

# DDPG 알고리즘을 이용한 양팔 매니플레이터의 협동작업 경로상의 특이점 회피 경로 계획

## Singularity Avoidance Path Planning on Cooperative Task of Dual Manipulator Using DDPG Algorithm

이종학<sup>1</sup>·김경수<sup>2</sup>·김윤재<sup>2</sup>·이장명<sup>†</sup>

Jonghak Lee<sup>1</sup>, Kyeongsoo Kim<sup>2</sup>, Yunjae Kim<sup>2</sup>, Jangmyung Lee<sup>†</sup>

**Abstract:** When controlling manipulator, degree of freedom is lost in singularity so specific joint velocity does not propagate to the end effector. In addition, control problem occurs because jacobian inverse matrix can not be calculated. To avoid singularity, we apply Deep Deterministic Policy Gradient(DDPG), algorithm of reinforcement learning that rewards behavior according to actions then determines high-reward actions in simulation. DDPG uses off-policy that uses  $\epsilon$ -greedy policy for selecting action of current time step and greed policy for the next step. In the simulation, learning is given by negative reward when moving near singularity, and positive reward when moving away from the singularity and moving to target point. The reward equation consists of distance to target point and singularity, manipulability, and arrival flag. Dual arm manipulators hold long rod at the same time and conduct experiments to avoid singularity by simulated path. In the learning process, if object to be avoided is set as a space rather than point, it is expected that avoidance of obstacles will be possible in future research.

**Keywords:** Dual Armed Manipulator, Singularity, Reinforcement Learning, DDPG

### 1. 서 론

매니플레이터는 공장에 적용되어 조립, 이송, 가공 등의 작업을 수행<sup>[1]</sup>하는 것부터 시작하여 4차 산업혁명과 맞물리면서 인공지능이 적용된 시스템에 대한 수요와 보급이 증가하고 있다. 매니플레이터의 작업은 장애물 회피, 목적지 도착으로 구성되어 있기 때문에 움직이는 경로를 설계하는 것은 매우 중요하며<sup>[2]</sup> 이것이 제대로 이루어지지 않는다면 그 효용성이 사라지게 된다. 이 때, 매니플레이터의 경로를 설계하고 제어하면서 문제가 되는 것이 바로 특이점이다<sup>[1-6]</sup>. 매니플레이터를 제어할 때 자코비안이 사용되는데, 특이점에서는 자코비안의

역행렬이 계산되지 않기 때문에 제어에 오류를 야기한다. 특이점을 회피하기 위해 Nakamura<sup>[7]</sup>와 Wampler<sup>[8]</sup>는 자코비안의 성분에 조정항을 추가하는 방법을 사용하였고 이를 Singularity Robust Inverse 혹은 Damped Least Squares 방법이라고 한다. 이는 의사 역행렬이 항상 존재한다는 장점이 있지만<sup>[3]</sup> 조정항으로 인해 진동의 위험이 있는 단점이 있다. 또한 Whitney<sup>[9]</sup>는 관성 행렬을 가중치로 사용하여 에너지를 최소화<sup>[10]</sup> 시킴으로써 기구학해를 최적화하는 Weighted Least Square 방법을 사용하였다. 그리고 2000년 이후로는 자코비안의 의사역행렬을 대신하여 자코비안의 전치행렬로 제어하는 Jacobian Transpose<sup>[11]</sup> 방법이 제안되었다. 이 방법은 계산 시간이 오래 걸리는 특이값 분해가 필요하지 않아 고속 연산에 유리하고 시스템의 안정성이 검증되었다는 장점이 있지만 추종 오차가 다른 알고리즘에 비해 크다는 단점이 있다<sup>[3]</sup>.

최근에는 Chung 등<sup>[12]</sup>이 직선 경로상에 특이점이 발생할 경우<sup>[12]</sup> 일정 영역을 설정한 후 해당 영역 안에서 회피 모션을 수행하여 오리엔테이션의 오차 발생을 최소화하는 연구를 진행하였다. 회피 모션을 수행함으로써 특이점으로 인한 모터의

Received : Dec. 21. 2020; Revised : Feb. 2. 2021; Accepted : Feb. 26. 2021

※ This research is based upon work supported by the Ministry of Trade, Industry & Energy (MOTIE, Korea) under Industrial Technology Innovation Program. No. 10073147

1. MS Student, Pusan National University, Busan, Korea (jonghak1696@pusan.ac.kr)

2. BS Student, Pusan National University, Busan, Korea (rudtn95423, onsis4@pusan.ac.kr)

† Professor, Corresponding author: Electronics Engineering, Pusan National University, Busan, Korea (jmlee@pusan.ac.kr)

급격한 각도 변화를 방지한다는 장점이 있었지만 특이점 영역을 얼마나 설정해야 좋은지 결정하기 위해서는 반복적 실험을 통해야 하는 번거로움이 있다. 따라서 이 회피 영역을 로봇 스스로 결정하고 이에 따라 회피 경로를 생성하는 연구가 필요하다.

