J_t$와 보상 함수를 식 (23)와 같이 정의하였다.

$$J_t = \frac{1}{N}\sum_{k=0}^{N} \left| \theta_1(k) - \theta_{1d}(k) \right| \tag{22}$$

$$r_t = (\sigma - J_t) \tag{23}$$

$J_t$는 한번의 반복동작에서 단위시간 $k$가 0부터 $N$까지의 추종오차의 절대값의 평균으로 정의하였고, 보상 $r_t$는 비용을 빼는 것으로 비용 $J_t$가 감소하면 보상 $r_t$가 증가하도록 설정하였다. 강화학습은 보상을 최대로 만드는 행동을 선택하기 때문에 추종오차가 적어지도록 제어게인을 수정하게 된다. $J_t$는 로봇이 한번의 작업이 끝나고 나면 제어게인을 수정하고 보상으로 가치를 계산하고 나면 다음 제어게인의 비용을 측정하기 위해 초기화한다. $J_t$만을 사용하여 보상을 이루도록 할 경우에 Q-table 내부의 가치 값들이 감소만 하기 때문에 한 번의 iteration에서의 목표로 하는 평균 추종오차 $\sigma$를 더해주어 오차가 일정범위 이내로 줄어들면 가치가 양의 값을 가지도록 설정하였다. 이때 $\sigma$는 보상이 쉽게 양의 값이 된다면 시스템이 낮은 성능에도 충분한 성능을 가진다고 판단하기 때문에 상황에 맞게 허용할 최대의 평균 추종오차 값으로 설정한다.

기본적인 가치기반 강화학습 중 하나인 Q-learning[15]은 배열구조의 Q-table을 통해서 어떤 대상이 특정 상태에 대해 가장 높은 보상을 받을 수 있도록 효율적인 행동을 결정한다. Q-table을 구성하는 가치함수의 Bellman equation은 식 (24)과 같다.

$$\begin{aligned} Q(s_t, a_t) = (1-\rho)\, Q(s_t, a_t) \\ -\rho(r_t + \gamma \max_a Q(s_{t+1}, a)) \end{aligned} \tag{24}$$

여기서, $\rho$는 한번의 학습이 부분적으로 적용되는 정도를 의미하는 학습율을 나타내고, $\gamma$는 현재 보상과 미래에 얻을 수 있는 보상에 대한 가치의 가중치를 조절할 수 있는 할인율을 의미한다. Q-learning을 튜닝 알고리즘에 적용시킬 때, 갱신되는 데이터의 비중을 크게하고 local minimum 문제를 방지하기 위해 학습율 $\rho$와 할인율 $\gamma$는 최소 0.5 이상으로 가능한 높게 설정하는 것이 좋다. $Q$값은 해당 상태에서의 행동에 대한 가치를 의미한다. 행동은 어떤 상태에서 가장 높은 가치를 가질 수 있도록 결정되고 행동결정을 식 (25)와 같이 나타낼 수 있다.

$$a_t = \arg\max_a (\epsilon_{greedy};\, Q(s_t, a) + (1-\epsilon_{greedy}) \cdot randn) \tag{25}$$

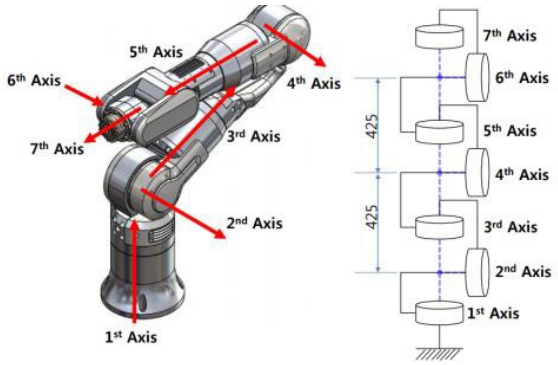
$$\epsilon_{greedy} = g \cdot t \cdot \ln(1 + \frac{1}{g \cdot t}) \tag{26}$$

