



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공 학 석 사 학 위 논 문

기지 환경 하의 전방향구동용 무인  
지상차량을 위한 다중목표점 경로  
계획 및 추적제어기 설계



2019 년 2 월

부 경 대 학 교 대 학 원

기계설계공학과

김 창 규

공 학 석 사 학 위 논 문

기지 환경 하의 전방향구동용 무인지상차량을 위한  
다중목표점 경로계획 및 추적제어기 설계

지도교수 김 상 봉

이 논문을 공학석사 학위논문으로 제출함.



2019 년 2 월

부 경 대 학 교 대 학 원

기계설계공학과

김 창 규

김창규의 공학석사 학위논문을  
인준함.

2019년 2월



주 심 공학박사 최연욱 (인)

위 원 공학박사 서진호 (인)

위 원 공학박사 김상봉 (인)

# 목 차

목 차	i
<b>Abstract</b>	iii
<b>제 1 장 서론</b>	<b>1</b>
1.1 연구 배경 및 동기	1
1.2 연구 목적 및 방법	6
1.3 연구 내용 및 범위	8
<b>제 2 장 시스템 구성</b>	<b>10</b>
2.1 기구 시스템의 구성	11
2.2 제어 시스템의 구성	14
2.2.1 센서부	16
2.2.2 제어부	22
2.2.3 구동부	24
2.2.4 전원공급장치	26
<b>제 3 장 다중 목표점 경로계획 알고리즘</b>	<b>28</b>
3.1 D* Lite 알고리즘	28
3.2 다중 목표점 경로계획 알고리즘	31
3.3 전체 목표점 경유 경로 생성	39
<b>제 4 장 시스템 모델링 및 제어기 설계</b>	<b>43</b>
4.1 OAGV의 기하학적 모델링	43
4.2 OAGV의 수학적 모델링	46

4.2.1	기구학적 모델링	46
4.2.2	동역학적 모델링	50
4.3	OAGV 제어기 설계	56
<b>제 5 장</b>	<b>시뮬레이션 및 실험 결과</b>	<b>64</b>
5.1	시뮬레이션 결과	65
5.2	실험 결과	81
<b>제 6 장</b>	<b>결론 및 향후 연구</b>	<b>90</b>
6.1	결론	90
6.2	향후 연구	95
<b>감사의 글</b>		<b>96</b>
<b>참고 문헌</b>		<b>99</b>
<b>학술논문 발표 및 게재</b>		<b>105</b>
<b>부 록</b>		<b>107</b>
부록 A		107
부록 B		110
부록 C		114

# **Multiple Target Point Path Planning and Its Tracking Controller Design for an Omnidirectional Automatic Guided Vehicle in Known Environment**

Chang Kyu Kim

*Dept. of Mechanical Design Engineering*

*The Graduate School, Pukyong National University, Republic of Korea*

## **Abstract**

Today, Automatic Guided Vehicles (AGVs) with a path planning are used in many industrial fields. Mainly, Automatic Guided Vehicle (AGV) is a convenient mobile robot that cleans a room or transports industrial material in a known environment. In order to be used more safely and widely, AGV needs to be able to plan a path for itself and furthermore provide a tracking controller. However, the existing path planning algorithms have the disadvantage of traversing between obstacles or passing over the sharp edges of obstacles. In order to solve these weaknesses, a new path algorithm to set multiple target points in the

existing path planning and a controller design to track the planned paths are needed deeply.

This thesis proposes multiple target point path planning for an Omnidirectional Automatic Guided Vehicle (OAGV) and designs a tracking controller to track paths generated by the proposed path planning. The path planning algorithms include various algorithms such as A\*, D\*, D\* Lite. These three path planning algorithms are focused on moving from the starting point to the target point in the shortest path avoiding obstacles in a known environment. Multiple target points are created with a modified D\* Lite algorithm based on the most efficient D\* Lite algorithm among these three algorithms. To do this task, the followings are done. Firstly, it is assumed that the workspace is divided into cells and the size of one cell is the size of OAGV. Secondly, a method of setting multiple target points using the Hamiltonian path is presented. Thirdly, the configuration of the OAGV used in the experiment is described. The OAGV system consists of several interconnected devices such as main controller, monitor, keyboard, mouse, DC servo driver, and industrial computer, the omnidirectional drive wheel, and so on. The laser sensor NAV-200 is used to detect the OAGV posture in known environment. Fourthly, kinematic modeling and dynamic modeling of the OAGV are presented. In particular, a tracking controller that tracks paths obtained by the multiple target point path planning algorithm is presented. A tracking controller is designed using the backstepping method, and the stability of the designed tracking controller is proven by the Lyapunov stability theory. Finally, simulation shows the comparison of the multiple target point path planning and D\* Lite

algorithm. Simulation and Experimental results are shown to verify the effectiveness and the performance of the proposed path planning algorithm and the proposed controller.

**Keywords:** Path planning, OAGV, Multiple target points, Backstepping tracking controller, Three wheels



# 제 1 장 서 론

## 1.1 연구 배경 및 동기

자동화라는 것은 시간을 절약시킬 수 있고, 불필요한 인력의 소모를 줄일 수 있음에 최근 산업용 로봇들이 많이 연구되고 있다. 특히, AGV(Automatic Guided Vehicle)가 대표적인 산업 자동화 로봇으로 병원에서 의료기기의 운반[1], 도서관이나 서점에서 책의 운반, 반도체공장에서 물류 조달 등 많은 산업현장에 투입되어 작업의 효율성을 높여주고 편리함을 준다[2]. 또한 최근에는 이러한 분야에 국한되지 않고 방사능 오염으로 사람이 접근하기 힘든 환경이나 위험한 작업현장에 사람을 대신하여 AGV가 투입되고 있다[3]. 이처럼 AGV의 자동화를 수행하기 위해서는 주어진 경로에 대한 경로계획이 필요하며, AGV의 공장 자동화가 상용화됨에 따라 많은 경로계획과 제어기법들이 연구되고 있다. Fig. 1.1은 서점에서 사용되고 있는 NAVERLABS M1[17]이고, Fig. 1.2는 SK 하이닉스의 산업물류이송 AGV[18]이다.



Fig. 1.1 NAVERLABS M1



Fig. 1.2 SK Hynix Mask AGV

AGV의 경로계획에 있어서 크게 점대점 방식(Point-to-Point)[4]과 전체 커버리지 경로계획방식(Complete Coverage Path Planning)[5,6]으로 분류된다. 점대점 방식이란 시작점에서 목표점까지 최적의 경로를 이루면서 이동하는 방법을 말하며, 전체 커버리지 경로계획방식은 청소로봇과 같이 이동로봇의 작용기(Effector)가 주어진 영역의 자유영역(Free space)을 모두 지나가는 경로를 생성하는 전체 영역을 이동하여야 하는 경우에 적용되는 방법을 말한다.

점대점 방식 경로계획에서는 셀 분할을 이용한 방식에 기초한 경로계획 알고리즘들이 존재한다. 예를 들면, LPA\*(Lifelong Planning A\*)[7], A\*[8], D\*[9], D\* lite[10,11] 경로계획 알고리즘 등이 있다. 그러나, 이러한 알고리즘들의 단점이 있다. 가령, 장애물의 날카로운 모서리를 지날 때, 최단경로로 지나기 위해서 AGV가 장애물에 손상을 입을 수 있다. 또한 최단경로에 있어서 AGV가 지나기에 충분하지 않은 너비의 두 장애물 사이를 통과할 때, AGV가 장애물 사이를 통과하려다 날카로운 모서리에 손상을 입을 수 있다. D. Mackay는 점대점 경로계획방식에서 가장 효과적이고 발전된 경로계획방식인 D\* Lite 경로계획 알고리즘을 제안했다[10]. D\* Lite 경로계획 알고리즘은 시작점에서 목표점까지 최단경로로 이동하는데 최적화되어있다. 그러나, D\* Lite 경로계획 알고리즘은 최단경로로 이동하기위해 장애물의 날카로운 모서리를 지나거나 두 장애물 사이의 작은 틈을 횡단하려고 하는 단점이 있다. 즉, D\* Lite 경로계획 알고리즘은 다음과 같은 단점들이 있다.

첫째, 장애물의 날카로운 모서리를 지난다.

둘째, 두 장애물 사이를 횡단한다.

전체 커버리지 경로계획에서는 모스 셀 분할법(Morse cell decomposition)을 이용하여 주어진 이진(Binary)영역을 셀로 분할하는 방법이 사용된다. 김진욱은 주어진 영역에 대한 전체 커버리지 경로계획을 바탕으로 생성된 커버리지 경로에 대하여 이륜 이동로봇의 경로 추적을 제안했다[22]. 셀 간의 커버리지 순서결정을 위해 너비우선 탐색 알고리즘 및 깊이우선 탐색 알고리즘을 이용하여 전체 영역에 대해 최적의 경로를 생성할 수 있지

만 로봇청소기용이 아닌 기지환경의 물류공장등에 사용하기에 너무 많은 계산시간이 필요하다는 단점이 있다. E. U. Acar 등은 주어진 영역을 여러 슬라이스 함수(Slice fuction)를 이용하여 영역을 분할하고, 각 슬라이스 함수에 의해 분할된 여러 모양을 가지는 셀에 대해 커버리지 경로를 계획하였다[23]. 슬라이스 함수를 통해 셀을 분할하는 방법은 경우에 따라 적절한 방법일 수 있으나, 실제 현장에는 실용적이지 못하며 셀을 분할하는 방법이 복잡하다는 단점이 있다. 또한 P. P. Sandi는 다중신장트리에 기반하여 에너지소비가 최소가 되는 AGV의 경로계획을 제시하였다[14]. 하지만 에너지소비의 최소경로로 이동함에 있어 장애물의 날카로운 모서리에 AGV가 손상될 위험이 존재한다. 따라서 전체 커버리지 경로계획방식은 정밀하고 신속한 AGV의 움직임이 요구되는 기지환경의 물류공장 등에 사용되기에 적절하지 못하다.

또한 B. T. Luan은 레이저 센서를 사용한 PD 퍼지로직에 기반한 OAGV의 궤도를 추적하는 제어기를 제시하였다[13]. 그러나 경로계획을 통해 장애물 회피에 초점을 맞추기보다 주어진 경로를 잘 추적하는지에 대해서만 다루었다. S. H. Park 등은 원격조작이 가능한 이동로봇에 센서를 이용하여 확대된 안내원을 이용한 장애물 회피 알고리즘을 제안했다[15]. 이동로봇의 중심으로부터 확대된 안내원을 통해 장애물의 모서리를 지나지 않지만, 최단 경로를 이루면서 이동로봇이 목표점으로 가는 것은 다루지 않았다.

한편, 제어기법으로는 앞서 OAGV(Omnidirectional Automatic Guided Vehicle)를 연구한 T. D. Viet 등은 분산된 제어방식을 통해 외란과 마찰을 가진 전방향 이동 매니플레이터

시스템(OMMS)에 적용하였다[12]. 분산된 제어방식은 AGV 제어 장치의 고장발생시 매니플레이터시스템에 미치는 영향이 적다는 장점은 있지만, 제어장치의 수가 많아 복잡하다는 단점이 있다.

박유미는 퍼지 제어를 이용한 전방향 주행로봇 OMR(Omni-directional Mobile Robot)의 곡선경로 추적을 제안했다[24]. 퍼지 제어는 시스템을 효과적으로 수행할 수 있는 적절한 퍼지규칙(Fuzzy rule)을 설정하는 것이다. 그러나, 제어규칙을 설정하기 위해서는 전문가의 경험이 많이 요구된다는 단점이 있다.

김근우 등은 비선형 환경인 실외지역에서 GPS/IMU센서를 통한 이동로봇의 효율적인 위치추정을 위해 확장된 칼만필터를 이용하여 효과적인 실외 위치 추정 알고리즘을 제안했다[16]. 확장된 칼만필터는 비선형시스템에서 상태추정이 가능하고, 계산량이 적다는 장점이 있지만, GPS성능에 영향을 많이 받고 이를 보완하기 위해 사용된 IMU센서 자체의 특성오차와 외부환경에 의한 외란으로 장시간 사용시 오차가 누적되는 단점이 있다.

따라서 D\* Lite 경로계획 알고리즘을 수정하여 AGV가 손상되지 않고 안전한 주행을 할 수 있게 해주는 다중목표점 경로계획 알고리즘과 마찰과 외란을 가지는 비선형적 환경에서 AGV를 제어할 수 있는 제어기의 개발이 필요하다.

## 1.2 연구 목적 및 방법

본 연구에서는 경로계획 시 AGV가 장애물의 날카로운 모서리에 의해 손상을 입지 않고 안전한 주행을 할 수 있도록 점대점 경로계획방식인 D\* Lite 경로계획 알고리즘의 단점을 보완하여 다중목표점 경로계획 알고리즘을 제시한다. IMU센서의 오차누적의 단점을 보완하기 위해 기지환경 하의 AGV 자기위치 인식률이 좋은 NAV-200센서를 사용하며, 제어장치의 수가 많아 복잡한 분산된 제어방식이나 어려운 제어규칙을 설정하여야 하는 퍼지제어의 단점을 피하고 AGV의 동역학적 모델링을 통한 백스텝핑법으로 제어를 설계한다.

따라서, 본 연구의 목적은 OAGV(Omnidirectional Automatic Guided Vehicle)가 기지 환경 하에서 장애물을 회피하면서 시작점으로부터 다중목표점들에 도착하도록 하기 위한 경로를 생성하게 하는 다중목표점 경로계획 알고리즘을 제시하는 것이다. 또한 제안된 경로계획 알고리즘에 의해 생성된 경로를 추적하는 백스텝핑법을 이용한 OAGV시스템의 경로추적 제어를 제시하는 것이다. 본 연구의 목적을 달성하기 위해서 다음과 같이 수행한다.

첫째, 본 연구에 사용될 OAGV시스템의 구성 및 그 제원에 대해 서술한다. 둘째, D\* Lite 경로계획 알고리즘을 토대로 제안된 다중목표점 경로계획 알고리즘을 제시한다. 셋째, 제안된 다중목표점 경로계획 알고리즘을 통해 주어진 환경의 다중목표점을 가지는 경로를 생성한다. 넷째, 본 연구에 사용된 OAGV의 기구학적, 동역학적 모델링을 제시한다. 다섯째, 제시된 다중목표점 경

로계획 알고리즘에 의해 생성된 다중목표점을 가지는 경로를 추적하기 위하여 OAGV의 모델링에 기반한 비선형 백스텝핑법을 이용한 OAGV의 경로추적제어기를 설계한다. 마지막으로 제안된 OAGV의 제어기의 유효성을 보여주기 위해 시뮬레이션과 실험결과를 제시한다.



## 1.3 연구 내용 및 범위

본 연구에서는 기지의 환경에서 다중목표점 경로계획 알고리즘을 제시하고, 이 알고리즘으로 생성된 다중목표점을 가진 경로를 추적하는 백스텝핑법을 이용한 OAGV의 경로추적제어기를 제시한다. 이를 위해 본 연구는 다음과 같이 총 6장으로 구성되어 있으며, 각 장의 내용은 다음과 같이 요약된다.

**제 1 장 :** 본 연구의 연구 배경과 국, 내외 연구 동향을 살펴 그 문제점을 제시한다. 또한, 본 연구의 목적 및 방법에 대한 간략한 소개와 연구 내용 및 범위를 제시한다.

**제 2 장 :** 본 연구를 수행하기 위해 사용될 OAGV의 기구, 제어 시스템을 구성한다. 제어시스템으로는 산업용컴퓨터 ARK-5260을 사용하고, 레이저센서 NAV-200을 사용하여 OAGV의 자기위치인식을 정확하게 한다. 바퀴로는 더블 유니버설 전방향 구동바퀴를 사용하여 전방향구동이 가능하게 한다.

**제 3 장 :** OAGV가 기지의 환경에서 시작점에서 다중목표점 들까지 장애물을 회피하며 최소경로로 이동하는

D\* Lite 경로계획 알고리즘을 제시하고 Hamiltonian path이론을 이용하여 목표경로를 생성시키는 다중목표점 경로계획 알고리즘을 제안한다.

제 4 장 : OAGV의 동역학적 모델링을 제시하고 이를 바탕으로 백스텝핑 기법을 사용하여 다중목표점 경로계획 알고리즘에 의해 생성된 목표경로를 추적하기 위한 백스텝핑법을 사용한 비선형 경로추적제어기를 설계한다.

제 5 장 : OAGV를 사용하여 제안된 다중목표점 경로계획 알고리즘과 제어기의 유효성을 검증하기 위해 시뮬레이션 및 실험결과를 제시한다.

제 6 장 : 본 연구의 결론 및 향후 연구에 대하여 간략히 기술한다.

## 제 2 장 시스템 구성

본 장에서는 본 연구에 사용된 실제 전방향구동이 가능한 무인운반차(Omnidirectional Automatic Guided Vehicle, OAGV)의 시스템 구성 및 그 제원에 대해 서술한다. OAGV는 기구 시스템과 제어시스템으로 구성되어있으며, 제어시스템은 센서부, 제어부, 조작부 및 전원 공급장치로 구성된다.

Fig. 2.1은 OAGV의 전체적인 시스템을 나타낸다.



Fig. 2.1 OAGV system used for the thesis

## 2.1 기구시스템의 구성

Fig. 2.2는 본 연구에서 사용된 OAGV시스템의 기구적인 구성도를 나타낸다. 바닥면에서 OAGV는 원통형 구조이며,  $\phi 0.55\text{m} \times 0.53\text{m}(H)$ 로 3축 전방향 구동바퀴 시스템을 사용한다. OAGV의 총 무게는 45kg이다. 3개의 바퀴는 서로 독립 구동이며,  $120^\circ$ 의 각도를 이룬다. 각 바퀴는 기어비 12:1이고, 엔코더 360 P/R(Pulse/Revolution)를 갖춘 DC서보모터 (48.6W/ 24V/ 4000RPM)로 구동된다. OAGV의 중심에 위치하고 있는 NAV-200 레이저센서는 전역좌표계에서 OAGV의 위치를 측정해준다. 산업용컴퓨터(ARK-5260)는 OAGV의 제어를 위해 OAGV의 내부에 위치하고있다. 후면부에는 제어기변수들을 조정하게 해주고 나타내주는 7인치 모니터가 장착되어 있다. 외부에는 Fig. 2.1과 같이 OAGV의 전원을 공급해 주는 1개의 리튬이온배터리가 있다.



Fig. 2.2 Structure of the OAGV used for experiment

Fig. 2.3은 본 연구에 사용된  $\phi 0.125\text{m}$  인 전방향 구동바퀴이다. 전방향 구동바퀴는 별도의 조향장치 없이도 이동하거나 회전할 수 있다. 그 이유는 바퀴의 회전에 의하여 자체적인 구동력을 전달하는 능동모드(Active mode)와 자체적으로 구동력을 전달하지 않고 자유롭게 회전할 수 있는 수동모드(Passive mode)를 가짐으로써 평면상의 3자유도를 가지기 때문이다. 즉 다시말해, 평면상에서 능동구름(Active rolling) 방향의 자유도와 수동구름(Passive rolling) 방향의 자유도, 접촉면의 미끄러짐

(Slip)의 3자유도를 갖는다. 특히, 본 연구에 사용된 OAGV의 3개의 바퀴는 서로 독립 구동이기 때문에 전방향 이동이 가능하다. 게다가 주행방향에 직각으로 자유롭게 구르도록 8개의 롤러가 테두리인 림에 정렬되어 있으며, 하나의 유니버설바퀴가 아닌 수직 진동발생을 최소화 할 수 있는 더블 유니버설 전방향 구동바퀴를 선정하였다.