본 논문에서는 특이점 주변을 회피해야 할 대상으로 몸과 동시에 목적지까지 엔드 이펙터가 이동하는 경로를 생성하기 위해 미로 찾기 같은 경로 생성에서 많이 이용되는 강화학습을 이용한다. 그중에서도 Deep Deterministic Policy Gradient (DDPG, 심층 결정론적 정책 강화법) 알고리즘을 사용하였다. DDPG 알고리즘은 액터-크리틱 신경망이 존재해서 행동과 그 행동의 결과인 상태의 좋고 나쁜 정도를 계산하고 보상을 통해 신경망 업데이트하고 다음 행동을 개선한다. 이를 매니플레이터 시스템에 적용하면 행동은 엔드 이펙터의 움직임, 상태는 3차원 공간 상의 엔드 이펙터 위치 그리고 길 찾기를 특이점 회피 후 목적지 도착으로 치환하여 생각할 수 있겠다. 이는 Phaniteja 등<sup>[13]</sup>은 강화학습을 이용한 특이점 회피를 위해 상태로 조인트 각도, 행동으로 조인트의 각속도를 설정한 것보다 훨씬 직관적이라는 장점이 있다. 즉, 엔드 이펙터가 목적지까지 멀거나 특이점을 유발하는 위치 근처면 나쁜 상태, 목적지에 가까워지면 좋은 상태로 판단하면서 일부는 직선 경로로 움직이다가 특이점 근처에서만 회피 동작을 하는 경로 생성과 회피 영역의 크기를 로봇 스스로 설정하는 것이 목적이다.

DDPG 알고리즘은 신경망 갱신을 위해 보상식을 설정할 수 있는데<sup>[14,16]</sup>, 본 논문에서는 조인트와 3차원 공간 사이의 속도 전파성을 나타내는 매니플러빌리티를 추가하였다. Chung<sup>[1]</sup> 등의 연구에서 제안한 회피 영역의 크기를 결정하기 위해 속도 전파성도 고려하기 위함이다. 보상식에는 여러 인자를 추가할 수 있는데, 학습의 결과로 모든 인자들을 종합적으로 고려한 경로를 생성하게 된다. 결국 사람이 직접 정하거나 반복적 실험을 통해 결정해야 하는 회피 영역의 크기를 로봇 스스로 결정할 수 있게 된다.

본 논문에서 사용한 플랫폼은 총 12자유도의 양팔 로봇으로써 이는 물체를 옮기는 작업에 사용하기가 좋으며 그 중에서도 물건을 위에서 아래로 내리는 작업을 사람 대신에 수행하기 좋다. 따라서 양팔 매니플레이터가 긴 막대를 위에서 아래로 내리는 직선 경로 제어 시 특이점이 발생하는 상황을 확인하고 DDPG 알고리즘을 통하여 학습의 결과로 대체 경로를 생성한다. DDPG를 포함하는 개념인 강화학습을 이용한 이전의 연구<sup>[17]</sup>는 목표점 도착 후 물체를 잡기 위해 엔드 이펙터의 정보를 x, y, z 위치 정보만 주었다. 따라서 여러 가지의 도착 자세가 나왔고 그 중에는 물체를 잡기에 불안정한 자세도 포함되어 있었다. 반면 본 논문에서는 위치뿐만 아니라 물체에 변형이 가해지지 않는 범위 내에서 방위 정보도 추가하여 학습 모델을 구성하였기 때문에 일정한 자세를 유지할 수 있다.

## 2. 양팔 매니플레이터 해석

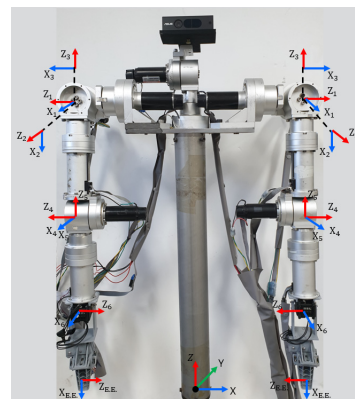
### 2.1 양팔 매니플레이터의 구조

본 연구에 사용한 매니플레이터는 [Fig. 1]과 같이 사람이 서 있는 형상을 하고 있으며 한 팔에 6 DOF (Degree Of Freedom, 자유도)씩 총 12 DOF이다. 기준이 되는 좌표계는 [Fig. 1] 하단의 X, Y, Z 축이 모두 표시된 좌표계이다.

좌표계 배치에 따라 DH table을 작성하면 [Table 1]과 같다. 오른팔과 왼팔의  $\theta, d, a$  파라미터는 같으며  $\alpha$ 만 다르다.  $\alpha$  부분에서 괄호 밖이 오른팔, 괄호 안이 왼팔의 파라미터이다.

### 2.2 수행 작업

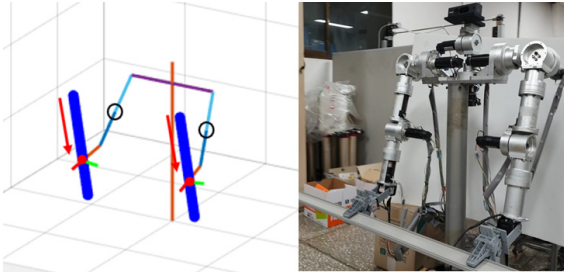
양팔 매니플레이터를 이용하여 여러 작업을 할 수 있는데, 그 중에서 하나의 물체를 두 개의 매니플레이터로 동시에 내리는 작업을 할 수 있다. 이 작업을 시작점과 도착점의 최단 경로인 직선 경로로 움직였을 때 3장에서 설명할 특이점이 발생하는 경우를 발견하였고 직선경로를 그대로 추종할 시 제어에 오류가 발생할 위험이 있다<sup>[18]</sup>. 특이점 발생 여부는 3.3절에서 설명할 매니플러빌리티를 경로의 점마다 계산하여 판단하였