식 (25)에서 argmax는 다음 값을 가장 크게 만드는 행동 $a$를 결정한다. $\epsilon_{greedy}$: $Q(s_t, a)$는 $Q$값을 통해 행동을 결정하는 활용항이고, $(1-\epsilon_{greedy})\cdot randn$은 정규분포에 따른 임의의 값 $randn$을 통해 무작위로 행동을 결정하는 탐색항이다. 동작의 반복횟수 $t$가 증가할 때마다 $\epsilon_{greedy}$의 크기가 0에서 1로 수렴하고 탐색항이 활용항보다 크기가 줄어들어 영향을 미칠 수 없게 된다[18]. 식 (26)에서 0부터 1까지의 상수 g는 클수록 $\epsilon_{greedy}$가 1로 수렴하는 속도를 빠르게 한다. 3장에서 설명된 알고리즘을 사용하여 로봇이 반복적인 작업을 수행할 때마다 추종오차를 비교하여 제어게인과 Q-table을 업데이트하고 다음 작업을 수행하 는 강화학습 튜닝 알고리즘을 구현할 수 있다.

## 4. 시뮬레이션 시스템 구성

제안된 튜닝 알고리즘을 검증하기 위해 MATLAB과 Simulink의 SimMechanics를 사용하여 가상의 시뮬레이션을 진행하였다. 제어대상으로 NT Robot사에서 제작한 7축 다관절로봇암 RoMAN7의 가상 모델을 구현하였다[Fig. 2]. [Table 2]는 로봇 링크의 사양을 나타낸다. 이 로봇을 제어하기 위해 [Fig. 3]의 제어 블록선도와 같이 시스템에서 역기구학, 궤적계획 그리고 SMCSPO를 적용시켰다.

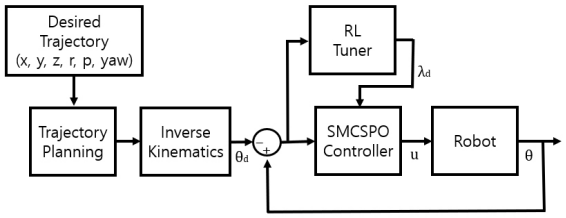
로봇의 말단부를 원하는 위치로 이동시킬 때 경로를 일정 간격으로 나눈 간단한 궤적계획 알고리즘과 자코비안을 이용



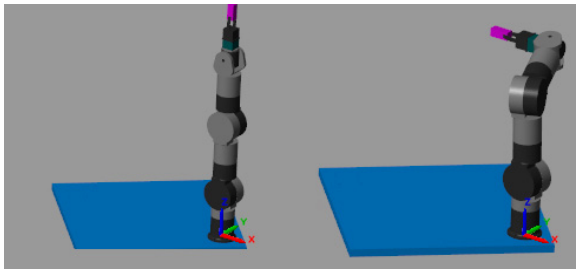

[Fig. 2] Hardware specifications of 7-axis robot (RoMAN 7)

[Table 2] Hardware specifications of 7-axis robot (RoMAN)

| | | | |
|---|---|---|---|
| 1st-length | 0.278 m | 1st-mass | 5 kg |
| 2nd-length | 0 m | 2nd-mass | 4 kg |
| 3rd-length | 0.425 m | 3rd-mass | 5 kg |
| 4th-length | 0 m | 4th-mass | 4 kg |
| 5th-length | 0.425 m | 5th-mass | 5 kg |
| 6th-length | 0 m | 6th-mass | 3 kg |
| 7th-length | 0.072 m | 7th-mass | 2 kg |