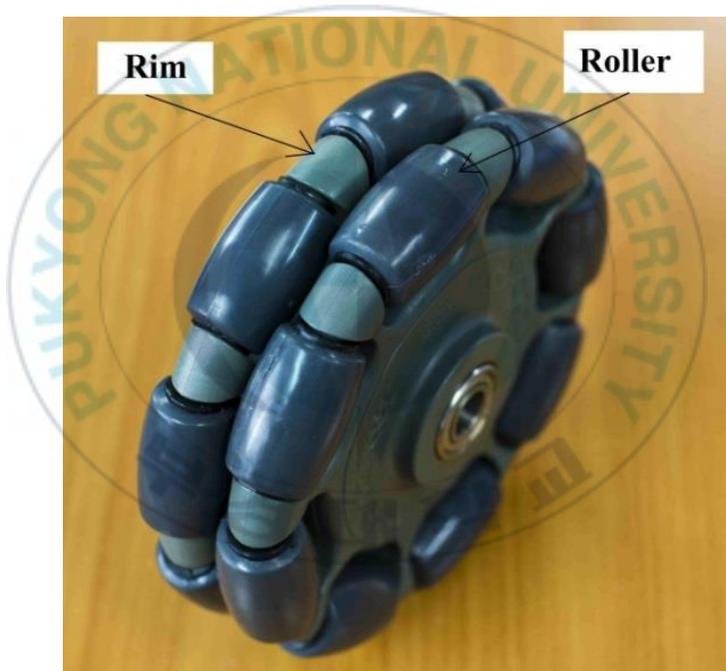
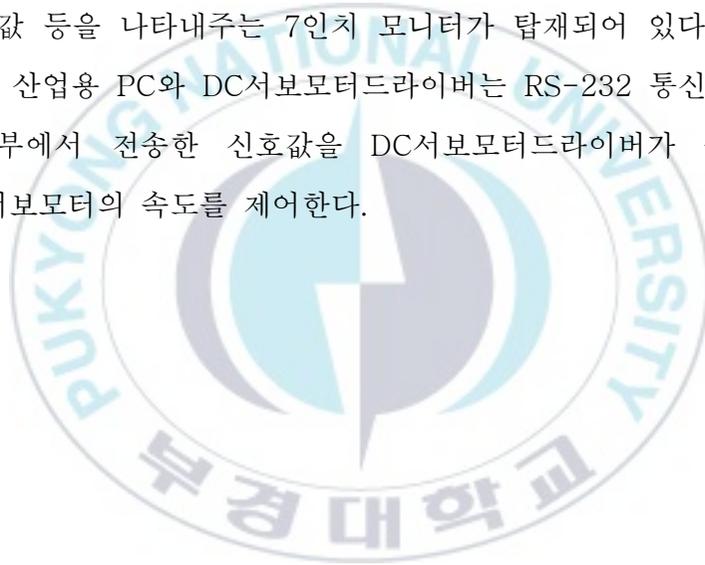


Fig. 2.3 Structure of the omnidirectional wheel

## 2.2 제어시스템의 구성

Fig. 2.4는 OAGV의 제어시스템의 구성을 나타낸다. 이 제어시스템은 크게 센서부, 제어부, 구동부로 나눌 수 있다. 센서부는 레이저센서인 NAV-200을 사용한다. NAV-200은 산업용 PC와 RS-232 통신으로 연결되며, 전송속도는 19,200Hz이다. 제어부는 산업용 PC인 ARK-5260제품을 사용하였고, 계산값 등을 나타내주는 7인치 모니터가 탑재되어 있다. 구동부에서 산업용 PC와 DC서보모터드라이버는 RS-232 통신을 하며, 제어부에서 전송한 신호값을 DC서보모터드라이버가 수신하여 DC서보모터의 속도를 제어한다.



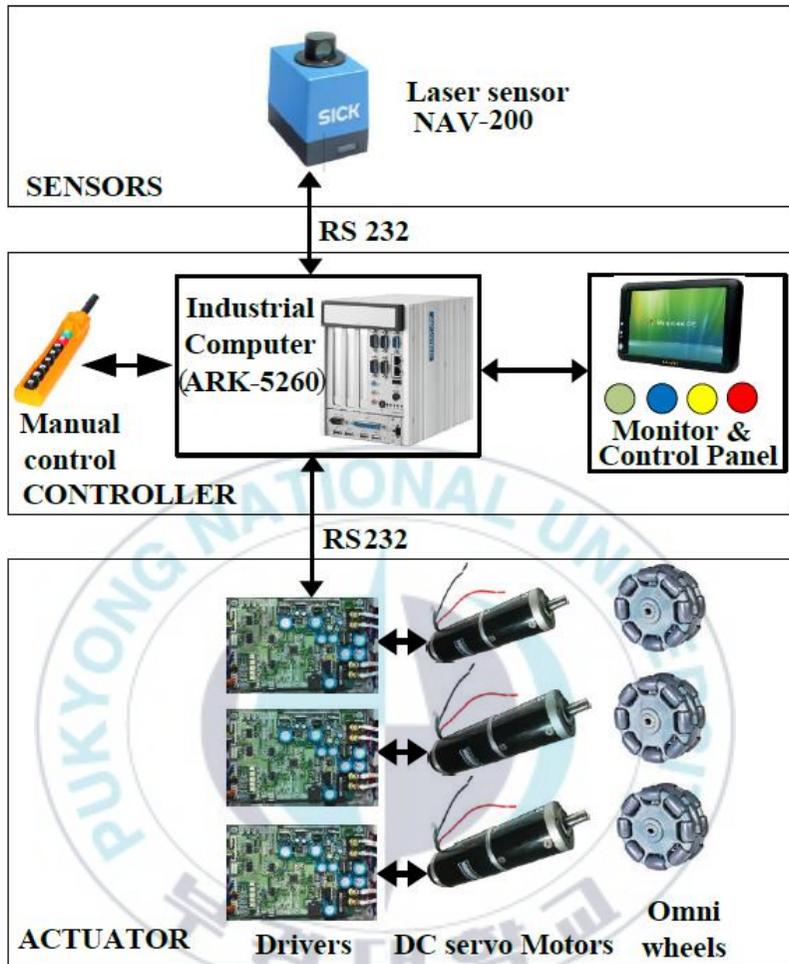


Fig. 2.4 Structure of the controller system of OAGV

## 2.2.1 센서부

본 연구에서는 레이저센서 NAV-200을 사용하였다.

Fig. 2.5는 레이저센서 NAV-200의 구성을 나타내며, Fig. 2.6은 NAV-200과 반사판(Reflector)을 사용한 OAGV의 위치 측정 개념도이다. OAGV에 장착되어 있을 때 NAV-200은 360°로 회전하며 고정된 반사판으로부터 2차원의 좌표를 연속적으로 측정 및 계산한다. OAGV의 위치인식을 위해 최소 3개이상의 반사판이 필요하며, 4~5개의 반사판 설치시 더욱 정확한 위치를 측정할 수 있다. 여기에서 측정된 정보들은 모두 산업용 PC로 전송된다.



Fig. 2.5 Laser navigation sensor NAV-200

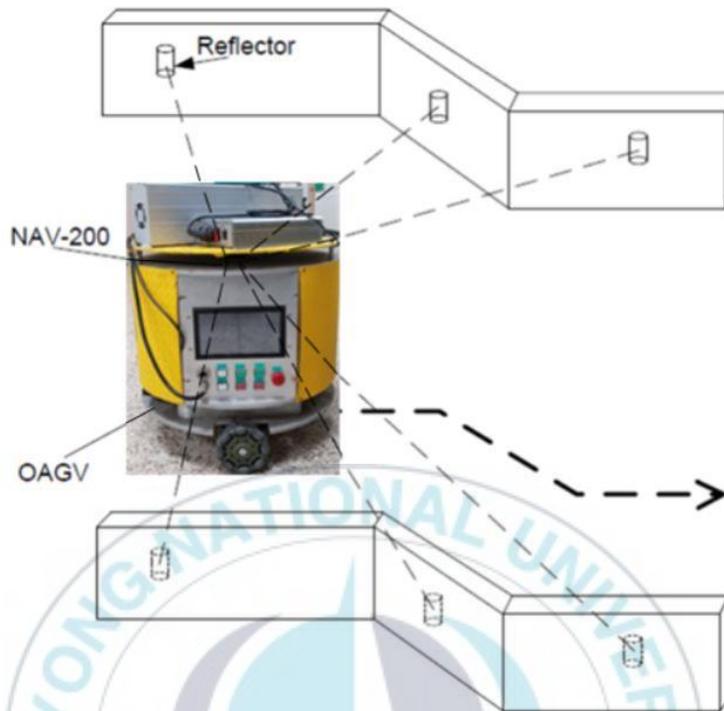


Fig. 2.6 Concept for position measurement by the NAV-200 and reflectors

Table 2.1은 레이저센서 NAV-200의 세부제원이다.

Table 2.1 Specification of laser sensor NAV-200

No	Parameter	Value
1	Light source	Infrared (855nm)
2	Laser class	1
3	Field of view	360°
4	Scanning frequency	8Hz

5	Angular resolution	$\pm 0.1^\circ$
6	Operating range	1.2m - 28.5m
7	Max. range 10% reflectivity	28.5,
8	Data communication	RS-232
9	Data transmission rate	19,200Hz
10	Operating voltage	$\geq 24V DC \pm 25\%$
11	Weight	3.3Kg
12	Dimensions (H×L×W)	176×178×115mm

Fig. 2.7은 레이저센서 NAV-200의 위치측정원리를 나타낸 것이다. 먼저,  $R_n(X_n, Y_n)$ 은 레이저센서 NAV-200이 인식가능한 반사판을 나타내며, 이를 기반으로 OAGV의 위치 및 방향을 계산한다.  $\theta$ 는 전역 좌표계(global coordinate frame)에의 X 축과 지역 이동좌표계(local coordinate frame)의 X 축사이의 각도를 나타내며,  $\theta_n$ 은 지역좌표계의 X 축과 각 반사판에서 NAV-200을 잇는 직선의 각도를 나타낸다.  $d_n$ 은 각 반사판과 NAV-200을 잇는 직선거리이며, 레이저빔이 NAV-200에 의해 송신되어 반사판에 의해 반사되어 다시 NAV-200에 수신될 때까지의 시간 및 레이저빔의 속도에 의해 구해진다.

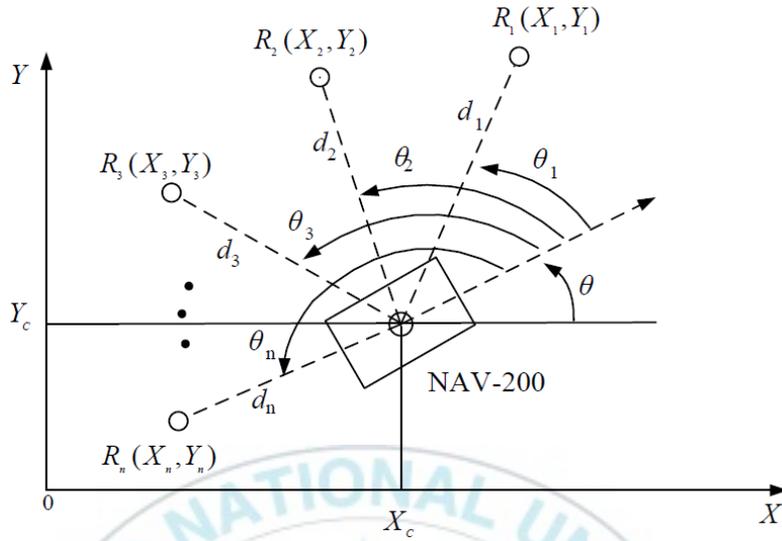


Fig. 2.7 Principle for position measurement of the NAV-200

OAGV의 중심에 설치되어 있는 레이저센서 NAV-200의 중심위치  $(X_c, Y_c)$ 와 방향각  $\theta$ 는 다음과 같이 구해진다[25].

$$X_c = \left[ \frac{1}{(l_2 - l_1)(1 + l^2)} \right] \left\{ (l_2 l - 1)[(l_1 + l)X_1 + (l_1 l - 1)Y_1] - (l_1 l - 1)[(l_2 + l)X_2 + (l_2 l - 1)Y_2] \right\} \quad (2.1)$$

$$Y_c = \left[ \frac{1}{(l_2 - l_1)(1 + l^2)} \right] \left\{ (l_1 + l)[(l_2 + l)X_2 + (l_2 l - 1)Y_2] - (l_2 + l)[(l_1 + l)X_1 + (l_1 l - 1)Y_1] \right\} \quad (2.2)$$

여기서,  $l_1 = \tan \theta_1$ ,  $l_2 = \tan \theta_2$ ,  $l_3 = \tan \theta_3$  과  $l = \tan \theta$  이며,  $l$  은 다음과 같이 주어진다.

$$l = \frac{(l_3 - l_1)(Y_1 - Y_2 - l_1 X_1 + l_2 X_2) - (l_2 - l_1)(Y_1 - Y_3 - l_1 X_1 + l_3 X_3)}{(l_3 - l_1)(l_1 Y_1 + X_1 - l_2 Y_2 - X_2) - (l_2 - l_1)(l_1 Y_1 + X_1 - l_3 Y_3 - X_3)} \quad (2.3)$$

$$\theta = A \tan 2(Y_1 - Y_C, X_1 - X_C) - \theta_1 \quad (2.4)$$

여기서,  $A \tan 2(Y_1 - Y_C, X_1 - X_C)$  는 2개의 인수로 된 arctangent 함수이며, 다음과 같은 조건 하에 정의된다.

$$A \tan 2(Y_1 - Y_C, X_1 - X_C) = \begin{cases} 0, & X_1 - X_C > 0, Y_1 - Y_C = 0 \\ \frac{\pi}{2}, & X_1 - X_C = 0, Y_1 - Y_C > 0 \\ \frac{3\pi}{2}, & X_1 - X_C = 0, Y_1 - Y_C < 0 \\ \pi, & X_1 - X_C < 0, Y_1 - Y_C = 0 \\ \tan^{-1}\left(\frac{Y_1 - Y_C}{X_1 - X_C}\right), & X_1 - X_C > 0, Y_1 - Y_C > 0 \\ \tan^{-1}\left(\frac{Y_1 - Y_C}{X_1 - X_C}\right) + \pi, & X_1 - X_C < 0, Y_1 - Y_C > 0 \\ \tan^{-1}\left(\frac{Y_1 - Y_C}{X_1 - X_C}\right) + \pi, & X_1 - X_C < 0, Y_1 - Y_C < 0 \\ \tan^{-1}\left(\frac{Y_1 - Y_C}{X_1 - X_C}\right) + 2\pi, & X_1 - X_C > 0, Y_1 - Y_C < 0 \end{cases}$$

## 2.2.2 제어부

본 연구에서 제어시스템으로는 산업용 PC인 ARK-5260을 사용한다. 산업용 PC ARK-5260의 구조가 Fig. 2.8에 도시된다.

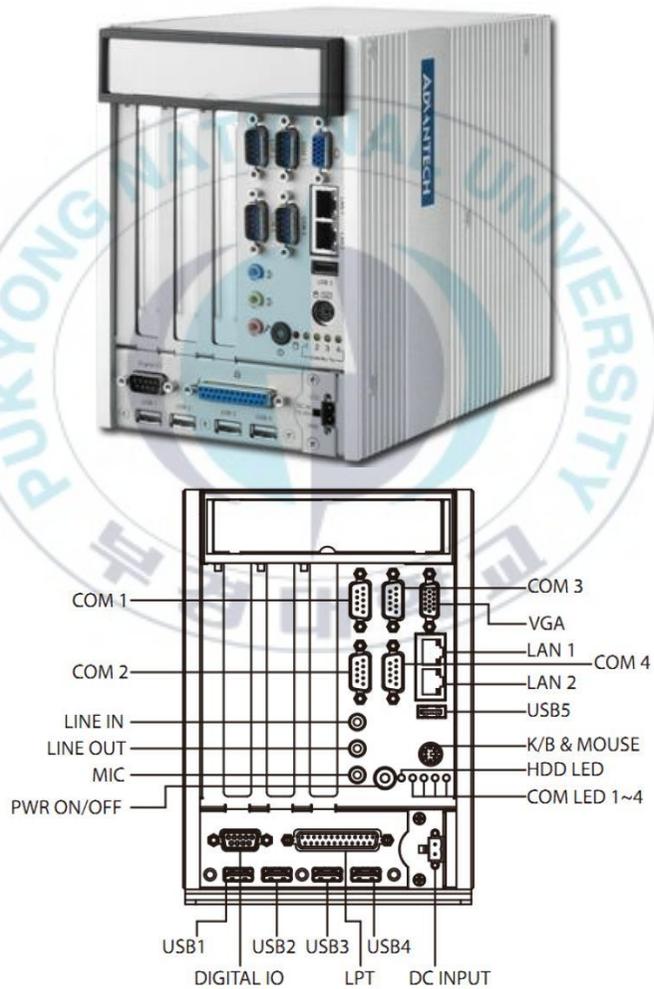


Fig. 2.8 Industrial PC ARK-5260

산업용 PC ARK-5260의 세부제원은 다음 Table 2.2와 같다.

**Table 2.2 Specification of Industrial PC ARK-5260**

<b>System</b>	
CPU	Intel Atom™ D510
Max. speed	1.66 GHz
Chipset	Intel ICH8M
Memory	2 GB DDR2 667MHz
Ethernet controller	Dual Realtek RTL8111E PCIe GbE controllers with ASF 2.0 support
<b>I/O and indicators</b>	
Ethernet	10/100/1000 Mbps 2Giga LAN controller Intel 82567V
RS-232	4xRS232/422/485 port
USB	5xUSB 2.0
VGA	1xVGA port(2048x1536)
Audio connector	1xLine-out port 1xMic-in port
<b>Storage</b>	
SATA	SATA HDD/SSD bay
<b>Power</b>	

Power range	12~24 VDC input
<b>Mechanical</b>	
Dimensions (W×H×L)	137x189x221 mm
Weight	4.2 Kg

### 2.2.3 구동부

본 연구에서 사용된 구동부는 모터드라이버와 DC서보모터로 구성되어 있다. 모터드라이버는 마이크로컨트롤러를 사용한 ARM COTEX-M3 STM32F103이며, Fig. 2.9에 나타난다. 모터드라이버의 최대허용전류는 10A이다. 산업용 PC와 모터드라이버는 RS-232로 통신한다.



Fig. 2.9 Motor driver

Fig. 2.10은 본 연구에 사용된 DC 서보모터 IG-52GM 04 TYPE를 나타내고, 24V를 사용한다.



Fig. 2.10 DC servo motor

Table 2.3은 모터의 세부제원을 나타낸다.

Table 2.3 Specification of DC motor IG-52GM 04 TYPE

No	Parameter	Value
1	Rated voltage	24V
2	Rated output	48.6W
3	No load current	0.7A
4	Rated current	2.85A
5	Rated speed	3,550RPM
6	No load speed	4,000RPM
7	Rated torque	1,300 gf · cm

8	Weight	1,580g
9	Gear Ratio	1/12
10	Armature resistance ( $R_a$ )	$6\Omega$
11	Emf coefficient ( $k_e$ )	$6.67 \times 10^{-3} \text{Vs/rad}$
12	Torque coefficient ( $k_t$ )	0.56368J/A
13	Encoder	360Pulse/Round

## 2.2.4 전원공급장치

Fig. 2.11은 본 연구에 사용된 (주)타보스 사의 리튬이온배터리 LC-MOTOR-25V-52AH-CV470이다. 리튬이온배터리는 25V를 OAGV의 모든 시스템에 공급한다.



Fig. 2.11 Lithium-ion battery

Table 2.4는 리튬이온배터리의 제원을 나타낸다.