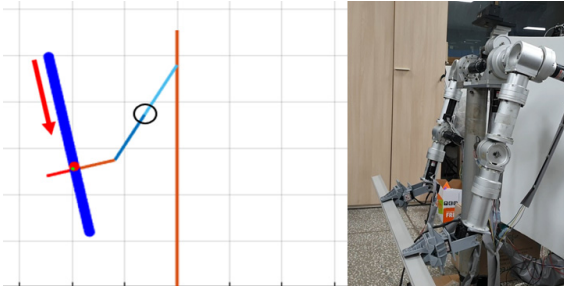
[Fig. 1] Dual 6 DOF manipulator

[Table 1] DH table of right and left manipulator

	$\theta$	$d$	$a$	$\alpha$
1	$\theta_1^*$	0	0	$-\pi/2 (\pi/2)$
2	$\theta_2^*$	0	0	$\pi/2 (-\pi/2)$
3	$\theta_3^*$	-25.1	0	$\pi/2 (-\pi/2)$
4	$\theta_4^*$	0	0	$-\pi/2 (\pi/2)$
5	$\theta_5^*$	-22.7	0	$-\pi/2 (\pi/2)$
6	$\theta_6^*$	0	17	0



[Fig. 2] Singularity occurrence configuration



[Fig. 3] Singularity occurrence configuration (side view)

고 이 상황을 [Fig. 2]와 [Fig. 3]에 나타내었다.

[Fig. 2]와 [Fig. 3]의 굵은 파란 선은 위에서 아래로 내려오는 경로를 의미하고 엔드 이펙터의 빨간 점은 직선 경로상에서 특이점이 발생한 지점이며 왼쪽은 시뮬레이션, 오른쪽은 실제 구동 장면이다. 검은색 원으로 표시한 위치의 4번째 조인트의 각도가 0이 되면서 3번째와 5번째의 조인트 회전축이 정렬되고 매니퓰러빌리티가 0이 되면서 특이점이 발생하였다. 이에 관한 자세한 사항은 3장에서 설명한다.

### 3. 자코비안과 특이점

#### 3.1 자코비안

매니퓰레이터의 조인트 공간에서 속도를  $\dot{\theta}$ 라 하고 3차원 공간에서 엔드 이펙터의  $x, y, z$ 에 대한 선속도, 각속도의 벡터를  $\dot{X}$ 라 했을 때  $\dot{\theta}$ 와  $\dot{X}$ 는 식 (1)와 같은 관계를 가지며 이 때,  $J(\theta)$ 를 자코비안이라고 한다.

$$\dot{X} = J(\theta)\dot{\theta} \quad (1)$$

이 때  $\dot{X}$ 는 위치와 방위의  $\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}$  6차원 속도 벡터이고,  $\dot{\theta}$ 는 6 DOF 조인트의 6차원 벡터이다. 따라서  $J(\theta)$ 는  $6 \times 6$  행렬이다.

제어대상은 조인트이기 때문에 식(1)은 식(2)와 같이 변형된다.

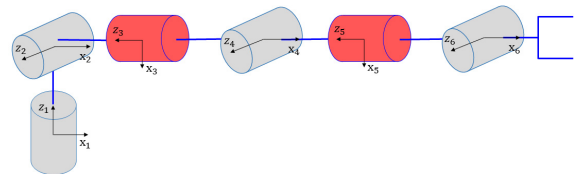
$$\dot{\theta} = J(\theta)^{-1}\dot{X} \quad (2)$$

자코비안의 역행렬이 존재하기 위해서는  $|\det(J(\theta))|$ 가 존재해야 하는데, 이는 행렬의 특잇값(singular value)의 곱으로 표현될 수 있다.  $\lambda_i$ 는 자코비안의 특잇값으로써 특잇값 분해를 통해 구할 수 있다.

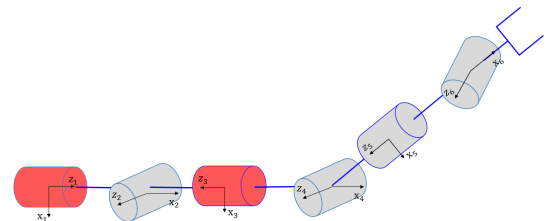
$$|\det(J(\theta))| = \prod_{i=1}^6 \lambda_i \quad (3)$$

#### 3.2 특이점

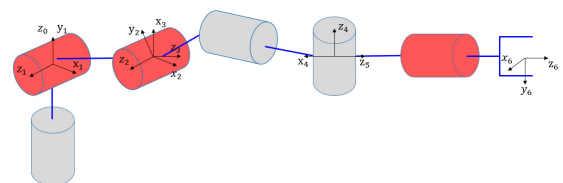
만약 자코비안의 특잇값 중 하나 이상이 0이 되면  $|\det(J(\theta))|$ 는 0이 되어 자코비안의 역행렬이 존재하지 않아 식 (2)에 따라 매니퓰레이터 제어를 할 수 없게 된다. 이 상태를 특이점이라 규정한다<sup>[4]</sup>. 특이점이 발생할 수 있는 기구학적 자세<sup>[12]</sup>를 [Fig. 4] ~ [Fig. 6]에 나타냈다. 특이점을 유발하는 원인이 되는 조인트를 빨간색으로 표시하였다. 이 중 [Fig. 4]의 손목 특이점은 2.2의 수행 작업 도중 발생할 수 있지만 [Fig. 5]의 어깨 특이점은 1축과 3축이 정렬되어야 하기 때문에 [Fig. 1]과 함께 참고하면 팔을 벌리고 있는 모습이 됨을 알 수 있다. 따라서 2.2의 작업 도중 나타날 수 없다. [Fig. 5]의 팔꿈치 특이점은 평행한 두 회전축이 만드는 평면상에 다른 회전축이 존재하면 발생하는데, [Fig. 1]을 보면 본 논문에서 사용한 매니퓰레이터는 1축부터 6축까지 순차적으로 회전축이 직교하기 때문에 모터의 배치 방향의 다름으로 인해 발생할 수 없다.