[Fig. 3] Block diagram of control system based on Q-learning



[Fig. 4] Initial position and desired position of end effector

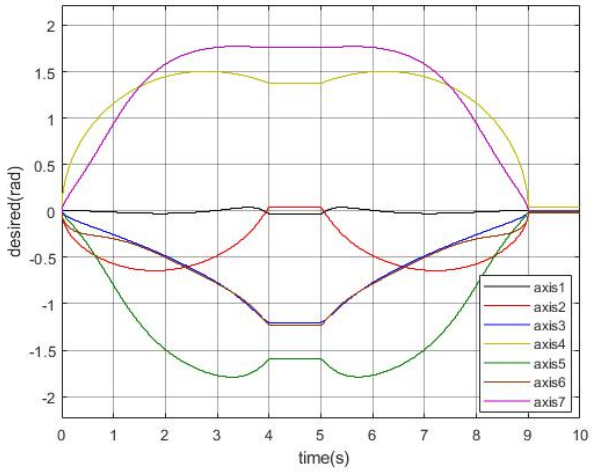
[Table 3] The parameters of control & reinforcement learning

| | |
|---|---|
| Control Parameter | Sampling time=2 ms, $\epsilon_c$=2, $\epsilon_o$=1, $\alpha_1$=0, $\alpha_2$=0 |
| RL Parameter | $\rho$=0.82, $\gamma$=0.96, $\sigma$=0.02, $g$=0.05 |

한 역기구학이 각 축의 목표 각도를 생성하고 이후 SMCSPO 제어기가 목표 각도를 받아 로봇을 제어할 수 있도록 하였다. 로봇이 목표위치로 이동하였다가 초기위치로 돌아가도록 궤적을 설정하였고, 한번의 왕복 이후 강화학습 튜너가 제어게인을 수정하고 다시 작업을 수행하도록 설정하였다.

[Table 3]은 시뮬레이션에서 사용되는 $\lambda_d$를 제외한 제어 파라미터와 강화학습 파라미터를 보여준다. 두 종류의 파라미터들은 시행착오적인 방법으로 설정되었다. 제어 파라미터는 실제 로봇에 적용되는 파라미터를 적용시켰으며, 강화학습 파라미터는 시뮬레이션을 구성한 후 강화학습 튜닝 알고리즘을 적용하면서 파라미터를 직접 설정하였다.

시뮬레이션 조건에서는 각 링크의 초기 $\lambda_d$를 1로 설정하고 동일한 모션을 500회 반복하여 한번의 반복마다 제어게인이 갱신되도록 하였다. 한번의 반복동작은 10초동안 [Fig. 4]의 오른쪽 그림과 같이 로봇의 말단부를 로봇의 중심에서부터 Cartesian 좌표계로 x, y, z를 각각 -500 mm, 400 mm, 800 mm로 roll, pitch, yaw를 0°, -90°, 0°로 이동시켰다가 다시 초기 위치로 복귀하도록 하였다. 이때의 각축의 궤적은 [Fig. 5]와 같다. 시뮬레이션을 통해 알고리즘이 로봇암의 제어시스템에 적합함을 관찰하고, 같은 모션에 대해 강화학습 알고리즘을 사용하였을 때의 제어게인과 알고리즘을 사용하지 않은 제어게




[Fig. 5] Each desired angle for simulation trajectory

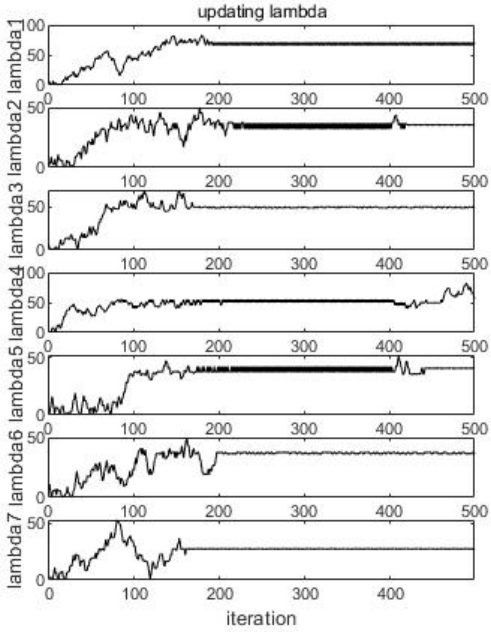
인의 성능을 비교하였다. 그리고 399회 이후의 시뮬레이션에서는 환경의 변화를 가정하여 각 축의 구동부에 60 Nm의 토크를 지속적으로 적용하였다. 이는 외란을 가함으로 알고리즘이 설정한 제어게인이 더 이상 최적이 아니라고 판단할 경우 다른 제어게인을 찾을 수 있는지를 관찰하였다. 이는 시뮬레이션으로 학습을 완료한 후 실제 시스템에 바로 적용할 경우 가상환경과 실제 환경의 차이가 있을 수 있기 때문에 그 차이를 외란의 형태로 가정하고 시뮬레이션을 진행하였다.