**Table 2.4 Specification of Lithium-ion battery**  
**LC-MOTOR -25V-52AH-CV470**

No	Parameter	Value
1	Nominal output voltage	25.2V
2	Max. directhocking output Soft start( $\Delta T \geq 1.5\text{sec}$ )	1,040W
3	Max. continuous output	620W
4	Energy	1,320Wh
5	Max. current(dischargeity)	52A
6	Max. current(charge)	26A
7	Weight	7Kg
8	Dimensions (L×W×H)	470×90×142mm

## 제 3 장 다중 목표점 경로계획 알고리즘

### 3.1 D\* Lite 알고리즘

Fig. 3.1은 D\* Lite 알고리즘을 보여준다. D\* Lite 알고리즘은 원래의 D\* 알고리즘이나 Focused D\* 알고리즘[9]을 기반으로 하진 않지만 동일한 동작을 구현한다. 심지어 Focused D\* 알고리즘 보다 성능면에서 뛰어나다. 그리고 이해하기 쉽고 더 적은 코드행으로 구현될 수 있다. 이 D\* Lite 알고리즘은 S. Koenig 등에 의해 소개된 LPA\*[7]를 기반으로 한다.

```

key(s)
01. return  $[\min(g(s), rhs(s)) + h(s_{start}, s); \min(g(s), rhs(s))]$ ;

UpdateState(s)
02. if  $s$  was not visited before
03.  $g(s) = \infty$ ;
04. if  $(s \neq s_{goal})$   $rhs(s) = \min_{s' \in Succ(s)} (c(s, s') + g(s'))$ ;
05. if  $(s \in OPEN)$  remove  $s$  from  $OPEN$ ;
06. if  $(g(s) \neq rhs(s))$  insert  $s$  into  $OPEN$  with  $key(s)$ ;

ComputeShortestPath()
07. while  $(\min_{s \in OPEN} (key(s)) < key(s_{start}) \text{ OR } rhs(s_{start}) \neq g(s_{start}))$ 
08. remove state  $s$  with the minimum key from  $OPEN$ ;
09. if  $(g(s) > rhs(s))$ 
10.  $g(s) = rhs(s)$ ;
11. for all  $s' \in Pred(s)$  UpdateState( $s'$ );
12. else
13.  $g(s) = \infty$ ;
14. for all  $s' \in Pred(s) \cup \{s\}$  UpdateState( $s'$ );

Main()
15.  $g(s_{start}) = rhs(s_{start}) = \infty$ ;  $g(s_{goal}) = \infty$ ;
16.  $rhs(s_{goal}) = 0$ ;  $OPEN = \emptyset$ ;
17. insert  $s_{goal}$  into  $OPEN$  with  $key(s_{goal})$ ;
18. forever
19. ComputeShortestPath();
20. Wait for changes in edge costs;
21. for all directed edges  $(u, v)$  with changed edge costs
22. Update the edge costs  $c(u, v)$ ;
23. UpdateState( $u$ );

```

Fig. 3.1 D\* Lite algorithm

Fig. 3.2는 각 노드에서의 D\* Lite 알고리즘의 경로계획을 보여준다. 기본적으로 D\* Lite 알고리즘은 장애물을 피하면서 최단거리의 시작점에서 목표점으로 이동한다. 굵은 실선으로 표시된 것이 D\* Lite 알고리즘으로 생성된 경로이다. 시작점 셀 (8,I)에서 목표점 셀 (4,C)로 이동하는 경로에서 최단거리로 이동하는 도중 셀 (4,G)에서 장애물의 모서리에 닿는다. 이것은 AGV의 움직임에 있어서 매우 치명적인 단점이며, AGV의 몸체를

손상시킬 수 있다. 따라서, 이러한 경로를 막기 위해서 다중 목표점을 설정하게 되었다.

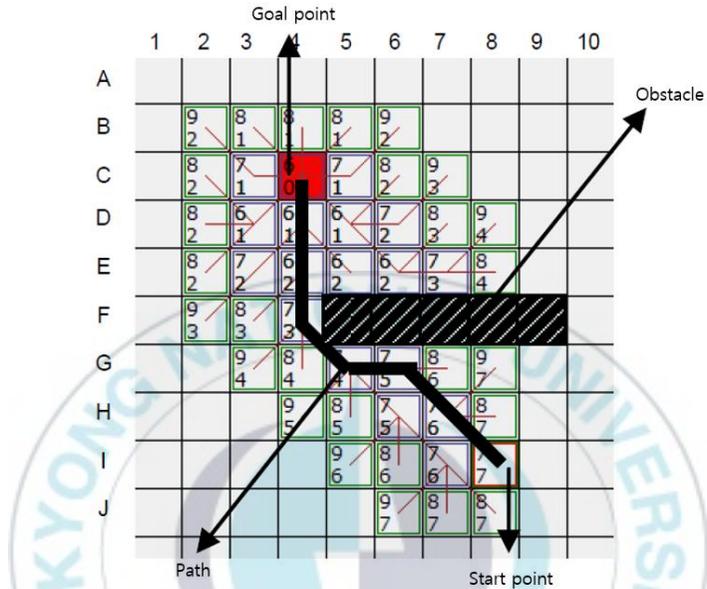


Fig. 3.2 D\* lite path planning algorithm in each node

## 3.2 다중 목표점 경로계획 알고리즘

경로계획 알고리즘에는 Dijkstra, A\*, LPA\*, D\*, D\* Lite 등과 같은 다양한 종류의 알고리즘이 있다. 그 중에서 D\* Lite 알고리즘이 가장 발전된 알고리즘이라 평가되고 있다. 그러나, 이러한 D\* Lite 알고리즘의 문제점이 존재한다. 즉, 장애물의 날카로운 모서리를 가로 지르거나 두 개의 장애물을 횡단하는 것과 같은 단점이 있다.

본 절에서는 이러한 D\* lite 경로계획 알고리즘의 단점을 보완하기 위해 D\* lite 경로계획 알고리즘을 수정하고 장애물을 피하면서 다중목표점을 지정함으로써 출발점에서 목표점까지의 최적경로를 생성하는 다중목표점 경로계획 알고리즘을 제시한다. 다중목표점 경로계획 알고리즘은 최적의 다중 목표 지점을 통해 초기 지점에서 목표 지점으로 이동시 장애물을 감지하는 레이저 센서 NAV-200으로 AGV를 제어하는데 사용되는 목표 경로를 생성시킨다.

경로계획에 대해 다음 조건이 가정된다.

<가정 1>:

1. 작업환경 및 모든 장애물의 위치에 대한 정보가 알려져 있다.
2. 각 셀의 크기는 AGV의 크기와 같다.

D\* lite 경로계획 알고리즘을 수정한 다중목표점 경로계획 알고리즘을 수행하기 위해 다음과 같은 과정을 거친다.

첫째, 작업 공간을 정사각형 셀로 나눈다. 각 셀의 중심은 노드로 지정된다[19].

둘째, 현재 노드와 이웃 노드를 연결하는 각 모서리의 비용을 계산한다. 수평 방향과 수직 방향의 비용은 1이고, 대각 방향의 비용은  $\sqrt{2} \approx 1.414$ 이며, AGV가 한 방향으로 움직이다가 장애물을 발견하면 Fig. 3.3.b와 같이 비용은  $\infty$ 가 된다.

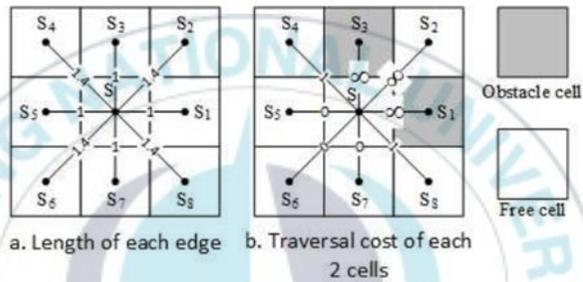


Fig. 3.3 Cost of each edge connecting current node to neighbor nodes

셋째, 초기 지점에서 모든 목표 지점까지의 최단 경로와 목표 지점에서 목표 지점을 포함한 나머지 목표 지점까지의 최단 경로를 계산하고 근접한 이웃 원칙에 따라 시작점에서 모든 다중목표점을 통과하여 최종목표점까지 갈 수 있는 해밀턴 경로[20]를 만들어 목표 경로를 생성시킨다. 해밀턴 경로는 다음과 같이 정의된다. 그래프 이론에서, 해밀턴 경로는 각 정점을 정확히 한 번 방문하는 방향성이 없거나 방향성이 있는 그래프의 단일 경로이다. 해밀턴 경로가 포함된 그래프를 추적 가능 그래프라고 한다. 각 쌍의 정점에 대해 경로 계획에서 해밀턴 경로를 생성하기 위해 두 꼭지점 사이에 해밀턴 경로가

있는 경우, 그래프는 해밀턴 연결이다. 각 정점은 시작점, 다중 목표점 및 최종목표점으로 간주된다.

아래의 Fig. 3.4는 정십이면체의 모든 꼭지점들을 통과하는 해밀턴 경로를 나타낸다. 해밀턴 경로는 정다면체의 그래프이고, 크기가 3이상인 완전그래프, 크기가 3이상인 순환그래프여야 한다. 그래프가 불연속적이거나 순환하지 않으면 해밀턴 그래프가 아니다. 또한 그래프 크기가 2이하인 꼭지점을 갖는 그래프는 해밀턴 그래프가 아니다.

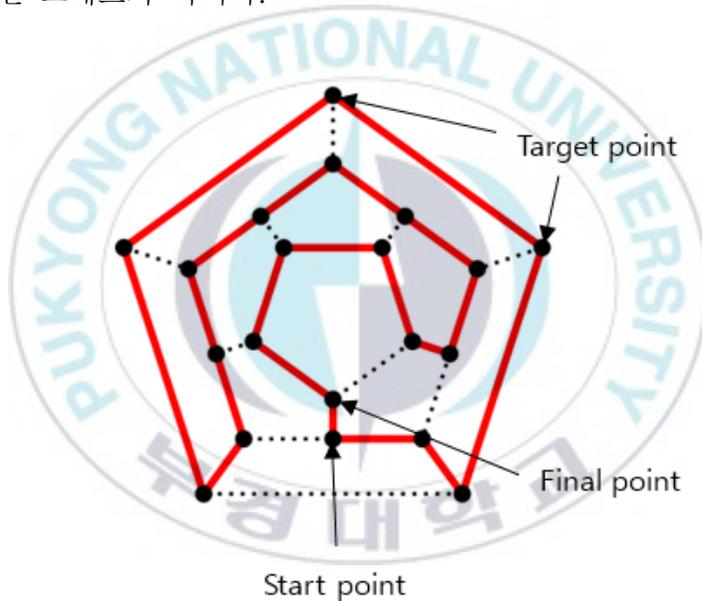


Fig. 3.4 Hamiltonian path in a dodecahedron

넷째, 레이저 센서가 장애물을 감지하고 목표지점의 갱신된 목록을 작성하는 즉시 최소비용경로를 다시 계산한다.

다섯째, 제안된 다중목표점 경로계획 알고리즘의 유효성을 검증하기 위해 시뮬레이션 결과를 보여준다. Fig. 3.5(a)에서 셀 (3,4)와 셀 (11,4)에 다중목표점을 설정하여 다중목표점 경로계획

알고리즘이 AGV가 장애물(obstacle)의 날카로운 모서리를 가로 지르는 것을 방지하며 또한, Fig. 3.5(b)에서는 셀 (7,3) 및 (10,3)에 다중목표점을 설정하여 AGV가 두 개의 장애물 사이를 가로 지르는 것을 방지하여 안전하게 장애물을 회피하며 최종 목표점에 도달할 수 있는 목표경로를 생성시킨다.

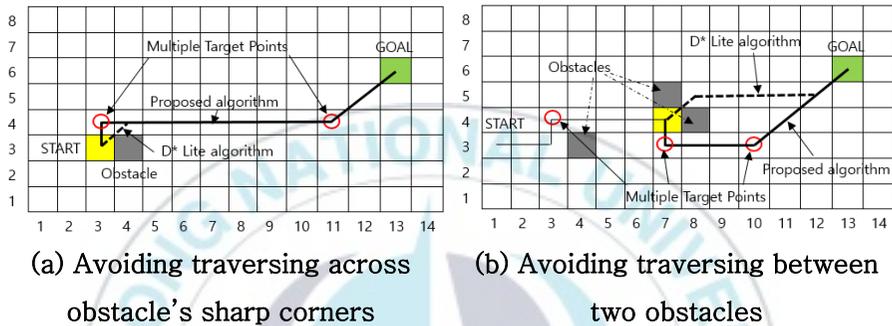


Fig. 3.5 Multiple target point path planning algorithm

마지막으로 A\* 알고리즘, D\* Lite 알고리즘과의 비교를 통해 다중목표점 경로계획 알고리즘의 유효성을 증명한다.

D\* Lite 경로계획 알고리즘은 기지환경에서 정보의 증분검색을 수행할 수 있다. 또한, 코드의 줄 수가 적고 이해하기 쉬운 장점이 있다. 그러나 때때로 D\* Lite 경로계획 알고리즘은 장애물의 날카로운 모서리를 지난다거나 두 장애물 사이를 가로지르는 단점이 있다. 따라서 시스템의 요구사항에 따라 적절하고 최적의 경로계획 알고리즘을 선택해야한다. 이러한 단점을 극복하기 위해 다중목표점을 설정함으로써 장애물로부터 안전하게 회피할 수 있다.

Fig. 3.6은 다중목표점 경로계획 알고리즘의 흐름도를 도시한다. AGV가 출발점에서 다음 목표점에 도달하면 장애물을 감지한다. 이때, AGV는 D\* Lite 알고리즘에 따라 최단경로로 목표점을 향해 이동하려 한다. 그러나, AGV가 두 장애물 사이를 통과하면 AGV가 심각하게 손상될 수 있다. 따라서, 다중 목표점을 설정함으로써 AGV를 손상시키지 않고 목표점으로 안전하게 이동시킬 수 있다.

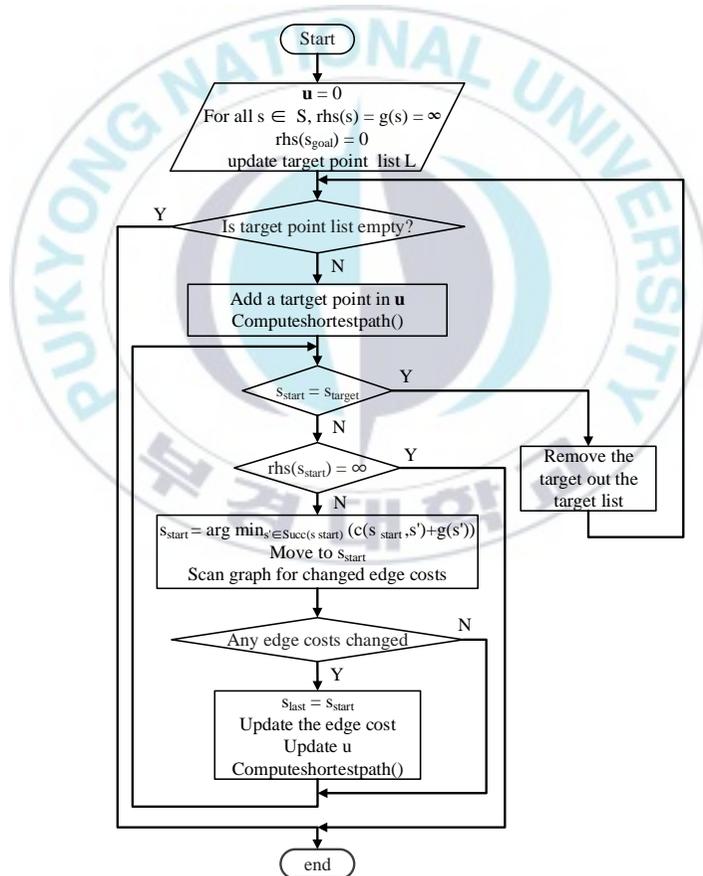


Fig. 3.6 Flowchart of the proposed multiple target point path planning algorithm

Fig. 3.7은 각 노드에서 제안된 다중목표점 경로계획 알고리즘의 시뮬레이션 예시를 보여준다. AGV가 시작점 셀 (6,H)지점에서 출발하여 최종목표점인 셀 (3,B)지점으로 최소경로로 이동한다. 굵은 실선으로 표시된 것이 다중목표점 경로계획 알고리즘으로 생성된 경로이다. 이동하는 도중 장애물의 위치를 명확하게 감지하고 회피하여 AGV몸체에 손상을 주지 않기 위해서 셀 (6,F), (3,E)지점에 목표점을 설정한다. 목표점을 설정함으로써 이제 더 이상 장애물의 날카로운 모서리를 통과하지 않는다.

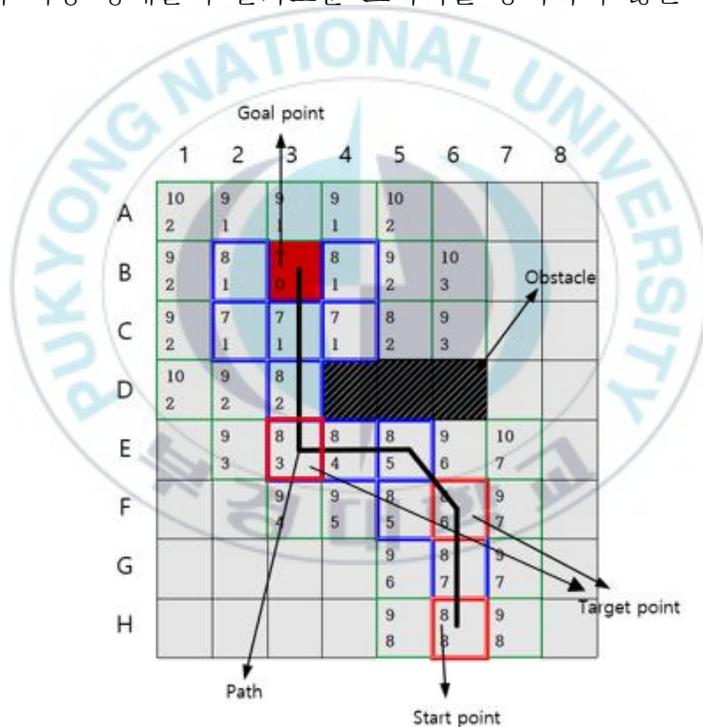


Fig. 3.7 Multiple target point path planning algorithm in each node

다중목표점 경로계획 알고리즘을 이용한 셀 간의 이동경로 생성에 대하여 A\* 알고리즘, D\* Lite 알고리즘과 비교함으로써 그

안전성을 증명하고자 한다. Fig. 3.8은 동일한 조건의 알려진 영역에서 시작점과 최종목표점 사이의 경로를 A\* 알고리즘, D\* Lite 알고리즘 및 다중목표점 경로계획 알고리즘을 이용하여 생성한 결과이다. 굵은 실선으로 표시된 것이 각 알고리즘을 이용하여 생성된 경로이다.