[Fig. 4] Wrist singularity posture



[Fig. 5] Shoulder singularity posture



[Fig. 6] Elbow singularity posture





즉, 학습 과정에서 엔드 이펙터가 여러 방향으로 움직여보면서 현재 위치가 특이점 발생 위치에 멀면서 목표점에 가까우면 좋은 상태, 그 반대이면 나쁜 상태로 판단하고 행동을 도출하는 신경망의 가중치를 개선하기 때문에 원하는 경로를 생성할 수 있을 것이라고 예상하였다. 5장의 시뮬레이션 시스템 역시 이와 같이 구성하였다.

## 5. 문제 상황 및 시뮬레이션 시스템 구성

### 5.1 문제 상황

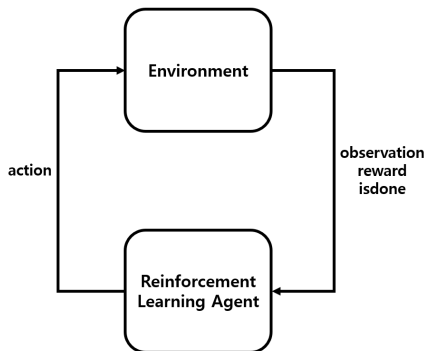
[Fig. 8]과 같이 긴 막대를 두 개의 매니플레이터로 동시에 잡아 오른쪽 사진처럼 목표점으로 움직이는 일을 할 때 직선 경로로 움직일 시 [Fig. 2]에서처럼 빨간 점에서  $\theta_4 = 0$ 이 되면서 특이점이 발생한다. 따라서 특이점이 발생하는 점을 피해가는 경로 설계를 자동으로 하기 위한 시뮬레이션 시스템을 구축한다.

### 5.2 시뮬레이션 시스템

경로 설계 시뮬레이션을 위해 SIMULINK로 [Fig. 9]와 같이 시뮬레이션 DDPG 시스템을 구성하였다. 환경(environment)은 로봇 시스템으로써 행동(action)을 받아 움직이고 에이전트(agent)의 내부에는 액터-크리틱 신경망이 있다. 로봇의 상태를 알 수 있는 파라미터인 observation, 목적에 부합하는 상태인지 알 수 있는 isdone, 1회 행동에 따른 보상 값인 reward를 에이전트



[Fig. 8] Starting point and goal point of rod transfer task

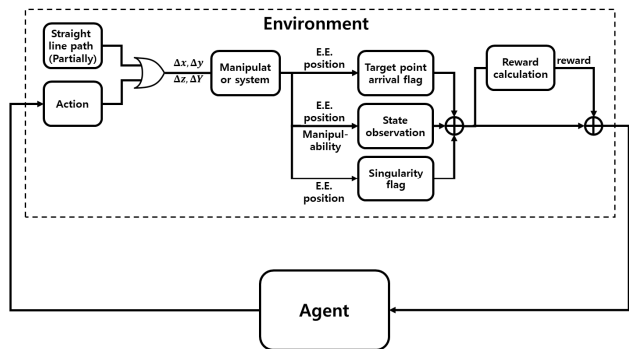


[Fig. 9] DDPG environment and agent

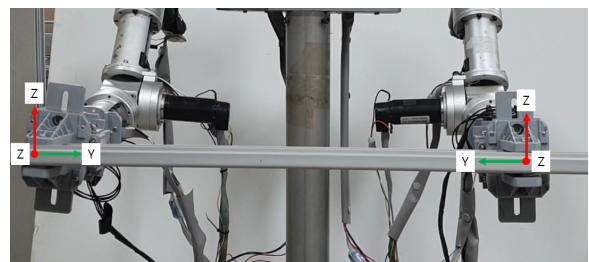
로 보낸다. 에이전트는 다시 이 값들은 받은 신경망에 대입하여 행동을 계산한다. 한 번 행동을 취할 때를 스텝이라고 하는데, 지정한 스텝 수대로 에이전트-환경 사이클을 반복하거나 목표에 도달하면 하나의 에피소드가 끝나며 이를 반복하여 학습한다.

### 5.3 시뮬레이션 시스템 - 환경

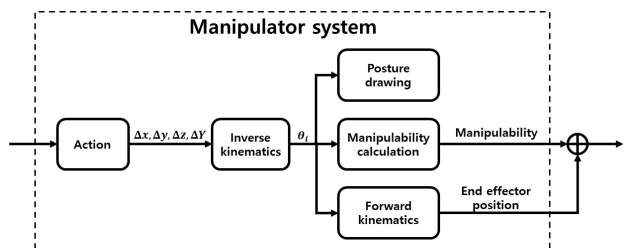
[Fig. 10]은 환경(environment) 내부의 시스템을 보여준다. 먼저 일정 부분 직선 경로로 매니플레이터가 움직이고 회피 대상인 특이점 근처에 다다르면 에이전트로부터 행동을 받는다. 이전 상태에서부터 x, y, z축으로 조금씩 이동하며 목표점에 도착해야하기 때문에 행동 인자로 엔드 이펙터의  $\Delta x, \Delta y, \Delta z$ 를 지정했고, [Fig. 11]과 같이 ZYZ Euler angle중 위로 솟은 Z축으로 회전하면 물체를 안전하게 잡을 수 없고, 화면 앞으로 나오는 Z축으로 회전하면 물체와 그리퍼에 힘이 가해져 변형이 일어날 수 있다. 따라서 Euler angle 중 Y 축의 회전 변화량( $\Delta Y$ )만 행동 인자로 추가하였다.