## 5. 시뮬레이션 결과 및 분석

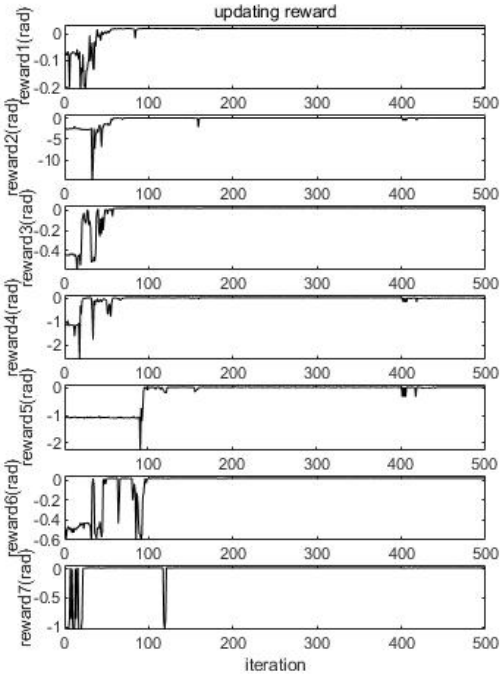
먼저 시뮬레이션 중 각 축의 $\lambda_d$의변화를 관찰하였다. [Table 4]는 반복회수 399회, 500회에 대한 각 축의 $\lambda_d$의 값을 나타내고, [Fig. 6]에서는 반복 횟수에 따른 $\lambda_d$의 변화를 나타낸다. 2축을 제외한 대부분의 제어게인들이 200회 반복 이후 수렴하는 것을 확인할 수 있었으며 강화학습 튜닝 알고리즘에서 제어게인을 유지하는 행동을 설정하지 않았기 때문에 최종 값에 도달한 이후 제어게인이 조금씩 떨리는 것을 볼 수 있었다. 외

[Table 4] Each control gains $\lambda_d$ for each axis

| iteration | Non-RL | 1 (initial) | 399 | 500 |
|---|---|---|---|---|
| $\lambda_{d1}$ | 42 | 1 | 66 | 71 |
| $\lambda_{d2}$ | 46 | 1 | 37 | 36 |
| $\lambda_{d3}$ | 36 | 1 | 49 | 48 |
| $\lambda_{d4}$ | 66 | 1 | 55 | 64 |
| $\lambda_{d5}$ | 60 | 1 | 37 | 41 |
| $\lambda_{d6}$ | 36 | 1 | 37 | 36 |
| $\lambda_{d7}$ | 20 | 1 | 27 | 28 |




[Fig. 6] updated control gain for SMCSPO




[Fig. 7] updated reward for auto-tuner

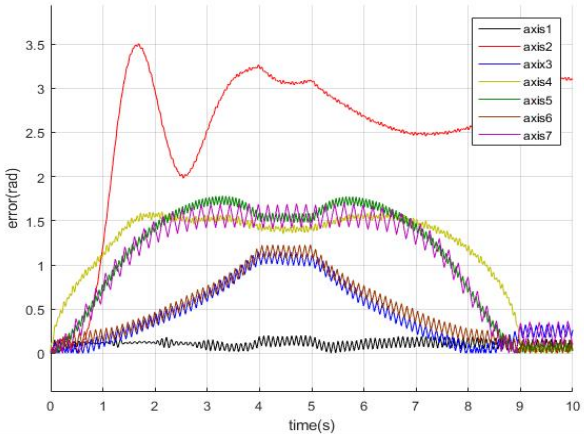
란이 발생한 400회 이후에는 2, 4, 5축의 제어게인이 수정되는 것을 확인할 수 있었으며 이는 작업도중 조건이 달라지는 경우에도 바로 환경에 적응하는 것을 볼 수 있었다. 이러한 제어게인의 변화로 보상 값이 최대가 되는 것을 보여준다.