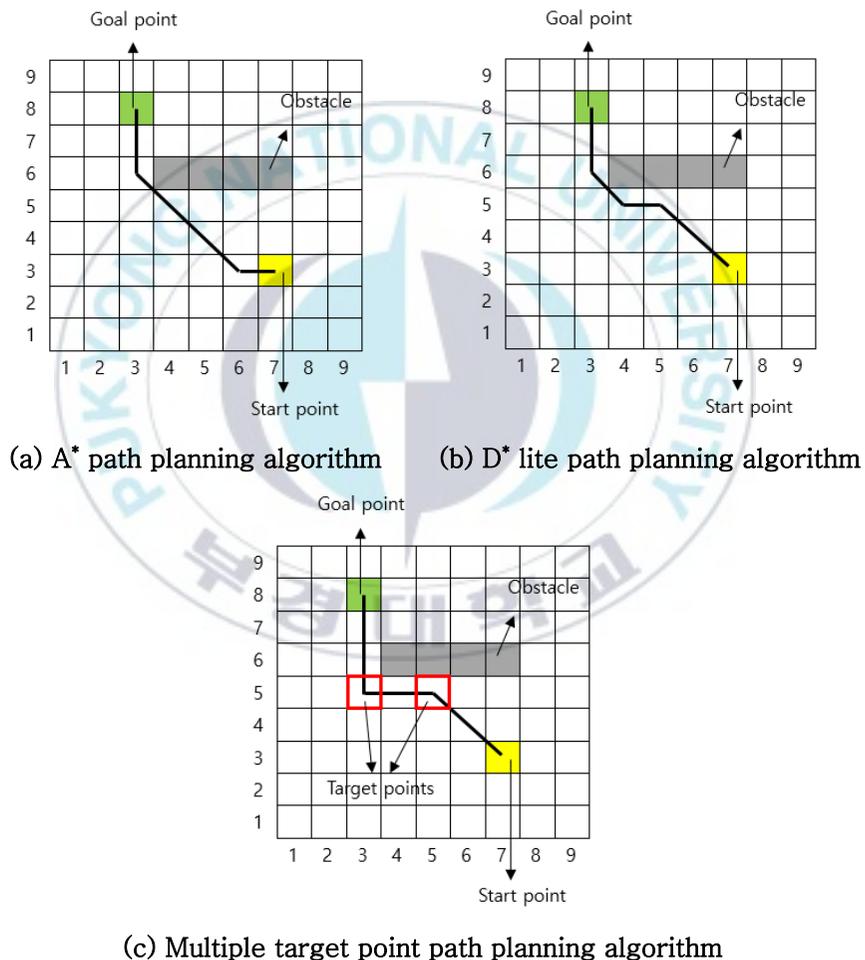


Fig. 3.8 Path comparison between A\* algorithm and D\* lite algorithm and multiple target point path planning algorithm

Fig. 3.8에서 픽셀하나의 길이가 1m 라고 정의하여 각 알고리즘의 전체 이동거리를 계산하고 장애물에 부딪히는 여부를 확인한 것이 Table 3.1과 같다. (a)의 A\* 알고리즘을 이용하여 생성된 경로를 살펴보면 직선 길이는 3m 이고 대각선 길이는  $3\sqrt{2}m(\approx 4.243m)$  이므로 총 이동경로는 7.243m 가 된다. (b)의 D\* Lite 알고리즘을 이용하여 생성된 경로를 살펴보면 직선 길이 3m 이고 대각선 길이  $3\sqrt{2}m(\approx 4.243m)$  이므로 총 이동경로는 7.243m 이다. 마지막으로 (c)의 다중목표점 경로계획 알고리즘을 이용하여 생성된 경로는 직선 길이 5m 이고 대각선 길이  $2\sqrt{2}m(\approx 2.828m)$  이므로 총 이동경로는 7.828m 이다. A\* 알고리즘과 D\* Lite 알고리즘보다 다중목표점 경로계획 알고리즘이 좀 더 긴 경로가 나왔지만, 장애물과 부딪히는 부분에서의 차이이므로 AGV의 안전성을 감안한다면 더 나은 선택이라고 볼 수 있다.

Table 3.1 Comparison between A\* algorithm and D\* lite algorithm and multiple target point path planning algorithm

Algorithm	Total distance	AGV damage
A*	7.243m	O
D* lite	7.243m	O
Multiple target point	7.828m	X

### 3.3 전체 목표점 경유 경로생성

Fig. 3.9는 실험을 위한 기지의 환경에서의 장애물 위치를 나타낸다. 실험에 사용된 장애물들은 정사각기둥, 정오각기둥, 정육각기둥이며, 실제 바닥 타일 한 칸의 가로, 세로 길이는 0.4m(400mm)로 같다.

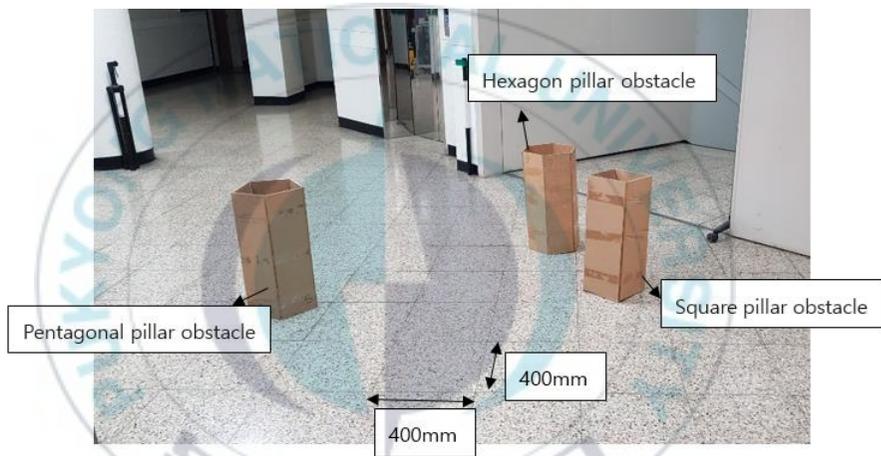
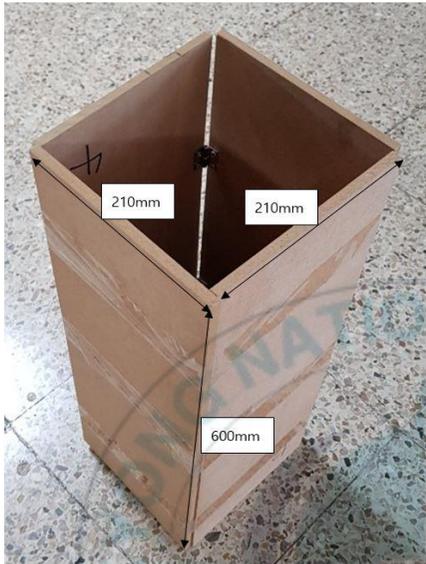


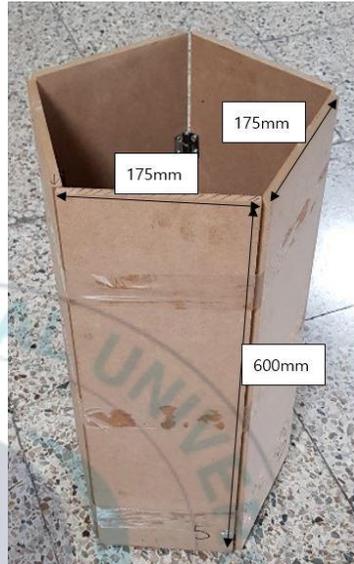
Fig. 3.9 Known environment with obstacles

Fig. 3.10은 장애물들의 종류와 치수를 나타낸다. 정사각기둥 (a)은 한 변의 길이가 0.21m(210mm) 이고, 높이가 0.6m(600mm) 이다. 정오각기둥(b)은 한 변의 길이가 0.175m(175mm) 이고, 높이가 0.6m(600mm) 이다. 정육각기둥(c)은 한 변의 길이가 0.15m (150mm) 이고, 높이가 0.6m(600mm) 이다. 장애물들의 높이가 모두 0.6m(600mm) 으로 같은 것은 OAGV의 레이저센서 (NAV-200)가

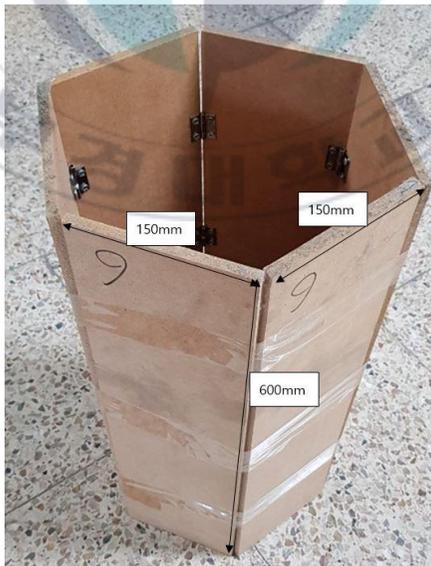
장착된 높이 0.53m(530mm)에 맞추어 제작되어 장애물을 감지하기 위함이다.



(a) Dimension of Square Pillar



(b) Dimension of Pentagonal Pillar



(c) Dimension of Hexagonal Pillar

Fig. 3.10 Types and dimensions of obstacles

Fig. 3.11에서 다중목표점 경로계획 알고리즘은 AGV가 초기 지점에서 다중목표점 1, 2를 통과하여 최종목표지점 (목표지점 3)까지의 최소거리경로가 되도록 목표경로를 생성시킨다. 굵은 실선이 다중목표점 경로계획 알고리즘에 의해 생성된 경로이다. 점선으로 표시된 절대좌표계( $X, Y$ )에서 시작점(셀 (0, 0))에서 목표점 1(셀 (500, 1000))까지의 경로는  $Y_r = 2X_r$  로 구해진다.  $\tan(\Phi_{Cr1}) = 2$  이므로  $\Phi_{Cr1} = \tan^{-1}(2) \approx 63^\circ \approx 1.1(\text{rad})$  로 구해진다. 목표점 1(셀 (500, 1000))에서 목표점 2(셀 (0, 2000))까지의 경로는  $Y_r = -2X_r + 2000$  로 구해진다.  $\tan(\Phi_{Cr2}) = -2$  이므로  $\Phi_{Cr2} = \tan^{-1}(-2) \approx -63^\circ \approx -1.1(\text{rad})$  로 구해진다. 목표점 2(셀 (0, 2000))에서 최종 목표점(셀 (1500, 2500))까지의 경로는  $Y_r = \frac{1}{3}X_r + 2000$  로 구해진다.  $\tan(\Phi_{Cr3}) = \frac{1}{3}$  이므로  $\Phi_{Cr3} = \tan^{-1}(\frac{1}{3}) \approx 18.4^\circ \approx 0.321(\text{rad})$  로 구해진다.

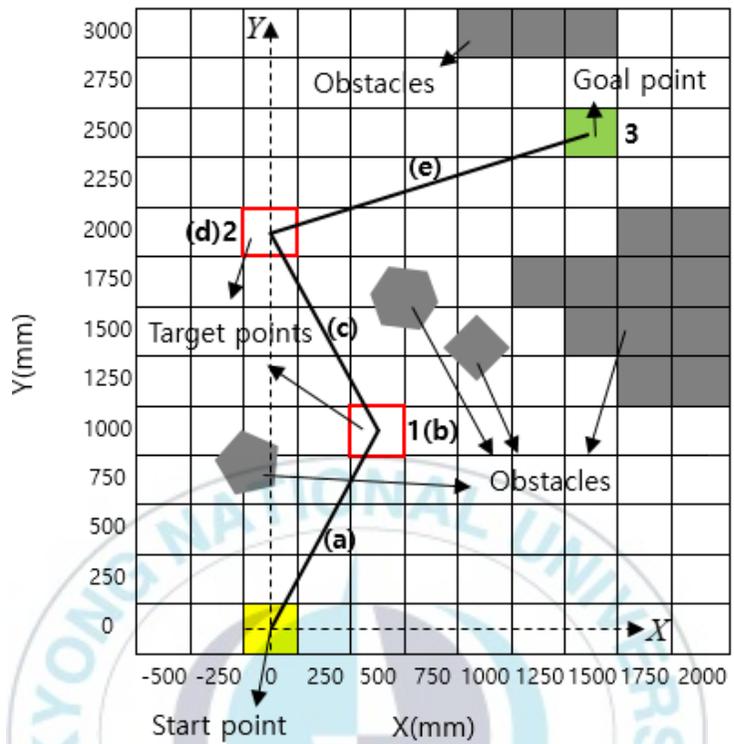


Fig. 3.11 Multiple target point path planning algorithm with optimal placement of target points

## 제 4 장 시스템 모델링 및 제어기 설계

본 장에서는 B. T. Luan[13]이 제시한 3개의 바퀴를 갖는 OAGV의 기하학적 모델링과 수학적 모델링을 제시한다. 또한 제시된 모델링을 바탕으로 제 3장의 다중목표점 경로계획 알고리즘으로 생성된 목표경로를 추적하는 경로 추적제어기를 제시한다.

### 4.1 OAGV의 기하학적 모델링

Fig. 4.1은 3개의 전방향 구동바퀴를 사용하는 OAGV의 기하학적인 모델링을 나타낸다.

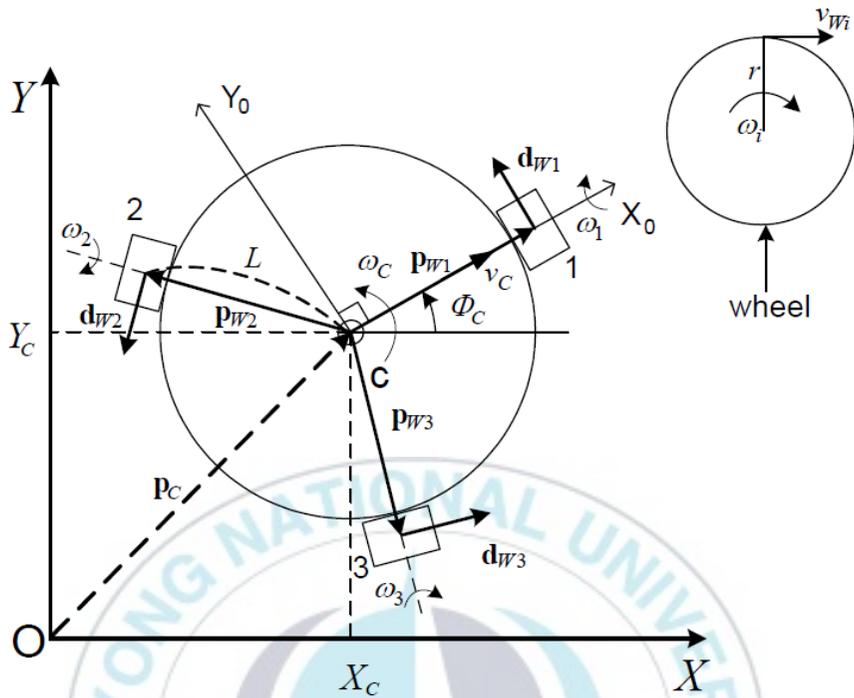
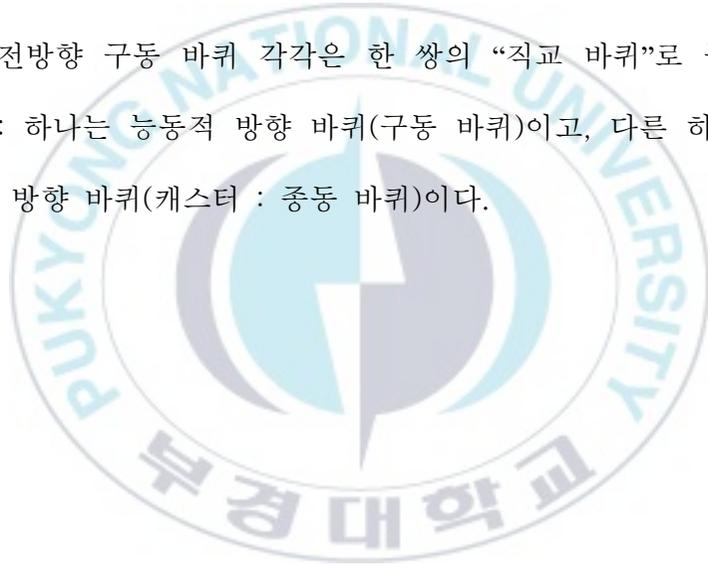


Fig. 4.1 Configuration for geometric model of the OAGV

본 연구에서 사용되는 3개의 전방향 구동바퀴를 사용한 OAGV는 바퀴 사이마다 정확히  $120^\circ$ 의 각도를 이루고 있다. 3개의 바퀴는 Fig. 4.1에서 표시된 바와 같이 같은 반지름  $r$ 을 가지며, DC서보모터에 의해 구동된다.  $L$ 은 OAGV의 기하학적중심  $C$ 로부터 바퀴들까지의 거리를 나타낸다.  $\mathbf{p}_{w_i} \in \mathbb{R}^{2 \times 1} (i=1,2,3)$ 는 OAGV의 기하학적중심  $C$ 로부터의 지역이동좌표계(local coordinate frame)  $CX_0Y_0$ 에 대한 각 바퀴의 위치벡터이다.  $\mathbf{d}_{w_i} \in \mathbb{R}^{2 \times 1} (i=1,2,3)$ 는 전역좌표계(global coordinate frame)  $OXY$

에 대한 각 바퀴의 구동방향벡터이다.  $CX_0$  축은 바퀴 1에 정렬되며,  $\phi_c$  는 전역좌표축  $X$  와 지역좌표축  $CX_0$  사이의 각도이다. OAGV에서 전역좌표계 중심  $C$  의 자세벡터는  $\mathbf{q}_c = [X_c \ Y_c \ \phi_c]^T$  에 의해 설정되며,  $\mathbf{p}_c = [X_c \ Y_c]^T$  는 전역좌표계 중심  $C$  의 위치 벡터로 정의된다.  $v_c$  는 OAGV의 선속도이며,  $v_{w_i}$  와  $\omega_i$  는 각각  $i(i=1,2,3)$  번째 바퀴의 선속도와 각속도이다.

전방향 구동 바퀴 각각은 한 쌍의 “직교 바퀴”로 구성되어 있다: 하나는 능동적 방향 바퀴(구동 바퀴)이고, 다른 하나는 수동적 방향 바퀴(캐스터 : 종동 바퀴)이다.



## 4.2 OAGV의 수학적 모델링

본 절에서는 OAGV의 기구학적 모델링과 동역학적 모델링을 제시한다.

### 4.2.1 기구학적 모델링

OAGV의 ‘기구학적 모델링’이 다음과 같은 가정 하에 유도된다.

<가정 1>:

1. 전방향구동 바퀴는 순수 구름접촉을 하며 미끄러짐이 없다(pure-rolling and no-slip).
2. 각 링크들은 강체이다.
3. 하중이 주어지기 전이나 후나 OAGV의 무게중심과 기하학적인 중심은 일치한다.

지역좌표에서 전역좌표까지의 회전행렬  $\mathbf{R}_1(\Phi_C)$ 는 다음과 같이 주어진다:

$$\mathbf{R}_1(\Phi_C) = \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \quad (4.1)$$

$\mathbf{p}_{w_i}$  ( $i=1,2,3$ ) 는 이동좌표계에 대한 각 바퀴의 위치벡터로써 회전행렬에 의해 다음과 같이 주어진다.

$$\begin{aligned} \mathbf{p}_{w_1} &= L \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{p}_{w_2} = \mathbf{R}_1 \left( \frac{2\pi}{3} \right) \times \mathbf{p}_{w_1} = \frac{L}{2} \begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix}, \\ \mathbf{p}_{w_3} &= \mathbf{R}_1 \left( \frac{4\pi}{3} \right) \times \mathbf{p}_{w_1} = -\frac{L}{2} \begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} \end{aligned} \quad (4.2)$$

이와 유사하게  $\mathbf{d}_{w_i} \in \mathcal{R}^{2 \times 1}$  ( $i=1,2,3$ ) 은 지역좌표계와 관련된 단위벡터이고, 각 바퀴의 구동방향벡터를 지정하는 벡터는 다음과 같이 주어진다.

$$\mathbf{d}_{w_i} = \frac{1}{L} \mathbf{R} \left( \frac{\pi}{2} \right) \times \mathbf{p}_{w_i}, \mathbf{d}_{w_1} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \mathbf{d}_{w_2} = -\frac{1}{2} \begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix}, \mathbf{d}_{w_3} = \frac{1}{2} \begin{bmatrix} \sqrt{3} \\ -1 \end{bmatrix} \quad (4.3)$$

여기서,  $\mathbf{p}_i$  는  $i$  번째 바퀴의 위치벡터이고,  $\mathbf{v}_{g_i} = \dot{\mathbf{p}}_{g_i}$  는  $i$  번째 바퀴의 선속도벡터이다.