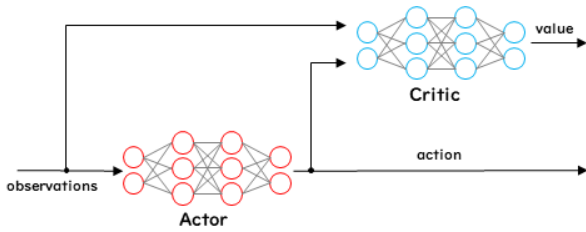
[Fig. 10] Environment system



[Fig. 11] ZYZ Euler angle and rod grasping



[Fig. 12] Manipulator system



[Fig. 13] Actor-Critic network structure

[Fig. 12]는 환경 내부의 매니플레이터 시스템을 보여준다. 매니플레이터 시스템에서 행동을 받아 움직이고 난 후 엔드 이펙터의 좌표와 매니플러빌리티를 결과로 내놓는다. 그런 다음 목표점에 도착했는지, 현재 위치가 어디인지, 특이점에 빠졌는지 판단하는 정보와 이 데이터로 계산한 보상을 에이전트로 넘겨준다. 이 과정이 한 번의 스텝이다.

보상 함수는 인자 및 계수를 변형해가며 실험한 결과 제일 좋은 결과를 보인 식 (10)과 같이 정하였다.

$$R_t = -10 * D + 10^4 * A + 0.5 * \sqrt{M} - 10^3 * S \quad (10)$$

식 (10)에서  $D$ 는 엔드 이펙터와 목표점 사이의 거리,  $A$ 는 목표점에 도착했을 시 1, 그렇지 않을 시 0이 되는 신호이다.  $M$ 은 매니플러빌리티이며  $S$ 는 경로상 존재하는 특이점이 되는 점과 엔드 이펙터의 거리가 5 cm 이하가 되면 1, 그렇지 않으면 0이 되는 신호이다. 즉, 특이점이 되는 점에서 멀어짐과 동시에 목표점과 가까워지는 짧은 경로일수록 보상이 커지게 된다.

#### 5.4 시뮬레이션 시스템 - 에이전트

에이전트는 환경으로부터 받은 데이터를 신경망에 입력하여 행동을 도출하며 전체적인 구조는 [Fig. 13]과 같다.

액터 신경망은 양팔 엔드 이펙터의 좌표(observation)를 입력 받아 행동인  $\Delta x, \Delta y, \Delta z, \Delta Y$ 를 출력하는 역할을 하고 [Fig. 14]와 같이 구성하였다. 활성화 함수로는 계층(Layer)1, 2에는 Relu 함수, output에는 tanh 함수를 사용하였다.

크리틱 신경망은 엔드 이펙터의 좌표와 행동을 입력받아 행동 가치함수를 계산하여 액터 신경망을 업데이트하는 역할을 한다. 그 구조는 [Fig 15]과 같다. 활성화 함수는 모든 계층에 Relu 함수를 사용하였다.

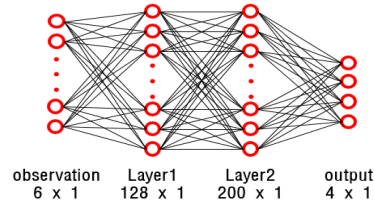
#### 5.5 학습 결과

학습의 결과는 [Fig. 16]과 같이 나왔으며 하나의 에피소드 당 100 step 움직이도록 설정하였고, 총 10,000 에피소드를 진행하며 학습하였다. 파란선은 하나의 에피소드에 대한 보상이

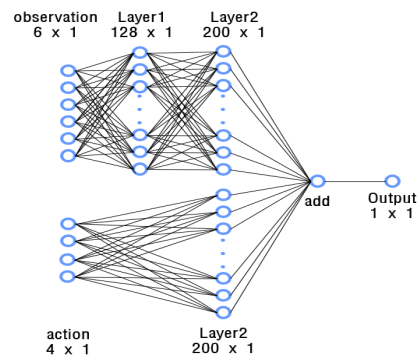
고 주황선은 에피소드들의 평균 보상 값이다. 특이점을 회피 하면서 목표점에 도달하면 양의 값을 보상으로 받도록 모델을 설정했기 때문에 학습 결과가 양의 값으로 수렴하면 좋은 모델이 된다.

약 900번째 에피소드에서 성공을 많이 하기 시작하였지만 실패하여 보상이 0점 아래로 떨어지는 경우도 생긴다. 4,000번 이후로 안정적으로 계속 성공하여 보상 값이 20,000에 가까운 에피소드들이 연속으로 나온다. 6,000번 근처에서 실패하는 경우가 3회 생겼지만, 이후로 안정적으로 수렴하였다.

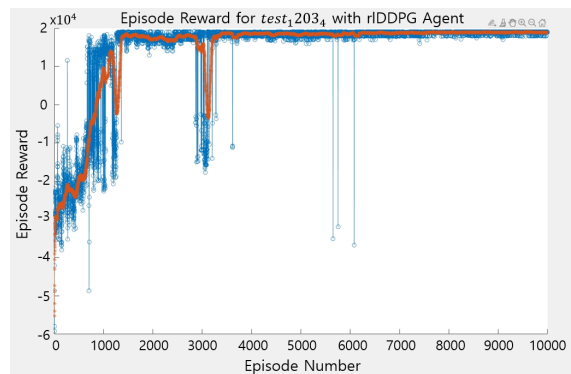
학습을 진행한 컴퓨터의 사양은 CPU i5-7600 3.50GHz, GPU NVIDIA GeForce GTX 1050, RAM DDR4 16GB이고 10,000번의 에피소드를 학습하는데 약 9시간이 소요되었다. 수렴 양상을 보이는 에피소드 4,000번대에 진입하는 데는 약 3시간 반이 소요되었다.



[Fig. 14] Actor network



[Fig. 15] Critic network



[Fig. 16] 10,000 episodes learning result

## 6. 시뮬레이션 및 실험

학습 결과로 저장된 에이전트를 시뮬레이션 대상으로 설정하고 시뮬레이션을 진행하였다. 그 결과는 [Fig. 17]에 나타내었다.