[Fig. 7]에서는 반복횟수에 따른 보상 $r_t$를 보여준다. $r_t$는 한 반복에서의 평균 추종오차 $J_t$와 관련이 있어 높을수록 좋은 추종성능을 보여주는 정량적 지표로 사용될 수 있다. 약 150회

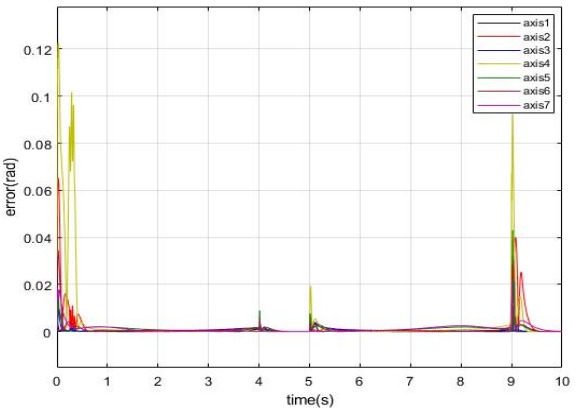
의 반복 이후 2, 5축을 제외한 모든 축이 0에 가까운 보상을 얻었으며, 170회 이후에는 2, 5축 또한 높은 보상을 얻는 것을 확인할 수 있었다. 이후 제어게인이 수렴하는 399회 반복까지 보상 $r_t$가 0보다 큰 값을 가지는 것을 확인하였다. 그리고 [Table 4]에서 알고리즘을 적용한 제어게인과 기존에 사용하던 제어게인의 성능을 비교하였다. 데이터가 수렴했을 때의 399번째의 각 축의 제어게인의 보상 $r_{399}$의 총합은 약 0.1305 rad이고, 강화학습을 사용하지 않고 시행착오적인 방법으로 얻어낸 각 축의 제어게인의 보상 $r_{Non-RL}$의 총합은 약 0.1301 rad이다. 0.0004 rad의 차이로 거의 비슷한 추종오차를 가지지만 약간 더 높은 보상을 얻었으며, 인공지능을 통해 7 축까지의 제어게인을 자동적으로 얻어낼 수 있었다.

다음으로 400회 이후 반복에서 60 Nm의 외란에 대해 다시 2, 4, 5축에서의 보상 값이 일시적으로 감소하고 다시 복구되는 것을 확인할 수 있었다. 각 축의 보상이 순간적으로 감소할 때도 -0.3 rad보다 낮아지지 않았기 때문에 실제 로봇에 적용한다고 가정할 때 크게 위험을 주는 동작은 하지 않는 것으로 판단할 수 있다. 나머지 축들은 외란을 받았으나 높은 강인성으로 인해 추종오차가 크게 변하지 않았고, 가까운 제어게인의 가치보다 현재 제어게인의 가치가 높았기 때문에 제어게인이 변하지 않았다.