한편, 전역좌표계와 관련된 OAGV의 각 바퀴는 다음에 의해 주어진다.

$$\mathbf{p}_{gi} = \mathbf{p}_C + \mathbf{R}_1(\Phi_C) \times \mathbf{p}_{wi}, \quad (4.4)$$

$$\mathbf{v}_{gi} = \dot{\mathbf{p}}_C + \dot{\mathbf{R}}_1(\Phi_C) \times \mathbf{p}_{wi} \quad (4.5)$$

$i$  번째 바퀴 ( $i=1,2,3$ )의 각속도는 다음과 같다.

$$\omega_i = \frac{1}{r} \mathbf{v}_{gi}^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{wi} \quad (4.6)$$

식 (4.5)의  $\mathbf{v}_{gi}$ 를 식 (4.6)에 대입한 결과는 다음과 같다.

$$\omega_i = \frac{1}{r} \left[ \dot{\mathbf{p}}_C^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{wi} + \mathbf{p}_{wi}^T \dot{\mathbf{R}}_1^T(\Phi_C) \mathbf{R}_1(\Phi_C) \mathbf{d}_{wi} \right] \quad (4.7)$$

식 (4.7)로부터, 가정 1에 의해 OAGV의 기구학적 방정식을 다음과 같이 세울 수 있다.

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) & L \\ -\sin(\frac{\pi}{3} - \Phi_C) & -\cos(\frac{\pi}{3} - \Phi_C) & L \\ \sin(\frac{\pi}{3} + \Phi_C) & -\cos(\frac{\pi}{3} + \Phi_C) & L \end{bmatrix} \begin{bmatrix} \dot{X}_C \\ \dot{Y}_C \\ \dot{\Phi}_C \end{bmatrix} \quad (4.8)$$

식 (4.8)에 대한 증명은 부록 A에 나타나 있다.

식 (4.8)은 다음 식과 같이 간략히 표현할 수 있다.

$$\boldsymbol{\omega} = \frac{\mathbf{v}_w}{r} = \frac{1}{r} \mathbf{H}^{-1}(\Phi_C) \dot{\mathbf{q}}_C \quad (4.9)$$

여기서,  $\boldsymbol{\omega} = [\omega_1 \ \omega_2 \ \omega_3]^T$  는 바퀴의 각속도벡터이고,  $\mathbf{v}_w = [v_{w1} \ v_{w2} \ v_{w3}]^T$  는 바퀴의 선속도벡터이며,  $\mathbf{H}^{-1} \in \mathbb{R}^{3 \times 3}$  행렬은 다음과 같이 주어진다.

$$\mathbf{H}^{-1}(\Phi_C) = \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) & L \\ -\sin(\frac{\pi}{3} - \Phi_C) & -\cos(\frac{\pi}{3} - \Phi_C) & L \\ \sin(\frac{\pi}{3} + \Phi_C) & -\cos(\frac{\pi}{3} + \Phi_C) & L \end{bmatrix} \quad (4.10)$$

식 (4.9)로부터 바퀴의 선속도벡터는 다음과 같다.

$$\mathbf{v}_w = r\boldsymbol{\omega} = \mathbf{H}^{-1}(\Phi_C) \dot{\mathbf{q}}_C \quad (4.11)$$

식 (4.9)로부터 OAGV의 중심  $C$ 의 속도벡터는 기구학적 모델링으로 정의된다.

$$\dot{\mathbf{q}}_C = r\mathbf{H}(\Phi_C)\boldsymbol{\omega} = \mathbf{H}(\Phi_C)\mathbf{v}_W \quad (4.12)$$

여기서,

$$\mathbf{H}(\Phi_C) = \frac{1}{3} \begin{bmatrix} -2\sin(\Phi_C) & \sin(\Phi_C) - \sqrt{3}\cos(\Phi_C) & \sin(\Phi_C) + \sqrt{3}\cos(\Phi_C) \\ -2\cos(\Phi_C) & -[\sqrt{3}\sin(\Phi_C) + \cos(\Phi_C)] & \sqrt{3}\sin(\Phi_C) - \cos(\Phi_C) \\ \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \end{bmatrix}$$

Fig. 4.1로부터 OAGV의 선속도와 각속도는 다음과 같이 얻어진다.

$$\begin{bmatrix} v_C \\ \boldsymbol{\omega}_C \end{bmatrix} = \begin{bmatrix} \cos(\Phi_C) & \sin(\Phi_C) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{X}_C \\ \dot{Y}_C \\ \dot{\Phi}_C \end{bmatrix} \quad (4.13)$$

#### 4.2.2 동역학적 모델링

OAGV의 ‘동역학적 모델링’은 다음과 같은 가정 하에 유도된다.

<가정 2>:

1. 바닥면과 바퀴사이의 마찰과 미끄러짐현상으로 인한 경계 외란 벡터는 알려져 있지 않다.
2. 각 링크들은 강체이다.

3. 하중이 주어지기 전이나 후나 OAGV의 무게중심과 기하학적인 중심은 일치한다.

마찰과 미끄러짐으로 인한 OAGV에 작용하는 힘은 다음 Fig. 4.2와 같다.

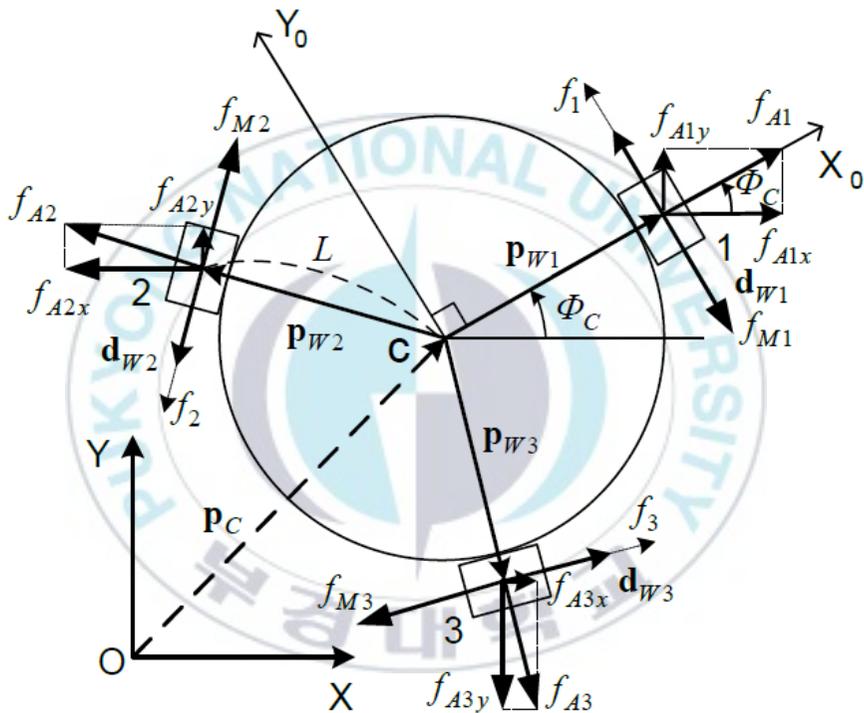


Fig. 4.2 Force acting on OAGV due to friction and slip

Fig. 4.2에서  $\mathbf{f}_A$ 는 전역좌표계와 관련된 바닥면과 바퀴사이의 마찰, 미끄러짐현상으로 인해 발생하는  $f_{Ai}$ 의 힘외란벡터이다.  $f_{Mi}$ 와  $f_{Ai}(i=1,2,3)$ 는 각각  $i$ 번째 바퀴의 작용방향 및 중심축방

향의 마찰력 크기이다.  $f_{Aix}, f_{Aiy}$  는 전역좌표계와 관련된  $i$  번째 바퀴의  $f_{Ai}$  벡터의  $x, y$  방향 마찰력의 크기이다.  $\ddot{\mathbf{p}}_C = \mathfrak{R}^{2 \times 1}$  은 전역좌표계에서 OAGV의 선형가속도 벡터이고,  $f_i (i=1,2,3)$  는  $i$  번째 모터로부터 바퀴가 받는 구동방향의 힘크기이다.  $m_1$  은 OAGV의 질량이며,  $I$  는 OAGV의 기하학적 중심  $C$  와 무게중심이 일치하는 관성모멘트이다.

힘과 모멘트의 평형원리에 의해 뉴턴의 제 2법칙에 따라, 전역좌표계에서 표면 마찰과 미끄러짐현상으로 야기되는 힘외란벡터를 포함한 OAGV의 균형을 잡아주는 힘과 모멘트 방정식이 다음과 같이 주어진다.

$$\sum_{i=1}^3 (f_i - f_{Mi}) \mathbf{R}_1(\Phi_C) \mathbf{d}_{wi} - \mathbf{f}_A = m_1 \ddot{\mathbf{p}}_C \quad (4.14)$$

$$\mathbf{f}_A = \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \quad (4.15)$$

$$L \sum_{i=1}^3 (f_i - f_{Mi}) = I \ddot{\Phi}_C \quad (4.16)$$

여기서,  $m_1 = M + m$  이며,  $M$  은 OAGV의 질량이고  $m$  은 OAGV에 가해진 하중의 질량이다.

DC서보모터의 동역학적 특성으로부터, DC서보모터로부터 생

성된 바퀴에 발생하는 힘은 다음과 같이 설명할 수 있다.

$$f_i = \rho_1 u_i - \rho_2 v_{wi} \quad (4.17)$$

여기서  $\rho_1$  과  $\rho_2$  는 DC모터의 카탈로그에서 얻을 수 있는 매개변수에 따른 모터 특성계수이다. 그리고  $u_i (i=1,2,3)$  은 DC모터에 인가되는 전압이며,  $v_{wi} (i=1,2,3)$  은 각 바퀴의 선속도이다. 식 (4.17)에 대한 증명은 부록 B에 나타나 있다.

식 (4.14)~(4.16)으로부터, OAGV의 동역학적 모델링으로 정의된다.

$$\begin{bmatrix} m_1 \ddot{X}_c \\ m_1 \ddot{Y}_c \\ I \ddot{\phi}_c \end{bmatrix} + \frac{3}{2} \rho_2 \begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ 2L^2 \dot{\phi}_c \end{bmatrix} = \rho_1 \mathbf{H}^T \mathbf{u} - \mathbf{f}_d \quad (4.18)$$

여기서,  $\mathbf{f}_d = [f_{1d} \ f_{2d} \ f_{3d}]^T \in \mathfrak{R}^{3 \times 1}$  는 전역좌표계와 관련된 OAGV의 힘외란벡터(전체의 마찰과 미끄러짐을 포함)이다[21].

$$\begin{cases}
f_{1d} = -f_{M1} \sin(\Phi_C) - f_{M2} \sin\left(\frac{\pi}{3} - \Phi_C\right) + f_{M3} \sin\left(\frac{\pi}{3} + \Phi_C\right) \\
\quad + f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\
f_{2d} = f_{M1} \cos(\Phi_C) - f_{M2} \cos\left(\frac{\pi}{3} - \Phi_C\right) - f_{M3} \cos\left(\frac{\pi}{3} + \Phi_C\right) \\
\quad + f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\
f_{3d} = L(f_{M1} + f_{M2} + f_{M3})
\end{cases} \quad (4.19)$$

$$\begin{cases}
-\frac{m_1 g}{3} \mu_{M \max} \leq f_{Mi} \leq \frac{m_1 g}{3} \mu_{M \max} \Rightarrow |f_{Mi}| \leq \frac{m_1 g}{3} \mu_{M \max} \\
-\frac{m_1 g}{3} \mu_{A \max} \leq f_{Ai} \leq \frac{m_1 g}{3} \mu_{A \max} \Rightarrow |f_{Ai}| \leq \frac{m_1 g}{3} \mu_{A \max} \quad (i=1, 2, 3)
\end{cases} \quad (4.20)$$

여기서,  $\mu_{M \max}$  와  $\mu_{A \max}$  는 각각 바퀴의 구동 방향과 바퀴의 축 방향의 최대정지마찰계수이다.  $g$  는 중력가속도이며,  $f_{Mi}$  과  $f_{Ai}$  의 미지의 경계로 나타나는 불확실한 마찰이나 미끄러짐으로 생기는 힘외란벡터가 불평등을 초래한다. 즉,  $\|\mathbf{f}_d\|_\infty \leq k_{\max}$  이다. 여기서,  $\|\mathbf{f}_d\|_\infty$  는 벡터  $\mathbf{f}_d$  의 무한대 놈, 즉 크기를 나타내며,  $k_{\max}$  는  $\|\mathbf{f}_d\|_\infty$  의 최소상한이다. 식 (4.18)에 대한 증명은 부록 C에 나타나 있다.

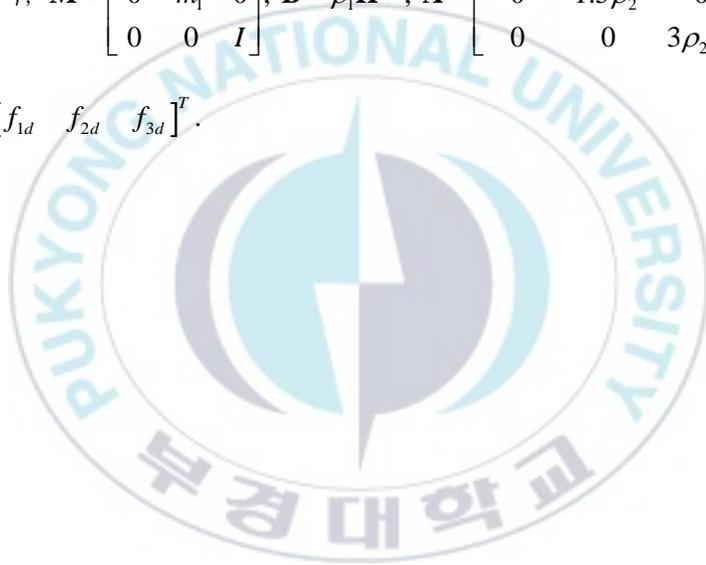
OAGV의 위치벡터를  $\mathbf{x}_1 = \mathbf{q}_C = [X_C \ Y_C \ \Phi_C]^T$  로 정의함으로써 속도벡터는  $\mathbf{x}_2 = \dot{\mathbf{q}}_C = [\dot{X}_C \ \dot{Y}_C \ \dot{\Phi}_C]^T$  로 얻어진다.

따라서, 식 (4.18) 즉 동역학적 모델링으로부터 다음을 얻을 수 있다.

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 \\ \mathbf{M}\dot{\mathbf{x}}_2 = -\mathbf{A}\mathbf{x}_2 + \mathbf{B}\mathbf{u} - \mathbf{f}_d \end{cases} \quad (4.21)$$

$$\text{여기서, } \mathbf{M} = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_1 & 0 \\ 0 & 0 & I \end{bmatrix}, \mathbf{B} = \rho_1 \mathbf{H}^{-T}, \mathbf{A} = \begin{bmatrix} 1.5\rho_2 & 0 & 0 \\ 0 & 1.5\rho_2 & 0 \\ 0 & 0 & 3\rho_2 L^2 \end{bmatrix},$$

$$\mathbf{f}_d = [f_{1d} \quad f_{2d} \quad f_{3d}]^T.$$



### 4.3 OAGV 제어기 설계

본 절에서는 백스텝핑법[26-28]을 이용하여 3장에서 제시된 다중목표점 경로계획법에 의해 생성된 OAGV의 목표경로를 추적하는 비선형(Nonlinear) 경로추적제어기를 다음과 같이 설계한다.

첫째, OAGV의 동역학적 방정식 (4.21)과 식 (4.10)으로부터 다음식이 유도된다.

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 = \mathbf{A}^* \mathbf{x}_2 + \mathbf{B}^* \mathbf{u} + \mathbf{f}_d^* \end{cases} \quad (4.22)$$

여기서,  $\mathbf{A}^* = -\mathbf{M}^{-1}\mathbf{A} = \begin{bmatrix} \frac{-1.5\rho_2}{m_1} & 0 & 0 \\ 0 & \frac{-1.5\rho_2}{m_1} & 0 \\ 0 & 0 & \frac{-3\rho_2 L^2}{I} \end{bmatrix}$ ,

$$\mathbf{H}^* = -\mathbf{M}^{-1}\mathbf{H}^{-T} = \begin{bmatrix} \frac{\sin(\Phi_C)}{m_1} & \frac{\sin\left(\frac{\pi}{3} - \Phi_C\right)}{m_1} & \frac{\sin\left(\frac{\pi}{3} + \Phi_C\right)}{m_1} \\ \frac{\cos(\Phi_C)}{m_1} & \frac{\cos\left(\frac{\pi}{3} - \Phi_C\right)}{m_1} & \frac{\cos\left(\frac{\pi}{3} + \Phi_C\right)}{m_1} \\ \frac{L}{I} & \frac{L}{I} & \frac{L}{I} \end{bmatrix},$$

$$\mathbf{B}^* = \mathbf{M}^{-1}\mathbf{B} = \rho_1 \mathbf{H}^*, \mathbf{f}_d^* = -\mathbf{M}^{-1}\mathbf{f}_d = \left[ -\frac{f_{1d}}{m_1} \quad -\frac{f_{2d}}{m_1} \quad -\frac{f_{3d}}{m_1} \right]^T, \|\mathbf{f}_d^*\|_\infty \leq k_{\max}.$$

둘째, 유도된 OAGV의 동역학적 방정식을 바탕으로 다중목표점 경로계획 알고리즘에서 생성된 경로를 추적하는 백스텝핑법을 적용한 비선형 경로추적제어기를 다음과 같이 설계한다. 백스텝핑법은 다음과 같이 5가지 단계로 이루어진다.

1. 가상 상태 변수벡터를 정의한다.
2. 첫 번째 가상상태 변수벡터에 대해서 CLF(Control Lyapunov Function)을 정의한다[29,30].
3. 가상의 피드백 제어기벡터를 선택한다. 이를 이용한 Lyapunov 안정성의 성립여부를 점검한다. 시스템의 Lyapunov 함수  $V \geq 0$  이고 이를 미분한  $\dot{V} \leq 0$  이면, 이 시스템은 점근적 안정 또는 안정될 것이다.
4. 3의 경우 Lyapunov 안정성 여부가 판별되지 않을 경우 전체 시스템에 대한 CLF를 정의하고, 1계 도함수를 구한다.
5. 전체 시스템에 대한 CLF를 이용한 Lyapunov 안정성을 만족시키는 제어치벡터  $\mathbf{u}$  를 선정한다. 시스템의 Lyapu-

nov 함수  $V \geq 0$  이고 이를 미분한  $\dot{V} \leq 0$  이면, 이 시스템은 점근적 안정 또는 안정될 것이다.

제 1 단계 : 제 3 장에서 얻어진 다중목표점 경로계획 알고리즘에서 생성된 목표경로를  $\mathbf{y}_r = [X_r, Y_r, \phi_r]^T$  으로 설정 후, 가상 상태 변수벡터를 다음과 같이 정의한다.

$$\begin{cases} \mathbf{z}_1 = \mathbf{x}_1 - \mathbf{y}_r \\ \mathbf{z}_2 = \mathbf{x}_2 - \boldsymbol{\alpha} - \dot{\mathbf{y}}_r \end{cases} \quad (4.23)$$

여기서,  $\mathbf{z}_1$  과  $\mathbf{z}_2$  는 추적오차벡터들이라 볼 수 있다.  $\boldsymbol{\alpha}$  는 피드백 제어기벡터이다.

식 (4.22)와 (4.23)으로부터 다음이 얻어진다.

$$\dot{\mathbf{z}}_1 = \dot{\mathbf{x}}_1 - \dot{\mathbf{y}}_r = \mathbf{x}_2 - \dot{\mathbf{y}}_r = (\mathbf{z}_2 + \boldsymbol{\alpha} + \dot{\mathbf{y}}_r) - \dot{\mathbf{y}}_r = \mathbf{z}_2 + \boldsymbol{\alpha} \quad (4.24)$$

제 2 단계 : 가상상태 변수벡터  $\mathbf{z}_1$  에 대해서 CLF를 다음과 같이 정의한다.

$$V = \frac{1}{2} \mathbf{z}_1^T \mathbf{z}_1 \quad (4.25)$$

식 (4.24)와 (4.25)를 이용하여  $V$ 의 1계 도함수는 다음과 같이 구할 수 있다.

$$\dot{V} = \mathbf{z}_1^T \dot{\mathbf{z}}_1 = \mathbf{z}_1^T (\mathbf{z}_2 + \boldsymbol{\alpha}) \quad (4.26)$$

제 3 단계 :  $\dot{V} \leq 0$  (*stable*)을 만들어주기 위해서 가상의 피드백 제어기를 다음과 같이 선정한다.

$$\boldsymbol{\alpha} = -k_1 \mathbf{z}_1 \quad (4.27)$$

여기서,  $k_1 \geq 0$ .