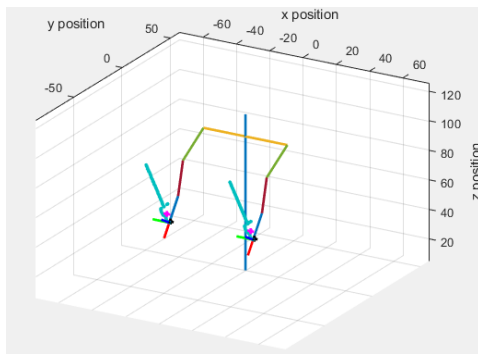
청록색은 에이전트가 회피 경로로 만들어낸 엔드 이펙터의 궤적을 나타낸 것이고, 양팔 매니퓰레이터가 분홍색 십자 모양 지점의 특이점을 회피하여 도착점에 도착한 것을 볼 수 있다. 특이점은 2.2절과 3절에서 언급했듯이 4번째 조인트의 각도가 0이 되며 3축과 5축의 회전축이 정렬되고 이 때 매니퓰러빌리티가 0이 되면서 발생한다.

해당 경로를 앞에서 본 [Fig. 18]을 보면 특이점 발생지점인 분홍 십자 위치를 x축의 음의 방향으로 회피하며 지나가는 모

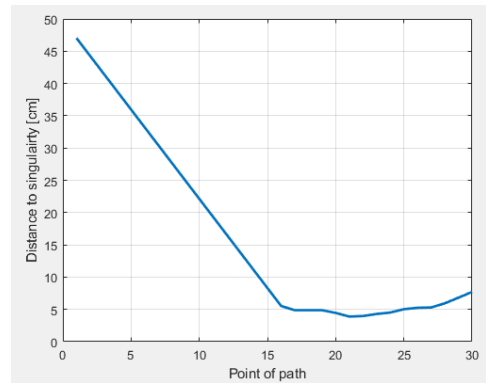
습을 볼 수 있다.

[Fig. 19]는 해당 경로를 위에서 본 모습이며 앞서 언급한 내용과 동일한 결과를 확인할 수 있다.

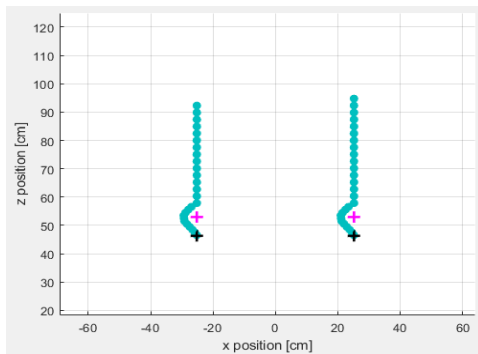
[Fig. 20]은 경로를 지나가면서 특이점까지 거리를 나타낸 그래프이다. X축은 경로를 이루는 점의 순번이기 때문에 무차원이다. X축이 16일 때까지 이어지는 직선은 단순 직선 경로 일 때의 거리이고 x축이 17일 때 이후부터 에이전트가 생성한 경로를 따르는 상황이다. 특이점과 가장 가까운 때의 거리가 3.9 cm인 것을 확인할 수 있는데, 이는 특이점 근접을 판단하는 기준인 5 cm보다 가까운 거리이다. 보상 식에 조인트의 속도가 엔드 이펙터의 속도로 전파되는 정도를 나타내는 매니퓰러빌리티를 추가하였기 때문에 회피 시 계속 5 cm를 유지하는



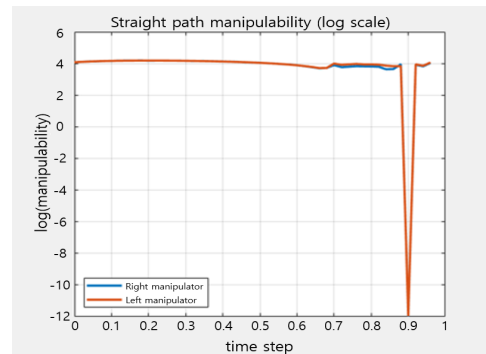
[Fig. 17] Singularity avoidance path



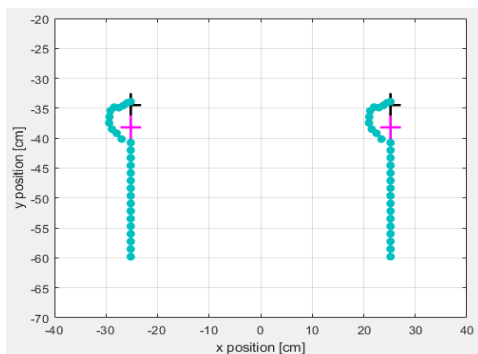
[Fig. 20] Distance to singularity



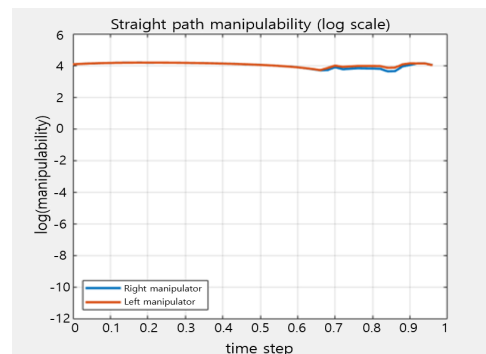
[Fig. 18] Singularity avoidance path - front view



[Fig. 21] Manipulability - straight path



[Fig. 19] Singularity avoidance path - top view



[Fig. 22] Manipulability - avoidance path

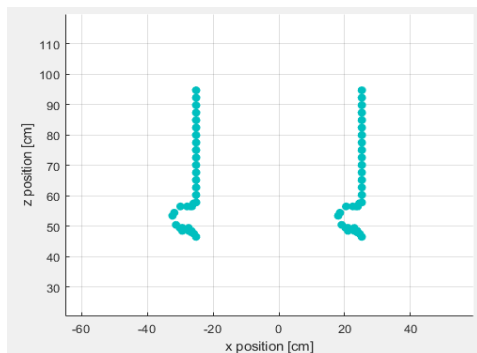
것보다 더 가까이 가면서 회피하는 것이 속도 전파성에서 더 좋다고 해석할 수 있다.