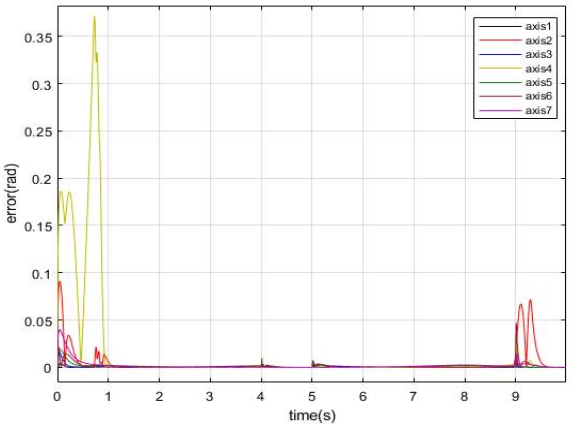
[Fig. 8]은 $\lambda_d$가 모두 1로 설정된 첫번째 사이클인 초기 상태의 추종오차의 절대값이며, 목표 각도가 그대로 추종 오차에 드러나는 것처럼 정상적인 제어가 되지않는 것을 보여준다. 이후 [Fig. 9]에서 나타낸 것처럼 각축의 $\lambda_d$가 적절한 값에 수렴한 399회의 반복에서는 추종오차가 거의 없어진 것을 볼 수 있다. 로봇이 출발하고 도착할 때의 급격한 움직임의 변화로 진동이 발생하여 약간의 오차가 발생하는 것을 확인할 수 있었고, 그 이외의 경우에는 매우 작은 추종오차를 가지는 것을 관찰할 수 있었다. 진동 또한 0.5초 이내의 빠른 시간에 억제되고, 이외의 시간에서 추종오차가 거의 0에 가까운 값을 유지하




[Fig. 8] tracking error in initial control gain




[Fig. 9] tracking error for control gain in 399 iterations




[Fig. 10] tracking error for control gain in 500 iterations

는 것으로 비선형요소인 로봇을 제어하는 동안 적절한 추종성능과 안정성을 가진다고 할 수 있다. [Fig. 10]의 500회째의 반복에서는 시작부터 외란에 영향을 받기 때문에 399회의 반복보다 4축에서의 진동이 크게 발생하는 것을 볼 수 있었으나 1초 이후에는 거의 차이가 없는 성능을 보였다. 5장에서는 간단한 시뮬레이션을 통해 강화학습 Q-learning이 기존의 제어시스템에 적용되어 반복적으로 로봇이 동작할 때 모션의 추종오차가 0에 가까워지도록 제어게인을 설정하는 것을 관찰하였다.

## 6. 결 론

본 연구에서는 기존의 SMCSPO제어기에 강화학습 Q-learning을 적용시켜 설계자가 직접 제어게인을 설정하지 않고 강화학습 모델이 학습하며 적절한 제어게인을 찾게 하는 알고리즘을 제안하였다. Q-learning에 의해 튜닝알고리즘을 적용시킨 7축 로봇팔 제어시스템을 시뮬레이션으로 구현하고 Q-learning을 통해 자동적으로 얻은 제어게인과 시행착오적으로 설정한 제어게인의 성능을 비교하여 Q-learning이 로봇을 제어하는 SMCSPO 게인튜닝에 적합한지를 확인하였고, 외란을 통한

시뮬레이션으로 실제환경으로 이전가능성을 확인하였다. 이 방법은 유전 알고리즘으로 제어게인을 튜닝하는 것에 비해 적은 데이터로 제어게인을 결정할 수 있었고, 배열형태의 Q-table과 이를 이용한 행동결정은 간단한 구조를 가져 유전알고리즘보다 구현하기가 용이하다는 것이 확인되었다. 또 다른 장점으로는 약간의 설정변경으로 SMCSPO뿐만 아닌 다른 제어기의 제어게인을 결정하는데도 사용 가능하다.

본 연구에서는 기본적인 강화학습 방법 Q-learning을 사용하였지만 DPG[19], DQN[20]등의 고성능 강화학습 모델을 사용하면 더 좋은 성능을 기대할 수 있다. 또다른 개선이 필요한 부분으로 탐색방법이 효율적이지 못하다는 부분이 있다. 학습 초기단계에서 무작위한 탐사방법에 의해 제어게인이 수렴할 때까지 탐사하지 못한 상태가 존재할 수 있고 이미 탐색한 상태에 대해서도 다시 탐색하여 불필요한 작업을 할 가능성이 있다. UCB (Upper Confidence Bound)[21] 등의 다른 탐색알고리즘을 적용할 경우 적은 반복 횟수로 효율적인 탐색을 수행할 수 있을 것으로 생각 되지만, 본 연구에서는 강화학습의 적용을 위한 초기 연구로 시뮬레이션으로만 진행하였다. 향후 연구에서는 이러한 문제점을 해결하고 실제 로봇에 적용할 계획이다.