식 (4.27)을 식(4.26)에 대입하여 정리하면 다음과 같다.

$$\dot{V} = \mathbf{z}_1^T (\mathbf{z}_2 - k_1 \mathbf{z}_1) = -k_1 \mathbf{z}_1^T \mathbf{z}_1 + \mathbf{z}_1^T \mathbf{z}_2 \quad (4.28)$$

식 (4.28)만으로  $\dot{V} \leq 0$ 인지 알 수 없으므로 안정여부가 판별되지 않으므로 새로운 CLF를 선정한다.

제 4 단계 : 전체 시스템에 대한 CLF를 다음과 같이 선정한다.

$$V_a = \frac{1}{2} \mathbf{z}_1^T \mathbf{z}_1 + \frac{1}{2} \mathbf{z}_2^T \mathbf{z}_2 \quad (4.29)$$

식 (4.22)~(4.24), (4.27)을 이용하면, 식 (4.23)의 1계 도함수는 다음과 같이 구해진다.

$$\dot{\mathbf{z}}_1 = \mathbf{z}_2 - k_1 \mathbf{z}_1 \quad (4.30)$$

$$\begin{aligned} \dot{\mathbf{z}}_2 &= \dot{\mathbf{x}}_2 - \dot{\boldsymbol{\alpha}} - \ddot{\mathbf{y}}_r = \mathbf{A}^* \mathbf{x}_2 + \mathbf{B}^* \mathbf{u} + \mathbf{f}_d^* + k_1 \dot{\mathbf{z}}_1 - \ddot{\mathbf{y}}_r \\ &= \mathbf{A}^* \mathbf{x}_2 + \mathbf{B}^* \mathbf{u} + \mathbf{f}_d^* + k_1 (\mathbf{z}_2 - k_1 \mathbf{z}_1) - \ddot{\mathbf{y}}_r \end{aligned} \quad (4.31)$$

식 (4.30)과 식 (4.31)을 이용하면, 식 (4.29)의 1계 도함수는 다음과 같이 얻어진다.

$$\begin{aligned} \dot{V}_a &= \mathbf{z}_1^T \dot{\mathbf{z}}_1 + \mathbf{z}_2^T \dot{\mathbf{z}}_2 = \mathbf{z}_1^T (\mathbf{z}_2 - k_1 \mathbf{z}_1) \\ &\quad + \mathbf{z}_2^T \{ \mathbf{A}^* \mathbf{x}_2 + \mathbf{B}^* \mathbf{u} + \mathbf{f}_d^* + k_1 (\mathbf{z}_2 - k_1 \mathbf{z}_1) - \ddot{\mathbf{y}}_r \} \\ &= -k_1 \mathbf{z}_1^T \mathbf{z}_1 + \mathbf{z}_2^T \{ \mathbf{z}_1 + \mathbf{A}^* \mathbf{x}_2 + \mathbf{B}^* \mathbf{u} + \mathbf{f}_d^* + k_1 (\mathbf{z}_2 - k_1 \mathbf{z}_1) - \ddot{\mathbf{y}}_r \} \end{aligned} \quad (4.32)$$

**제 5 단계** : 시스템 안정화를 도모하기 위해,  $\dot{V}_a \leq 0$  이도록 제어척벡터  $\mathbf{u}$  를 다음과 같이 선택한다.

$$\mathbf{u} = (\mathbf{B}^*)^{-1} \{ -\mathbf{A}^* \mathbf{x}_2 - \mathbf{f}_d^* + \ddot{\mathbf{y}}_r - (k_1 + k_2) \mathbf{z}_2 + (k_1^2 - 1) \mathbf{z}_1 \} \quad (4.33)$$

여기서,  $k_2 \geq 0$ .

식 (4.32)에 식 (4.33)을 대입하면, 다음 식이 얻어진다.

$$\dot{V}_a = -k_1 \mathbf{z}_1^T \mathbf{z}_1 - k_2 \mathbf{z}_2^T \mathbf{z}_2 \leq 0 \quad (4.34)$$

따라서, 식 (4.33)의 제어칙벡터로 주어질 때, 식 (4.29)와 식 (4.34)로부터 **Barbalat's lemma**에 의해  $t \rightarrow \infty$ 일 때,  $\mathbf{z}_1 \rightarrow 0$ 이고,  $\mathbf{z}_2 \rightarrow 0$ 이다. 그러므로 식 (4.33)은 시스템을 점근적으로 안정케 한다.

식 (4.33)을 식 (4.31)에 대입하여 정리하면 다음과 같다.

$$\dot{\mathbf{z}}_2 = -\mathbf{z}_1 - k_2 \mathbf{z}_2 \quad (4.35)$$

위의 식 (4.31)과 식 (4.35)를 행렬화한 페루프 시스템은 다음과 같이 유도된다.

$$\dot{\mathbf{z}} = \mathbf{A}_z \mathbf{z} \quad (4.36)$$

여기서,  $\mathbf{A}_z = \begin{bmatrix} -k_1 & 1 \\ -1 & -k_2 \end{bmatrix}$ ,  $\mathbf{z} = [\mathbf{z}_1^T \quad \mathbf{z}_2^T]^T$ .

$\mathbf{A}_z$ 가 **Hurwitz** 행렬이면 시스템은 안정하다. 즉,  $\text{Re}(\lambda) < 0$ 이면, 시스템은 안정하므로 다음과 같이 증명된다.

$$\begin{aligned}
& |\lambda \mathbf{I} - \mathbf{A}_z| = 0 \\
& \Leftrightarrow \begin{vmatrix} \lambda + k_1 & -1 \\ 1 & \lambda + k_2 \end{vmatrix} = 0 \\
& \Leftrightarrow (\lambda + k_1)(\lambda + k_2) + 1 = 0 \\
& \Leftrightarrow \lambda^2 + (k_1 + k_2)\lambda + (k_1 k_2 + 1) = 0 \\
& \Leftrightarrow \lambda = \frac{-(k_1 + k_2) \pm \sqrt{(k_1 + k_2)^2 - 4(k_1 k_2 + 1)}}{2} = \frac{-H \pm \sqrt{D}}{2}
\end{aligned}$$

여기서,  $H = (k_1 + k_2)$  라 두고,  $D = (k_1 + k_2)^2 - 4(k_1 k_2 + 1)$  라 두면,  $\text{Re}(\lambda) < 0$  가 되기 위해서 다음과 같은 3가지 경우로 나뉜다. 그런데,  $D = 0$  일 경우와  $D < 0$  일 경우는 자동적으로  $\text{Re}(\lambda) < 0$  이 된다.

$$\begin{cases}
1) D > 0 \\
2) D = 0 \Rightarrow \lambda = -\frac{(k_1 + k_2)}{2} < 0 \\
3) D < 0 \Rightarrow \text{Re}(\lambda) = -\frac{(k_1 + k_2)}{2} < 0
\end{cases}$$

따라서,  $D > 0$  일 경우에 대해 살펴보면,  $H^2 - D > 0 \Rightarrow (H + \sqrt{D})(H - \sqrt{D}) > 0$  가 되므로

$$\begin{cases}
1) H > \sqrt{D} > 0 \Rightarrow H^2 - D = (k_1 + k_2)^2 - \{(k_1 + k_2)^2 - 4(k_1 k_2 + 1)\} = 4(k_1 k_2 + 1) > 0 \\
\text{이므로 } \lambda = \frac{-H + \sqrt{D}}{2} < 0 \text{ 이 성립한다.} \\
2) H < \sqrt{D} < 0 \Rightarrow \text{이러한 경우는 모순이다. } (\because k_1, k_2 > 0)
\end{cases}$$

여기서  $k_1, k_2$  는 제어이득(Control gain)이다.

그러므로  $\text{Re}(\lambda) < 0$  이므로 시스템이 안정하다는 것을 보여준다.

Fig. 4.3은 참조목표점에 기반한 OAGV 시스템을 위한 제안된 백스텝핑 제어기에 대한 블록선도를 나타낸 것이다.

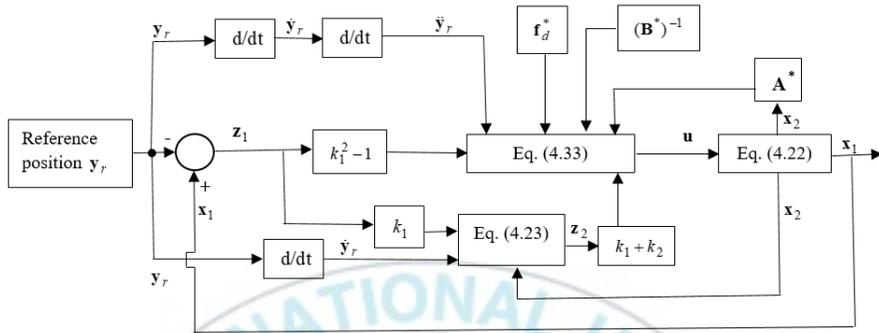


Fig. 4.3 Block diagram of the proposed backstepping control

## 제 5 장 시뮬레이션 및 실험 결과

본 장에서는 3장과 4장에서 제안된 다중목표점 경로계획 알고리즘에 의해 3.3절에서 목표경로를 생성했다. 이를 추적하기 위한 제안된 시스템 모델링 및 제어기의 유효성을 입증하기 위해 시뮬레이션과 실험결과를 제시하였다. 시뮬레이션에 사용된 매개변수 값은 Table 5.1과 같다. OAGV의 초기 위치는  $(X_0, Y_0, \phi_0) = (0, 0, 63^\circ)$  이다.

Table 5.1 Parameters of OAGV

Parameter	Description	UNIT
$r$	0.0625	m
$L$	0.275	m
$m_1$	45	kg
$I$	0.945	kgm <sup>2</sup>
$\rho_1$	1.5	N/V
$\rho_2$	1.924	kg/s
$X_{r0}$	0	m
$Y_{r0}$	0	m

$\Phi_{r0}$	1.1	rad
$k_1$	10	$s^{-1}$
$k_2$	10	$s^{-1}$
$k_{max}$	0.005	N

## 5.1 시뮬레이션 결과

Fig. 5.1은 OAGV의 구간별 목표값( $X_r$ ,  $Y_r$ ,  $\Phi_r$ )을 나타낸다.

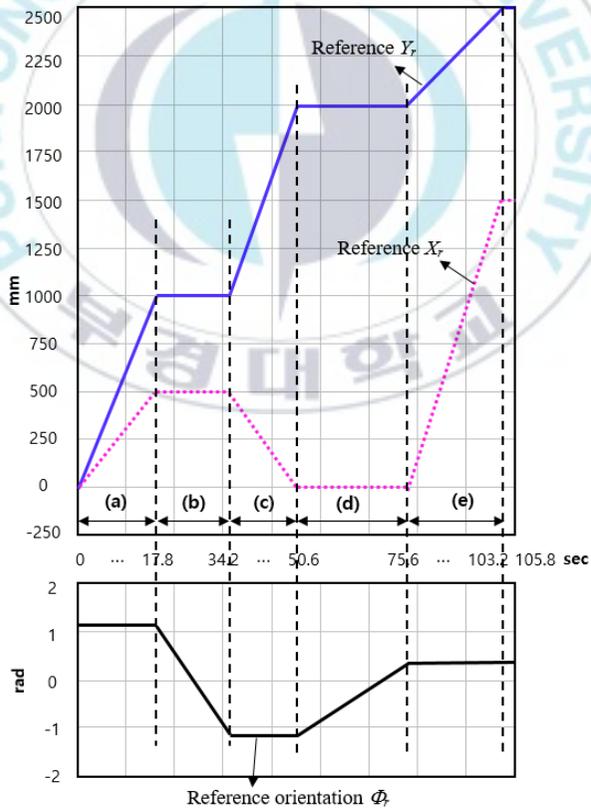


Fig. 5.1 References

Table 5.2는 Fig. 5.1의 구간별 목표값의 전체경로를 나타낸 것이다.

Table 5.2 Reference path in time domain

Parameter	Description
(a)	Path from start point to target point 1, $0 \leq t < 17.8$
(b)	Stationary path at target point 1(rotation $\phi_c = -1.1\text{rad}$ ), $17.8 \leq t < 34.2$
(c)	Path from target point 1 to target point 2, $34.2 \leq t < 50.6$
(d)	Stationary path at target point 2(rotation $\phi_c = 0.321\text{rad}$ ), $50.6 \leq t < 75.6$
(e)	Path from target point 2 to target point 3, $75.6 \leq t < 103.2$

제안된 제어기의 유효성을 검증하기 위해 시뮬레이션을 실시한다. Fig. 5.2는 시뮬레이션 및 실험에 사용되는 다중목표점 경로이며 3.3절의 Fig. 3.11과 같다. 시뮬레이션에 사용된 샘플링 시간은 0.1초이다.

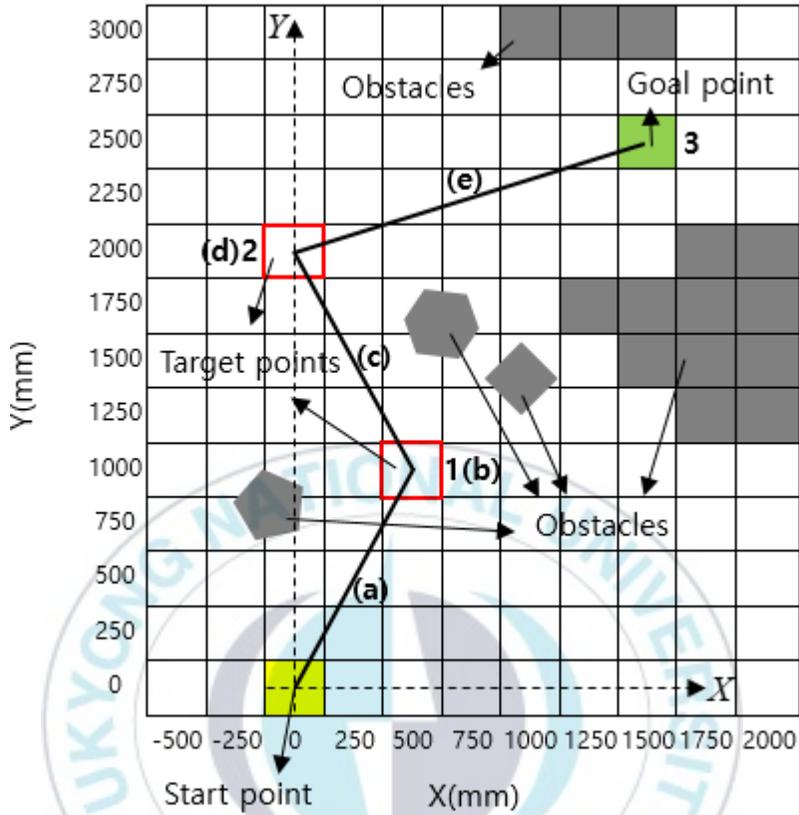


Fig. 5.2 Multiple target point path generated by the proposed multiple target point path planning method

Fig. 5.3~Fig. 5.8은 시간에 따른 OAGV의 중심위치벡터값 ( $X_c, Y_c, \phi_c$ )의 추적하는 시뮬레이션 결과를 나타낸다. Fig. 5.3에서는 X 축에서 시간에 따라 목표값  $X_r$ 을 추적하는 OAGV의 X 방향의 현재중심위치  $X_c$ 를 나타낸다. Fig. 5.4는 Fig. 5.3의 17.8~34.2초(b)와 50.6~75.6초구간(d)을 확대한 것이다. Fig. 5.4(a)의 (b)구간에서 OAGV의 현재위치  $X_c$ 는 약 499.982mm~500.055mm의 범위에서 목표값 500mm 주위를 진동하며 추적

한다. Fig. 5.4(b)의 (d)구간에서 OAGV의 현재위치  $x_c$ 는 약  $-0.059\text{mm} \sim 0.02\text{mm}$ 의 범위에서 목표값  $0\text{mm}$  주위를 진동하며 추적한다. Fig. 5.5는  $Y$  축에서 시간에 따라 목표값  $Y_r$ 을 추적하는 OAGV의 현재위치  $Y_c$ 를 나타낸다. Fig. 5.6은 Fig. 5.5의 17.8~34.2초(b)와 50.6~75.6초구간(d)을 확대한 것이다. Fig. 5.6(a)의 (b)구간에서 OAGV의 현재위치  $Y_c$ 는 약  $999.963\text{mm} \sim 1000.111\text{mm}$ 의 범위에서 목표값  $1000\text{mm}$  주위를 진동하며 추적한다. Fig. 5.6(b)의 (d)구간에서 OAGV의 현재위치  $Y_c$ 는 약  $1999.961\text{mm} \sim 2000.012\text{mm}$ 의 범위에서 목표값  $2000\text{mm}$  주위를 진동하며 추적한다. Fig. 5.7에서는 선정된 목표방향각  $\phi_r$  값을 추적하는  $\phi_c$ 을 나타낸 것이다. (a)와 (c)구간에서는 각각 목표방향각  $\pm 63^\circ$ 를 잘 추적하고 있음을 보여준다. Fig. 5.8은 Fig. 5.7의 34초 부근과 76초 부근을 확대한 것이다. Fig. 5.8(a)의 (b)구간에서 (c)구간으로 진입시 약  $-63.016^\circ$ 까지 진동 후 목표값  $-63^\circ$ 로 추적한다. Fig. 5.8(b)의 (d)구간에서 (e)구간으로 진입시  $18.413^\circ$ 까지 진동 후 목표값  $18.4^\circ$ 로 추적함을 보여준다.