[Fig. 21]은 목표점까지 직선 경로와 회피 경로에 대한 매니퓰러빌리티의 로그 스케일 값을 보여준다. [Fig. 19]의 왼쪽의 직선경로에서는 y축인 매니퓰러빌리티가 0에 매우 근접한  $10^{-12}$ 까지 떨어진 것을 보아 특이점이 발생했음을 확인할 수 있다. 반면 [Fig. 22]처럼 회피 경로에서는 매니퓰러빌리티가 큰 값으로 안정적으로 유지되면서 특이점을 회피했음을 알 수 있다.

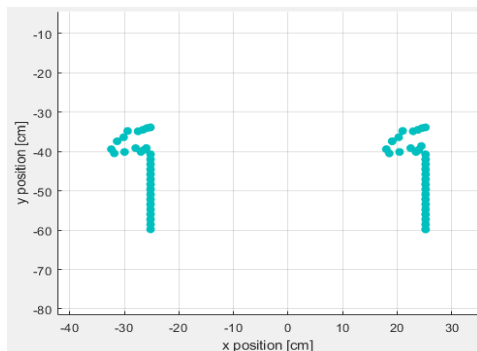
생성한 회피 경로로 실제 플랫폼에 적용하여 움직인 결과는 [Fig. 23]과 같다. Z자 모양으로 순서대로 본다.



[Fig. 23] Singularity avoidance path - experiment



[Fig. 24] End effector path - front view



[Fig. 25] End effector path - top view

[Fig. 24]와 [Fig. 25]는 [Fig. 23]의 실험에서 엔드 이펙터가 움직인 경로를 나타낸다. 시뮬레이션 경로인 [Fig. 18], [Fig. 19]와 비슷한 모양을 나타냄을 볼 수 있다.

## 7. 결 론

본 논문에서는 경로의 점마다 매니퓰러빌리티를 계산하고 이를 이용하여 직선 경로상에 존재하는 특이점을 찾아낸 후 이를 회피하는 경로 생성을 위해 DDPG 모델을 적용하였다.

DDPG 알고리즘으로 구성된 시뮬레이션 모델은 보상이 큰 목표점을 향해 엔드 이펙터가 나아가게 함과 동시에 특이점을 유발하는 경로 위의 점에 가까이 갈수록 음의 보상을 준다. 따라서 모델은 이 점을 장애물로 인식하고 회피하려는 동시에  $\dot{\theta}$ 의 속도 전파성을 나타내는 매니퓰러빌리티가 큰 경로를 생성하였다. 이는 자코비안 계산으로 특이점을 회피하는 기존 연구<sup>[7-9, 11]</sup>에 비해 속도 전파성까지 고려한 경로를 생성할 수 있다는 장점이 있다. 또한 기존의 강화학습을 이용한 장애물 회피 연구<sup>[17]</sup>는 엔드 이펙터의 방위 정보 없이 학습을 진행하였기 때문에 목표점에 도달하는 여러 자세가 결과로 도출된 반면, 본 연구에서는 방위 정보를 추가하여 안정된 자세로 목표점에 도달할 수 있었다.

학습의 결과 x축 음의 방향으로 반원 모양을 그리고 특이점 근접 판단 기준인 5 cm보다 낮은 3.9 cm를 보이는 등 기준보다 낮은 값을 보이면서 회피 경로를 생성하였다. 이는 특이점 근처라고 판단하여 음이 보상을 주는 기준인 5 cm에서 매니퓰러빌리티를 함께 고려한 결과라고 판단된다. 즉, 5 cm보다 가깝게 특이점을 회피하는 것이 속도 전파 측면에서 더 좋다는 의미로 해석된다.

그리고 본 시스템은 학습의 결과로 신경망의 가중치가 저장된 에이전트 파일을 저장하고 이를 경로 생성에 활용하는데, 새로운 목표점 혹은 회피점이 있을 시 다시 학습을 진행하지 않고 기존의 에이전트 파일을 활용하여 경로를 생성하도록 더 연구해야 할 부분이 있다.

더불어 학습 시 특이점을 점 장애물로 인식하고 학습을 진행하였는데, 점들을 이어 구역으로 회피 대상을 설정하면 공간 장애물에 대한 회피도 가능할 것으로 예상된다. 이는 비전 센서와 결합하여 실제 장애물에 대한 정보를 받으면 물체에 대한 회피 연구로 발전 가능할 것이다.

## References

- [1] S. Y. Chung, H. S. Kang, and M. J. Hwang, "Singularity Avoidance for Linear Motion of 6-DOF Manipulator," *Journal of Institute of Control Robotics and Systems*, vol. 24, no. 11, pp. 1025-1032, 2018, DOI: 10.5302/J.ICROS.2018.18.0164.