## References


[1] S. H. Han, H. C. Cho, and K. S. Lee, "Position Control of Nonlinear Crane Systems using Dynamic Neural Network," *Trans. Korean. Inst. Elect. Eng.*, vol. 56, no. 5, pp. 966-972, 2007, [Online], https://www.dbpia.co.kr/journal/articleDetail? nodeId=NODE01280780.

[2] K. D. Young, V. I. Utkin, and U. Ozguner, "A control engineer's guide to sliding mode control," *IEEE Transactions on Control Systems Technology*, vol. 7, no. 3, pp. 328-342, 1999, DOI: 10.1109/87.761053.

[3] V. Utkin and H. Lee, "Chattering Problem in Sliding Mode Control Systems," *International Workshop on Variable Structure Systems, 2006. VSS'06*, Alghero, Italy, pp. 346-350, 2006, DOI: 10.1109/VSS.2006.1644542.

[4] J. T. Moura, H. Elmali, and N. Olgac, "Sliding Mode Control with Sliding Perturbation Observer," *ASME. J. Dyn. Sys., Meas., Control*, vol. 119, no. 4, pp. 657-665, 1997.

[5] M. G. Jung and M. C. Lee, "Study on Robust Control of Industrial Manipulator for Assembly Based on SMCSPO," *Journal of Institute of Control, Robotics and Systems*, vol. 24, no. 6, pp. 552-560, 2018, DOI: 10.5302/J.ICROS.2018. 18.0034.

[6] K.-G. Cha, S. M. Yoon, and M. C. Lee, "SPO based Reaction Force Estimation and Force Reflection Bilateral Control of Cylinder for Tele-Dismantling," *Journal of Korea Robotics Society*, vol. 12, no. 1, pp. 1-10, March, 2017, DOI: 10.7746/jkros.2017.12.1.001.

[7] T.-C. Kuo, Y.-J. Huang, and S.-H. Chang, "Sliding mode control with self-tuning law for uncertain nonlinear systems," *ISA Transactions*, vol. 47, no. 2, pp. 171-178, April, 2008, DOI: 10.1016/j.isatra.2007.10.001.

[8] K. S. You, M. C. Lee, and W. S. Yoo, "Sliding mode controller with sliding perturbation observer based on gain optimization using genetic algorithm," *KSME International Journal*, vol. 18, no. 4, pp. 630-639, 2004, DOI: 10.1007/BF02983647.

[9] C. Szepesvári, "Algorithms for reinforcement learning," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 4, no. 1, pp. 1-103, 2010, DOI: 10.2200/S00268ED1V01Y201005AIM009.

[10] Y. Yu, Z. Cao, S. Liang, Z. Liu, J. Yu, and X. Chen, "A Grasping CNN with Image Segmentation for Mobile Manipulating Robot," *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Yunnan, China, pp. 1688-1692, 2019, DOI: 10.1109/ROBIO49542.2019. 8961 427.

[11] H. H. Kim, H. Khan, Y. J. An, and M. C. Lee, "Development of Reinforcement Learning Assembly Algorithm Based on Estimated Reaction Force Using Sliding Perturbation Observer," *International Conference on Control, Automation and System*, Busan, Korea, pp. 1018-1021, 2020, [Online], https://www.dbpia.co.kr/Journal/articleDetail?nodeId=NODE10493699.

[12] Y. Ansari, E. Falotico, Y. Mollard, B. Busch, M. Cianchetti, and C. Laschi, "A Multiagent Reinforcement Learning approach for inverse kinematics of high dimensional manipulators with precision positioning," *2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, University Town, Singapore, pp. 457-463, 2016, DOI: 10.1109/BIOROB.2016.7523669.