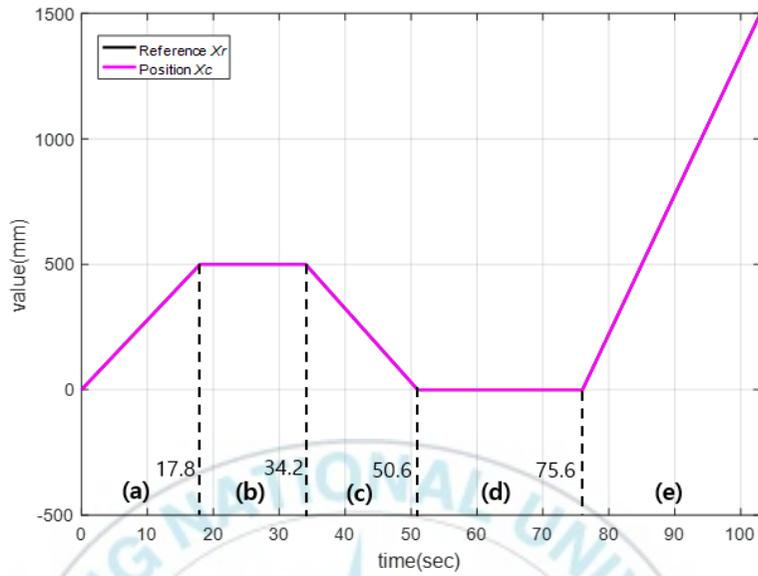
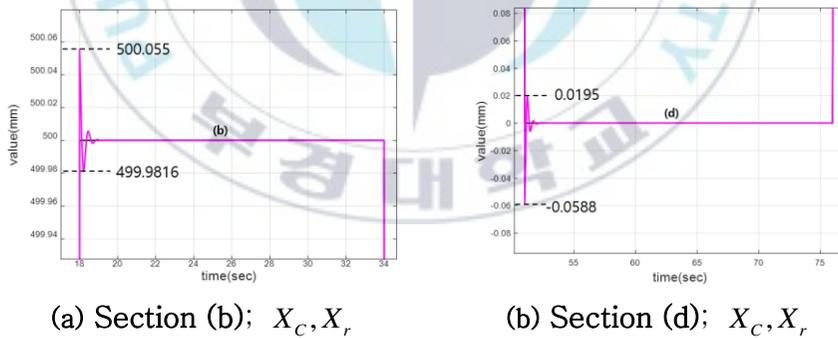


Fig. 5.3 Simulation results of OAGV's tracking reference with respect to  $X$  axis



(a) Section (b);  $X_c, X_r$

(b) Section (d);  $X_c, X_r$

Fig. 5.4 Simulation results of OAGV's tracking reference with respect to  $X$  axis for sections of (b) and (d)

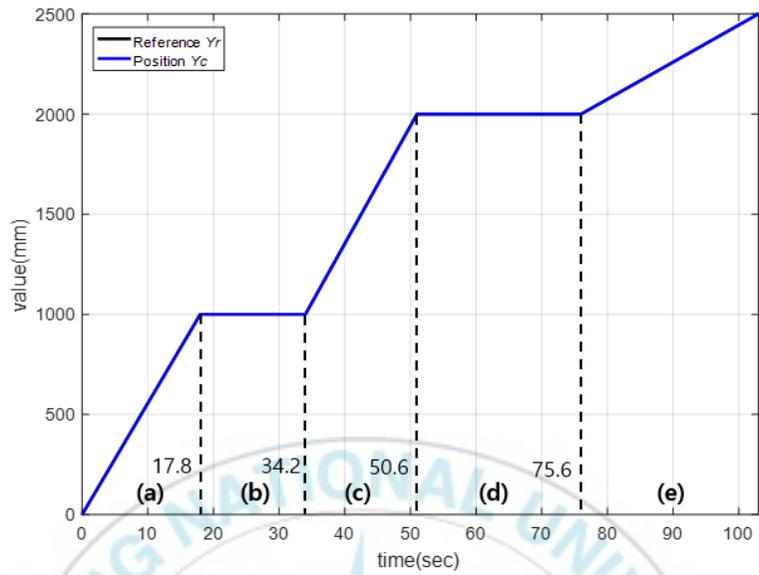
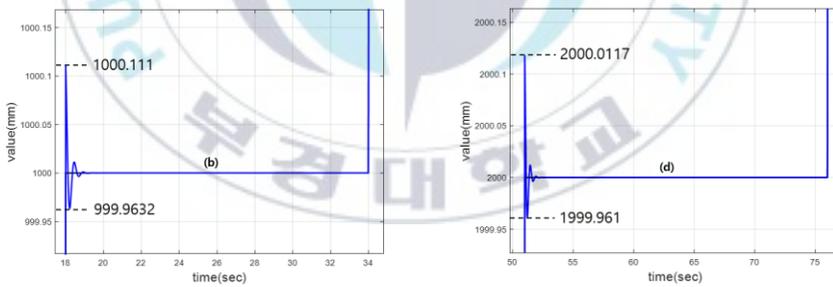


Fig. 5.5 Simulation results of OAGV's tracking reference with respect to  $Y$  axis



(a) Section (b);  $Y_c, Y_r$

(b) Section (d);  $Y_c, Y_r$

Fig. 5.6 Simulation results of OAGV's tracking reference with respect to  $Y$  axis for sections of (b) and (d)

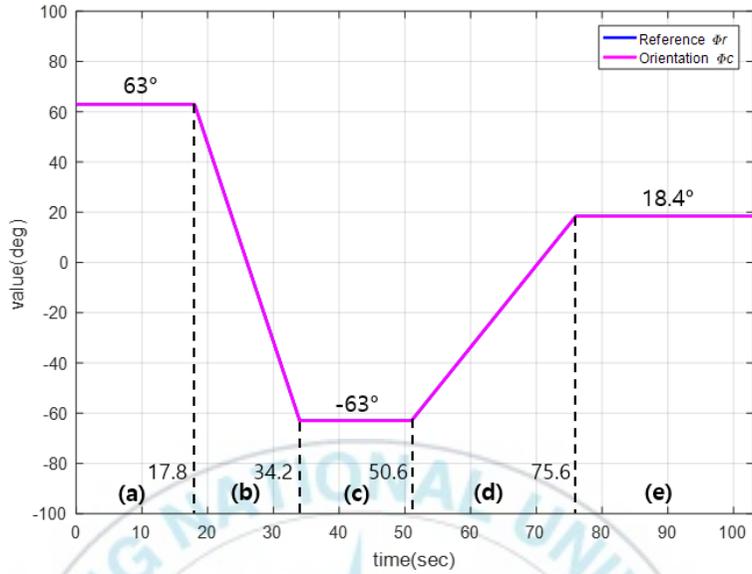


Fig. 5.7 Simulation results of OAGV's tracking reference with respect to orientation  $\Phi$



(a) Section (c);  $\Phi_c, \Phi_r$

(b) Section (e);  $\Phi_c, \Phi_r$

Fig. 5.8 Simulation results of OAGV's tracking reference with respect to orientation  $\Phi$  for sections of (c) and (e)

Fig. 5.9는 Fig. 5.2의 제안된 다중목표점 경로에서 시작점에서 최종목표지점을 향해 이동하는 OAGV의 위치 시뮬레이션

결과를 나타낸다. Fig. 5.9의 (a)구간에서는 시작점에서 목표점 1로 이동하는 구간이고 목표경로  $Y_r = 2X_r$ 에 따라서 수렴하면서 잘 추적함을 알 수 있다. (b)구간은 약 16.4초동안 제자리에서 목표방향각으로 변화한 후 다음 목표경로를 추적하는 구간이다. (c)구간에서는 목표점 1에서 목표점 2로 이동하는 구간이고 목표 경로  $Y_r = -2X_r + 2000$ 을 따라 잘 추적하고 (d)구간은 약 25초동안 제자리에서 다음 목표점으로 가기 위해 목표방향각으로 변화한 후 다음 최종목표점으로 추적하는 구간이다. (e)구간에서는 목표점 2에서 최종목표점으로 이동하는 구간이고 목표경로인  $Y_r = \frac{1}{3}X_r + 2000$ 을 따라 잘 추적함을 보여주고 있음을 알 수 있다.

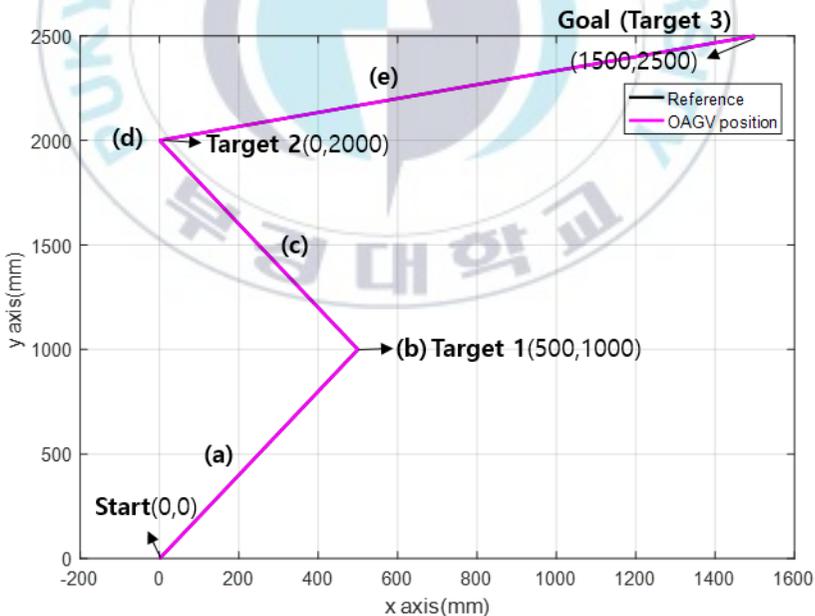


Fig. 5.9 Simulation results of OAGV's movement

Fig. 5.10는 제어척벡터  $u_1$  을 보여준다. 이는 모터 1에 인가되는 값으로 전압성분이다. 전체적으로  $-48\text{mV} \sim 52\text{mV}$  의 범위에서 진동하며 수렴한다. 시작점에서 목표점 1로 이동하는 구간인 0초에서부터 17.8초까지의 (a)구간은  $0.57\text{mV}$  를 따라 수렴하고 (b)구간에서는  $0\text{mV}$  를 따라 수렴하고 목표점 1에서 목표점 2로 이동하는 구간인 34.2초에서부터 50.6초까지의 (c)구간은  $52\text{mV}$  를 따라 수렴하고 (d)구간에서는  $0\text{mV}$  를 따라 수렴하며 목표점 2에서 최종목표점으로 이동하는 구간인 75.6초에서부터 103.2초까지의 (e)구간은  $-48\text{mV}$  를 따라 수렴함을 보여준다. Fig. 5.11은 제어척벡터  $u_2$  을 보여준다. 이는 모터 2에 인가되는 값이다. 전체적으로  $-63\text{mV} \sim 5\text{mV}$  의 범위에서 진동하며 수렴한다. 시작점에서 목표점 1로 이동하는 구간인 0초에서부터 17.8초까지의 (a)구간은  $-63\text{mV}$  를 따라 수렴하고 (b)구간에서는  $0\text{mV}$  를 따라 수렴하고 목표점 1에서 목표점 2로 이동하는 구간인 34.2초에서부터 50.6초까지의 (c)구간은  $-63\text{mV}$  를 따라 수렴하고 (d)구간에서는  $0\text{mV}$  를 따라 수렴하며 목표점 2에서 최종목표점으로 이동하는 구간인 75.6초에서부터 103.2초까지의 (e)구간은  $-19.96\text{mV}$  를 따라 수렴한다. 마찬가지로 Fig. 5.12의 제어척벡터  $u_3$  은 모터 3에 인가되는 값이다. 전체적으로  $-6\text{mV} \sim 65\text{mV}$  의 범위에서 진동하며 수렴한다. 시작점에서 목표점 1로 이동하는 구간인 0초에서부터 17.8초까지의 (a)구간은  $65\text{mV}$  를 따라 수렴하고 (b)구간에서는  $0\text{mV}$  를 따라 수렴하고 목표점 1에서 목표점 2로 이동하는 구간인 34.2초에서부터 50.6초까지의 (c)구간은  $9.491\text{mV}$  를 따라 수렴하고 (d)구간에서는  $0\text{mV}$  를 따라 수렴하며

목표점 2에서 최종목표점으로 이동하는 구간인 75.6초에서부터 103.2초까지의 (e)구간은 65mV를 따라 수렴함을 보여준다.

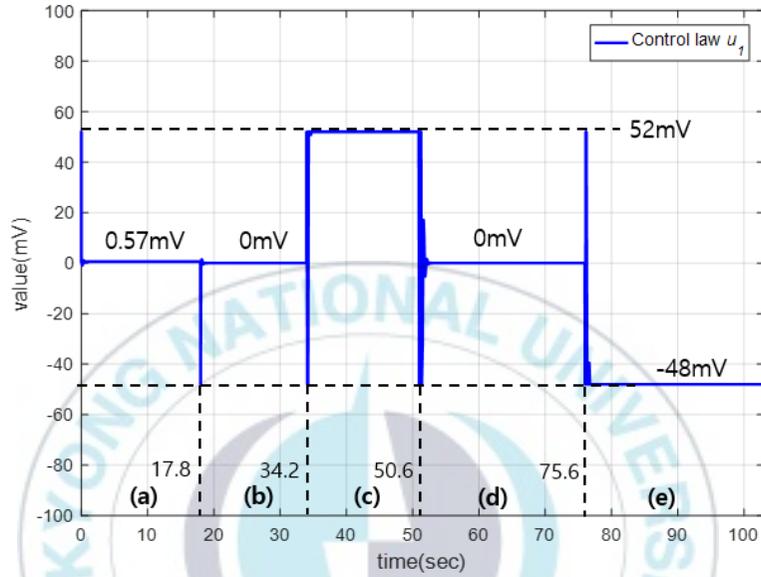


Fig. 5.10 Control law vector  $u_1$  with respect to motor 1

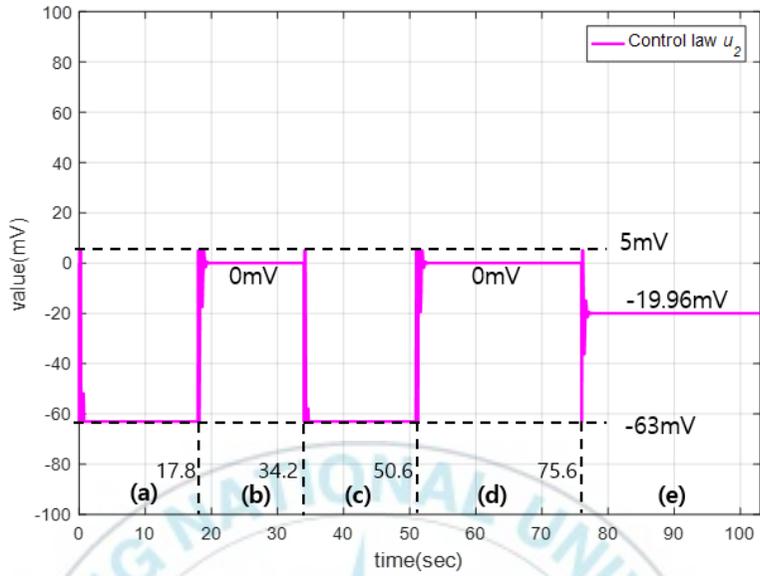


Fig. 5.11 Control law vector  $u_2$  with respect to motor 2

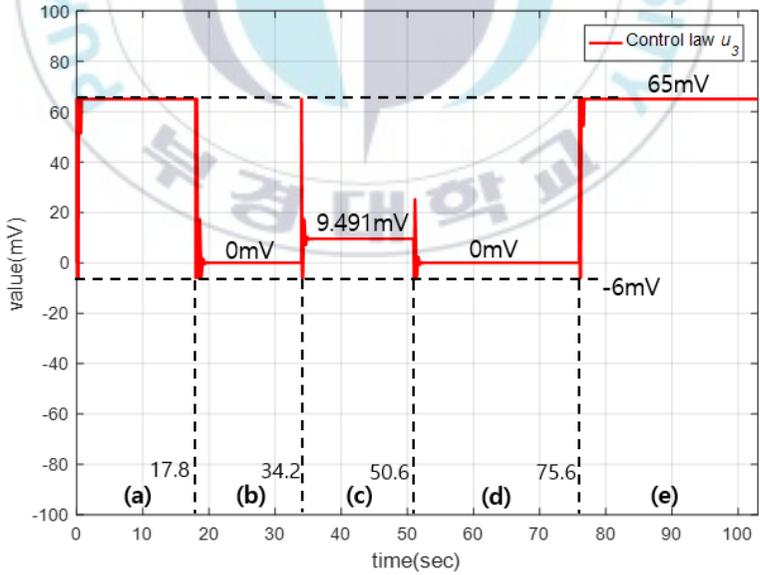


Fig. 5.12 Control law vector  $u_3$  with respect to motor 3

Fig. 5.13은 가상의 추적오차벡터인 추적위치오차벡터  $\mathbf{z}_1$  를 나타낸다. Fig. 5.13에서  $X_c, Y_c, \phi_c$ 는 각각  $X, Y, \phi$ 방향의 위치 오차성분이며, 시작지점부근에서  $X_c$ 는 약  $-0.011\text{mm} \sim 0.037\text{mm}$ 의 범위에서 진동하며 0으로 수렴하였고,  $Y_c$ 는 약  $-0.006\text{mm} \sim 0.018\text{mm}$ 의 범위에서 진동하며 0으로 수렴하였다.  $X_c, Y_c$ 방향의 최대추적위치오차는 각각 50.6초(d)구간에서 약  $0.118\text{mm}$ 이고 34.2초(c)구간에서 약  $0.059\text{mm}$ 이며,  $X_c, Y_c$ 방향의 최소추적위치오차는 각각 34.2초(c)구간에서 약  $-0.118\text{mm}$ 이고 75.6초(e)구간에서 약  $-0.111\text{mm}$ 이다. 이후 0으로 수렴한다.  $\mathbf{z}_1$ 의  $\phi_c$  성분인 방향각오차는 거의 0값을 갖음을 보여준다.

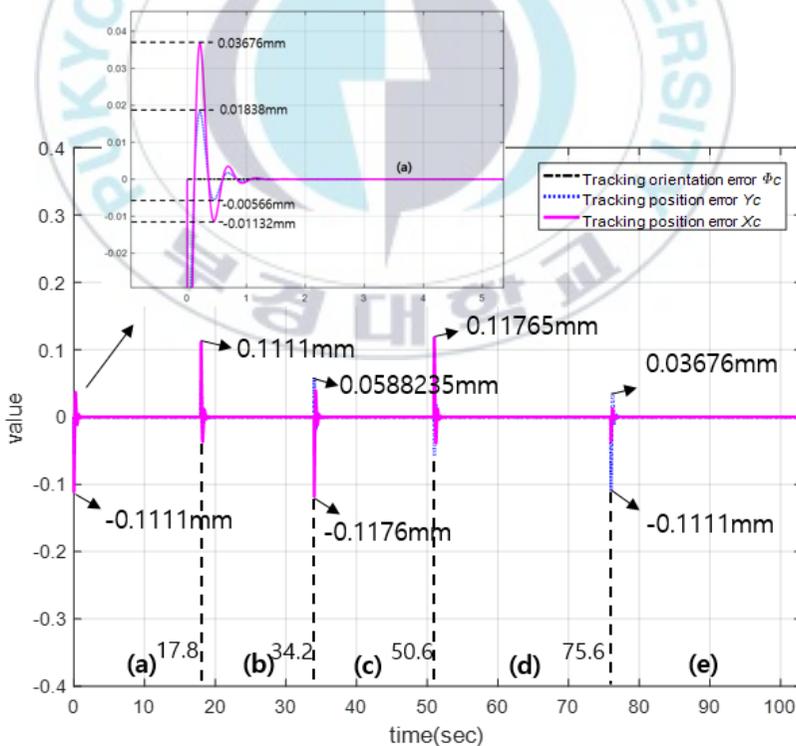


Fig. 5.13 Tracking posture error vector of OAGV  $\mathbf{z}_1$

Fig. 5.14~Fig. 5.16은  $\mathbf{x}_2$  성분으로 OAGV의 속도벡터들이다. Fig. 5.14는 OAGV의  $X$  방향의 현재속도  $\dot{X}_c$  를 나타낸다. Fig. 5.14의 (a)구간에서 OAGV의 속도는 약  $v_{0\sim 17.8\text{sec}} = 27.778\text{mm/s}$  로 나타나고 (c)구간에서  $v_{34.2\sim 50.6\text{sec}} = -29.412\text{mm/s}$  로 나타나며, (e)구간에서  $v_{75.6\sim 103.2\text{sec}} = 55.556\text{mm/s}$  로 나타난다. 회전구간인 (b)와 (d)구간은  $0\text{mm/s}$  임을 보여주고 있다.

Fig. 5.15는 OAGV의  $Y$  방향의 현재속도  $\dot{Y}_c$  를 나타낸다. Fig. 5.15의 (a)구간에서 OAGV의 속도는 약  $v_{0\sim 17.8\text{sec}} = 55.556\text{mm/s}$  로 나타나고 (c)구간에서 약  $v_{34.2\sim 50.6\text{sec}} = 58.824\text{mm/s}$  로 나타나며, (e)구간에서  $v_{75.6\sim 103.2\text{sec}} = 18.52\text{mm/s}$  로 나타난다. 회전구간인 (b)와 (d)구간은  $0\text{mm/s}$  임을 보여주고 있다. Fig. 5.14와 Fig. 5.15의 0초에서부터 17.8초구간인 (a)구간에서  $\dot{Y}_c$  값이  $\dot{X}_c$  값보다 2배 정도 높게 측정되는 것으로 볼 때, 목표경로  $Y_r = 2X_r$  을 따라 Fig. 5.2에서 보여지는 바와 같이  $Y$  축으로 2배만큼 많이 이동함을 알 수 있다.