- [2] D.-J. Park, D.-E. Kim, J.-H. Park, and J.-M. Lee, "Singularity and Collision Avoidance Path Planning based upon Artificial Potential Field and Manipulability Measure," *Journal of the Institute of Electronics and Information Engineers*, vol. 55, no. 5, pp. 89-96, 2018, DOI: 10.5573/ieie.2018.55.5.89.
- [3] S. Kim and J. Park, "Singularity Avoidance Algorithms for Controlling Robot Manipulator: A Comparative Study," *The Journal of Korea Robotics Society*, vol. 12, no. 1, pp. 42-54, Jan., 2019, DOI: 10.7746/jkros.2019.12.1.042.
- [4] T. Yoshikawa, "Analysis and control of robot manipulators with redundancy. Robotic Research," *The First International Symposium*, New Hampshire, USA, pp. 735-747, 1984, [Online], <https://ci.nii.ac.jp/naid/80014489288>.
- [5] H. A. Park, M. A. Ali, and C. S. G. Lee, "Closed-Form Inverse Kinematic Position Solution for Humanoid Robots," *International Journal of Humanoid Robotics*, vol. 9, no. 3, 2012, DOI: 10.1142/S0219843612500223.
- [6] P. P. R. Filho, S. P. P. da Silva, V. N. Praxedes, J. Hemanth, V. Hugo C. de Albuquerque, "Control of singularity trajectory tracking for robotic manipulator by genetic algorithms," *Journal of Computational Science*, vol. 30, pp. 55-64, 2019, DOI: 10.1016/j.jocs.2018.11.006.
- [7] Y. Nakamura and H. Hanafusa, "Inverse kinematic solutions with singularity robustness for robot manipulator control," *ASME Journal of Dynamic Systems, Measurement and Control*, vol. 108, pp. 163-171, 1986, DOI: 10.1115/1.3143764.
- [8] C. W. Wampler, "Manipulator Inverse Kinematic Solutions Based on Vector Formulations and Damped Least-Squares Methods," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1 pp. 93-101, 1986, DOI: 10.1109/TSMC.1986.289285.
- [9] D. E. Whitney, "The mathematics of coordinated control of prosthetic arms and manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 94, no. 4, pp. 303-309, 1972, DOI: 10.1115/1.3426611.
- [10] W. Wang, M. Zhu, X. Wang, S. He, J. He, and Z. Xu, "An Improved Artificial Potential Field Method of Trajectory Planning and Obstacle Avoidance for Redundant Manipulators," *International Journal of Advanced Robotic Systems*, vol. 15, no. 5, pp. 1-13, 2018, DOI: 10.1177/1729881418799562.
- [11] S. R. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Journal of Robotics and Automation*, vol. 17, pp. 1-19, 2004, [Online], <http://graphics.cs.cmu.edu/nsp/course/15464-s17/lectures/iksurvey.pdf>.
- [12] Alex Owen-Hill, Robotiq Inc., *3 types of robot singularities and how to avoid them*, [Online], <https://robhub.org/3-types-of-robot-singularities-and-how-to-avoid-them>, Accessed: Nov. 29, 2020.
- [13] S. Phaniteja, P. Dewangan, P. Guhan, A. Sarkar, and K. M. Krishna, "A deep reinforcement learning approach for dynamically stable inverse kinematics of humanoid robots," *2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Macau, China, pp. 1818-1823, 2017, DOI: 10.1109/ROBIO.2017.8324682.
- [14] Y.-G. Kang and C.-S. Lee, "Deep Reinforcement Learning of Ball Throwing Robot's Policy Prediction," *The Journal of Korea Robotics Society*, vol. 15, no. 4 pp. 398-403, Dec., 2020, DOI: 10.7746/jkros.2020.15.4.398.
- [15] S. Kim, S. A. Kim, R. de Lima, and J. Choi, "Implementation of End-to-End Training of Deep Visuomotor Policies for Manipulation of a Robotic Arm of Baxter Research Robot," *The Journal of Korea Robotics Society*, vol. 14, no. 1 pp. 40-49, 2019, DOI: 10.7746/jkros.2019.14.1.040.
- [16] T. Kim, Y. Park, J. B. Kim, Y. Park, and I. H. Suh, "Development of an Actor-Critic Deep Reinforcement Learning Platform for Robotic Grasping in Real World," *The Journal of Korea Robotics Society*, vol. 15, no. 2, pp. 197-204, Jun., 2020, DOI: 10.7746/jkros.2020.15.2.197.
- [17] M. Duguleana, F. G. Barbuceanu, A. Teirelbar, and G. Mogan, "Obstacle avoidance of redundant manipulators using neural networks based reinforcement learning," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 132-146, 2012, DOI: 10.1016/j.rcim.2011.07.004.
- [18] D. L. Pieper, "The kinematics of manipulators under computer control," Ph.D. dissertation, Stanford University, USA, 1968, [Online], <https://www.saildart.org/AIM/072/pdf>.
- [19] L. Rozo, N. Jaquier, S. Calinon and D. G. Caldwell, "Learning manipulability ellipsoids for task compatibility in robot manipulation," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, pp. 3183-3189, 2017, DOI: 10.1109/IROS.2017.8206150.



**이 종 학**

2018 부산대학교 환경공학전공(학사)  
2019~2021 부산대학교 전기전자공학과  
(석사)

관심분야: 매니플레이터, 제어 시스템, 휴머노이드



**김 윤 재**

2014~2021 부산대학교 전자공학과(학사)

관심분야: 회로 설계, 제어 시스템



**김 경 수**

2014~2021 부산대학교 전자공학과(학사)

관심분야: 회로 설계, 제어 시스템



**이 장 명**

1980 서울대학교 전자공학과(학사)  
1982 서울대학교 전자공학과(석사)  
1990 미국 남가주대학교 전자공학과(박사)  
1992~현재 부산대학교 전자공학과 교수

관심분야: 특수환경 Navigation and Localization, 지능로봇시스템 설계  
및 제어, 마이크로프로세서 응용