[13] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement Learning and Feedback Control: Using Natural Decision Methods to Design Optimal Adaptive Controllers," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76-105, Dec. 2012, DOI: 10.1109/MCS.2012.2214134.

[14] W. J. Shipman and L. C. Coetzee, "Reinforcement Learning and Deep Neural Networks for PI Controller Tuning," *IFAC-Papers OnLine*, vol. 52, no. 14, pp. 111-116, 2019, DOI: 10.1016/j.ifacol.2019.09.173.

[15] C. J. Watkins and P. Dayan. "Q-learning," *Machine Learning*, vol. 8, pp. 279-292, 1992, [Online], http://www.gatsby. ucl. ac.uk/~dayan/papers/wd92.html.

[16] J. M. Hollerbach, "A Recursive Lagrangian Formulation of Maniputator Dynamics and a Comparative Study of Dynamics Formulation Complexity," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 10, no. 11, pp. 730-736, Nov., 1980, DOI: 10.1109/TSMC.1980.4308393.

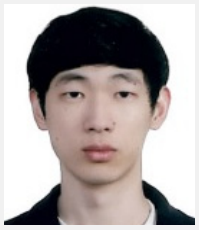
[17] J.-J. Slotine, J. K. Hedrick, and E. A. Misawa, "On Sliding Observers for Non-Linear Systems," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 109, no. 3, pp. 245-252, 1987, DOI: 10.23919/ACC.1986.4789217.

[18] E. Rodrigues Gomes and R. Kowalczyk. "Dynamic analysis of multiagent Q-learning with ε-greedy exploration," Proceedings of the *26th Annual International Conference on Machine Learning (ICML '09). Association for Computing Machinery*, New York, USA, pp. 369-376, 2009, DOI: 10.1145/1553374.1553422.

[19] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic Policy Gradient Algorithms," *31st International Conference on Machine Learning, Proceedings of Machine Learning Research*, vol. 32, no, 1, pp. 387-395, 2014, [Online], http://proceedings.mlr.press/v32/silver14.html.

[20] T. Hester, M. Vecerik, O. Pietquin, M. Lanctot, T. Schaul, B. Piot, D. Horgan, J. Quan, A. Sendonaris, G. Dulac-Arnold, I. Osband, J. Agapiou, J. Z. Leibo, and A. Gruslys, "Deep q-learning from demonstrations," *arXiv:1704.03732 [cs.AI]*, 2018, DOI: 10.48550/arXiv.1704.03732.

[21] W. Jouini, D. Ernst, C. Moy and J. Palicot, "Upper Confidence Bound Based Decision Making Strategies and Dynamic Spectrum Access," *2010 IEEE International Conference on Communications*, Cape Town, South Africa, pp. 1-5, 2010, DOI: 10.1109/ICC.2010.5502014.



**이 진 혁**

2020 동의대학교 메카트로닉스공학과 (공학사)

2020~현재 부산대학교 대학원 기계공학부 (석사과정 재학)

관심분야: 강화학습, 강인제어, 로보틱스, Data-driven-control

**김 재 형**

2019 부산대학교 대학원 기계공학부 (공학사)

2021 부산대학교 대학원 기계공학부 (공학석사)

2021~현재 부산대학교 대학원 기계공학부 (박사과정 재학)

관심분야: 지능로봇제어, 기구학, 기계학습, 자율로봇, 시뮬레이션

**이 민 철**

1983 부산대학교 기계공학과(공학사)

1988 쯔쿠바대학교 이공학연구과(공학석사)

1991 쯔쿠바대학교 물리공학연구과 (공학박사)

2000.8~2001.8 노스캐롤라이나 주립대학교 (NCSU) 방문교수

2009.8~2010.8 퍼듀대학교 방문교수

1991~현재 부산대학교 기계공학부 교수

관심분야: 시스템 규명, 로봇제어, 의료로봇, 메카트로닉스