Fig. 5.16은 OAGV의 회전각속도  $\omega_c$  이다. Fig. 5.15의 직선구간인 (a)구간에서 OAGV의 각속도는  $\omega_{0\sim 17.8\text{sec}} = -0.0075 \times 10^{-14} \text{deg/s}$  로 나타나고 (c)구간에서  $\omega_{34.2\sim 50.6\text{sec}} = -0.001 \times 10^{-14} \text{deg/s}$  로 나타나며, (e)구간에서  $\omega_{75.6\sim 103.2\text{sec}} = -0.006 \times 10^{-14} \text{deg/s}$  로 나타나나 거의  $0\text{deg/s}$  를 나타냄을 알 수 있다. 회전구간인 (b)와 (d)구간은  $0\text{deg/s}$  이다. 경로변경지점인 (a)구간의 시작점에서는  $-0.573 \times 10^{-14} \text{deg/s}$ , (b)구간의 시작점에서는  $1.147 \times 10^{-14} \text{deg/s}$ , (c)구간의 시작점에서는 약  $-0.159 \times 10^{-14} \text{deg/s}$ , (d)구간의 시작점에서는  $0.049 \times 10^{-14} \text{deg/s}$ , (e)구간의 시작점에서는  $-0.628 \times 10^{-14} \text{deg/s}$  로 나타내고 있음을 보여준다.

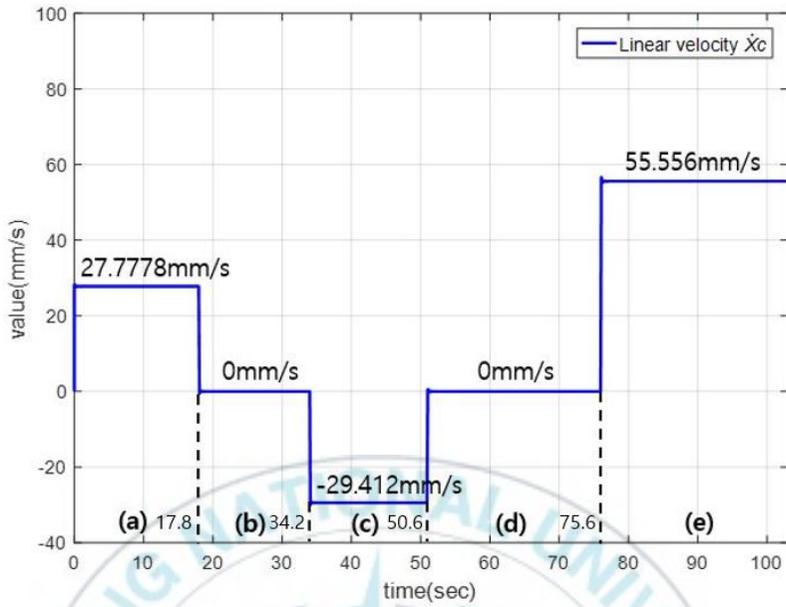


Fig. 5.14 Tracking linear velocity vector  $\dot{X}_c$  of OAGV  $x_2$

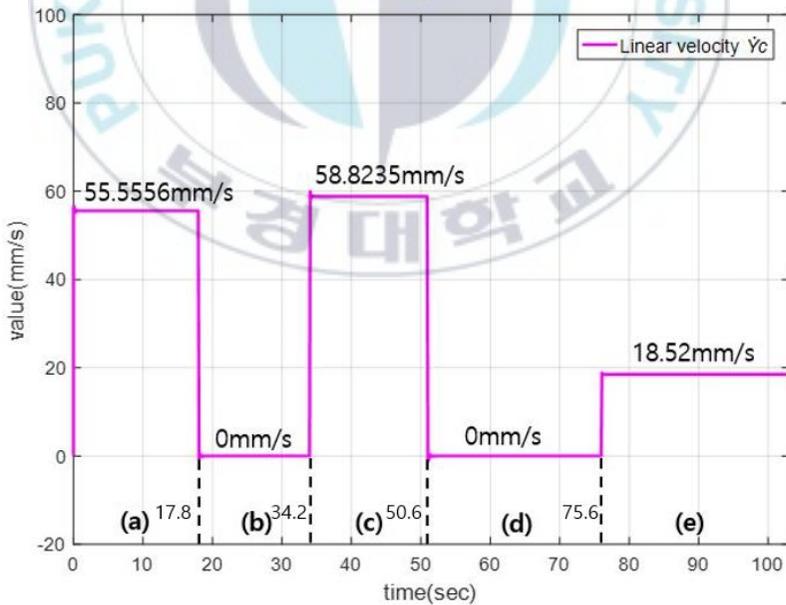


Fig. 5.15 Tracking linear velocity vector  $\dot{Y}_c$  of OAGV  $x_2$

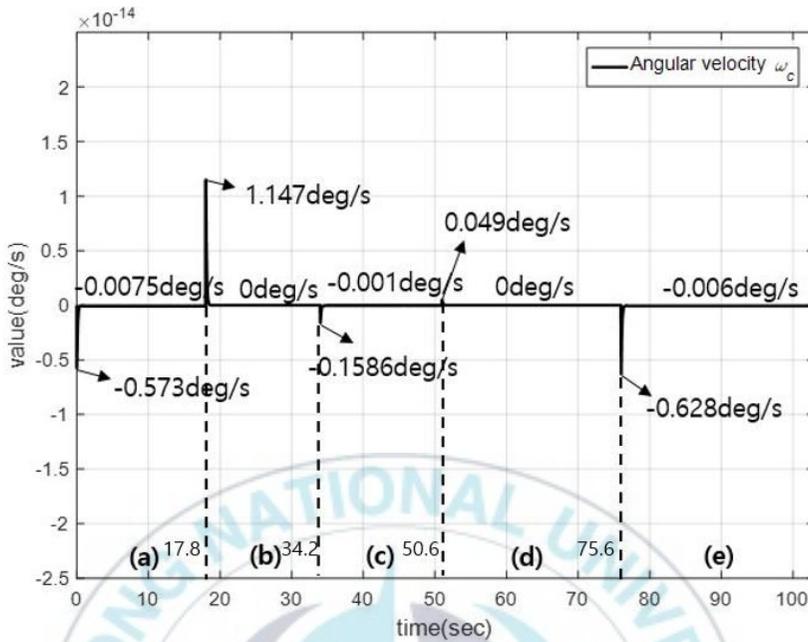


Fig. 5.16 Tracking angular velocity vector  $\omega_c$  of OAGV  $x_2$

Fig. 5.17는 가상의 추적속도오차벡터  $z_2$  를 보여준다. Fig. 5.17의  $\dot{X}_c, \dot{Y}_c, \omega_c$  는 각각  $X, Y, \phi$  방향의 속도오차성분이며, 시작지점부근에서  $\dot{X}_c$  는  $-0.31\text{mm/s} \sim 1.008\text{mm/s}$  의 범위에서 진동하며 0으로 수렴하였고  $\dot{Y}_c$  는  $-0.155\text{mm/s} \sim 0.505\text{mm/s}$  의 범위에서 진동하며 0으로 수렴하였다.  $\dot{X}_c$  의 최대추적속도오차는 50.6초 (d)구간에서  $60.12\text{mm/s}$  이고,  $\dot{X}_c$  의 최소추적속도오차는 34.2초 (c)구간에서  $-60.12\text{mm/s}$  이다.  $\dot{Y}_c$  의 최대추적속도오차는 34.2초 (c)구간에서  $30.06\text{mm/s}$  이고,  $\dot{Y}_c$  의 최소추적속도오차는 75.6초 (e)구간에서  $-56.78\text{mm/s}$  이다. 이후 0으로 수렴한다.  $\omega_c$  성분인  $z_2$  의 각속도 오차는 전구간에서 거의 0값을 가진다. 이를 통해

각 경로변경구간에서 속도오차성분이 크게 되는 현상을 확인가능하다.

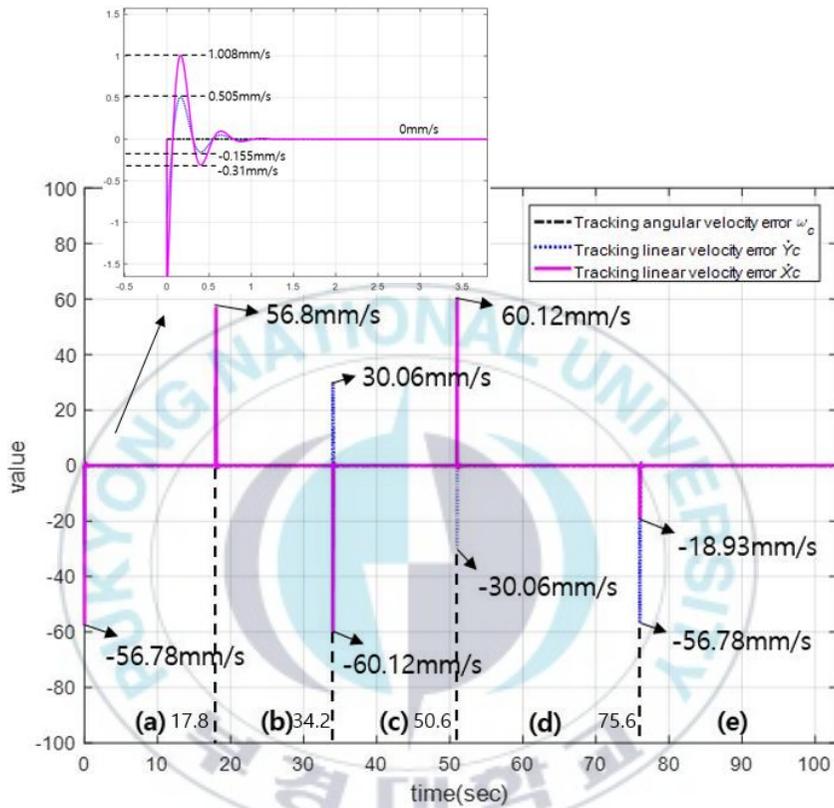


Fig. 5.17 Tracking velocity error vector of OAGV  $z_2$

## 5.2 실험 결과

실험 환경은 바닥에 보이는 1칸의 가로, 세로는 약 400mm이다. 실험에서의 샘플링 시간은 OAGV의 빠른 처리속도를 위해 0.2초로 설정하였다.

Fig. 5.18에서 시작점은 (0,0)mm,  $\phi_{c0} = 63^\circ$ 로 설정되었으며, OAGV가 각각 시작점에서 목표점 1, 2, 3까지 이동하는 과정을 나타내는 실험결과들이다. Fig. 5.18(a)에서 OAGV가 시작점 (0,0)mm에서 목표점 1 (500,1000)mm까지 이동한 구간(a)를 나타내며, 이때 실제이동거리는 약 1120mm이다. Fig. 5.18(b)는 목표점 1 제자리에서 다음 목표방향각  $\phi_r = -63^\circ$ 으로 회전하는 구간(b)이다. Fig. 5.18(c)는 OAGV가 목표점 1 (500,1000)mm에서 목표점 2 (0,2000)mm까지 이동한 구간(c)를 나타내며, 이때 목표점 1 (500,1000)mm에서 목표점 2 (0,2000)mm까지 실제이동거리는 약 1120mm이다. Fig. 5.18(d)는 목표점 2 제자리에서 다음 목표방향각  $\phi_r = 18^\circ$ 으로 회전하는 구간(d)이다. Fig. 5.18(e)는 OAGV가 목표점 2 (0,2000)mm에서 최종목표점 (1500,2500)mm까지 이동한 구간(e)를 나타내며, 이때 목표점 2 (0,2000)mm에서 최종목표점 (1500,2500)mm까지 실제이동거리는 약 1580mm이다.



(a) From start point to target point 1



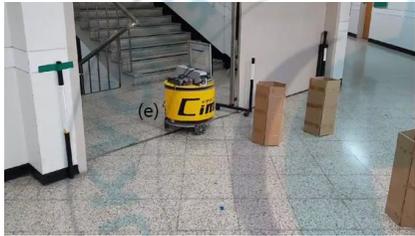
(b) At target point 1



(c) From target point 1 to target point 2



(d) At target point 2



(e) From target point 2 to goal point(target point 3)

Fig. 5.18 Experimental results of OAGV's movement

Fig. 5.19~Fig. 5.21은 Fig. 5.1의 목표값을 추적하는 OAGV의 실험결과인 추적위치값  $X_c, Y_c, \phi_c$  을 시뮬레이션 결과와 비교하여 각 구간별로 도시화한 것이다. Fig. 5.19에서 실험결과 추적위치값  $X_c$  는 시작지점부근 1.8초에서 시뮬레이션 값과 48.914mm의 차이를 보이다가 진동하며 8초에서 시뮬레이션 값에 수렴하였다.  $X_c$  의 최대추적위치는 약 104초에서 약 1584.359mm로 발생하였고 목표값 1500mm로 진동하며 수렴하였으며,  $X_c$  의 최소추적위

치는 (c)구간에서  $-45.08\text{mm}$  로 발생하였고 목표값  $0\text{mm}$  로 점차 진동하며 수렴하였다. Fig. 5.20에서 실험결과 추적위치값  $Y_c$  의 최대추적위치는 약 104초에서  $2512.38\text{mm}$  로 나타났고 목표값  $2500\text{mm}$  로 진동하며 수렴하였으며,  $Y_c$  의 최소추적위치는 시작지점부근 (a)구간 이동시 약  $-0.576\text{mm}$  로 발생하여 시뮬레이션 값으로 진동하며 수렴하였다. Fig. 5.21에서 추적방향각  $\phi_c$  의 최대추적위치는 (a)구간에서 약  $65.236^\circ$  로 발생하였고 시뮬레이션 값  $63^\circ$  로 진동하며 수렴하였으며,  $\phi_c$  의 최소추적위치는 (c)구간에서  $-65.55^\circ$  로 발생했고 점차 시뮬레이션 값  $-63^\circ$  를 따라 진동하며 수렴하였다.

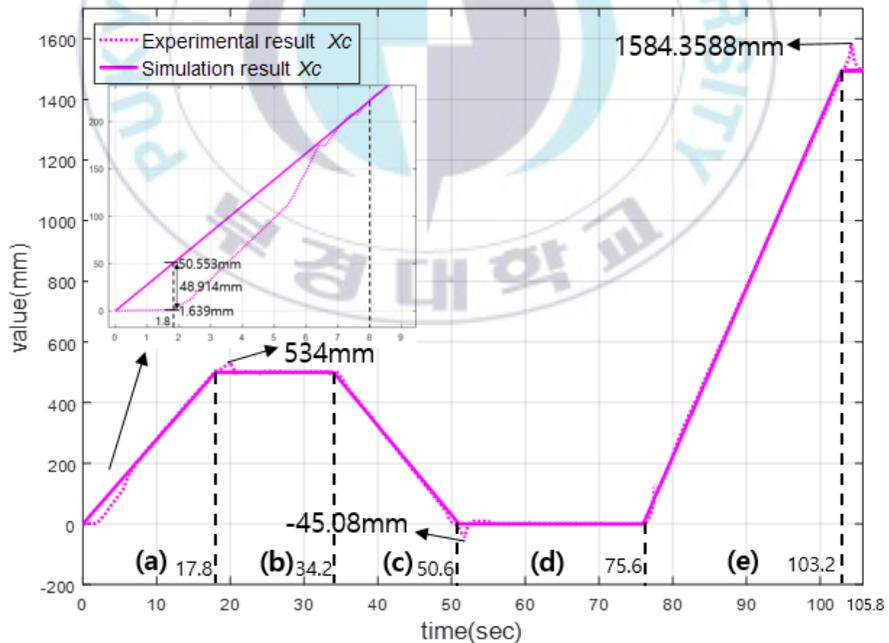


Fig. 5.19 Experimental results of OAGV's tracking position with respect to  $X$  axis

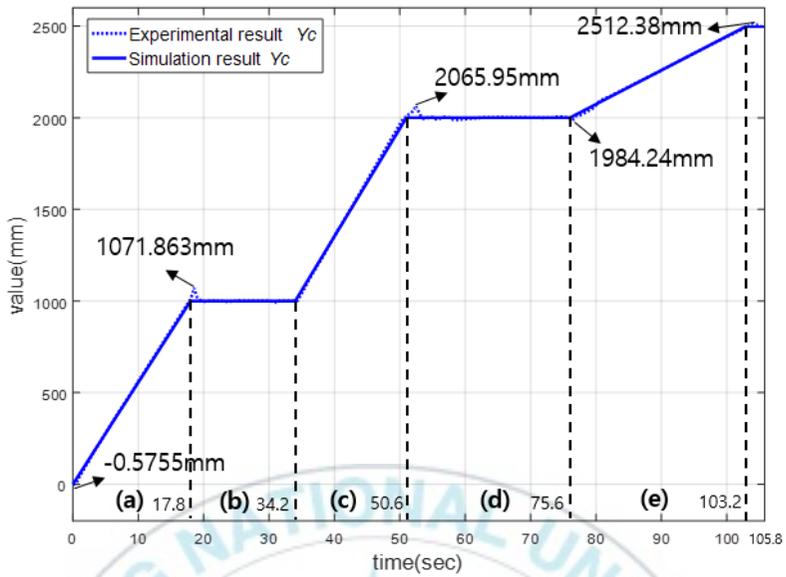


Fig. 5.20 Experimental results of OAGV's tracking position with respect to  $Y$  axis

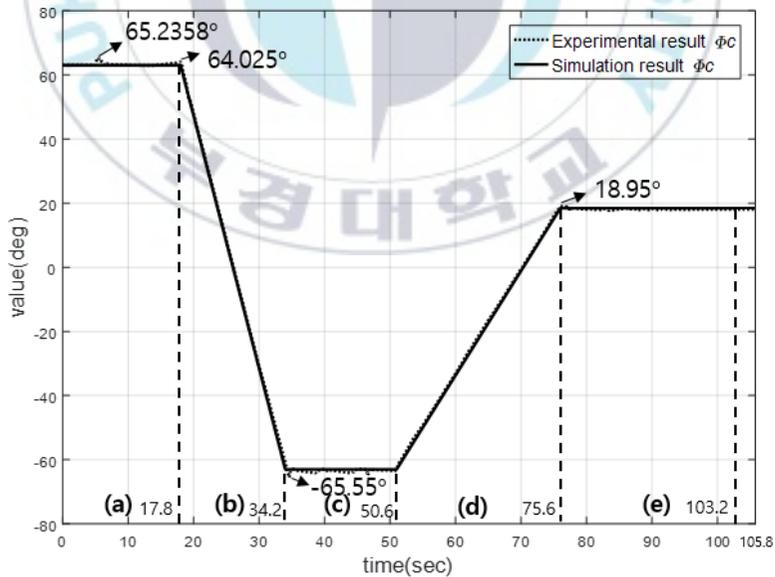


Fig. 5.21 Experimental results of OAGV's tracking position with respect to orientation  $\phi$

Fig. 5.22는 추적위치오차벡터  $z_1$  을 시뮬레이션 결과와 비교하여 각 구간별로 도시화한 것이다.  $X_c$  의 최대추적위치오차는 약 104초에서 84.36mm 로 나타나고,  $-6.154\text{mm} \sim 84.36\text{mm}$  의 범위에서 크게 진동하며 0mm 로 수렴하는 모습을 보이며, 최소추적위치오차는 3.6초(a)구간에서  $-54.31\text{mm}$  로 나타나며 점차 0mm 로 수렴한다.  $Y_c$  의 최대 및 최소추적위치오차는 각각 18초(b) 구간에서 약  $71.863\text{mm}$  로 나타나고, 75.6초(e)구간에서  $-30.25\text{mm}$  mm 로 나타나며 점차 0mm 로 수렴한다.  $\phi_c$  의 최대추적위치오차는 (b)구간인 17.8초에서  $3.046^\circ$  로 나타나고, 최소추적위치오차는 (c)구간인 34.2초에서  $-2.555^\circ$  로 나타나며 점차  $0^\circ$  로 수렴한다.

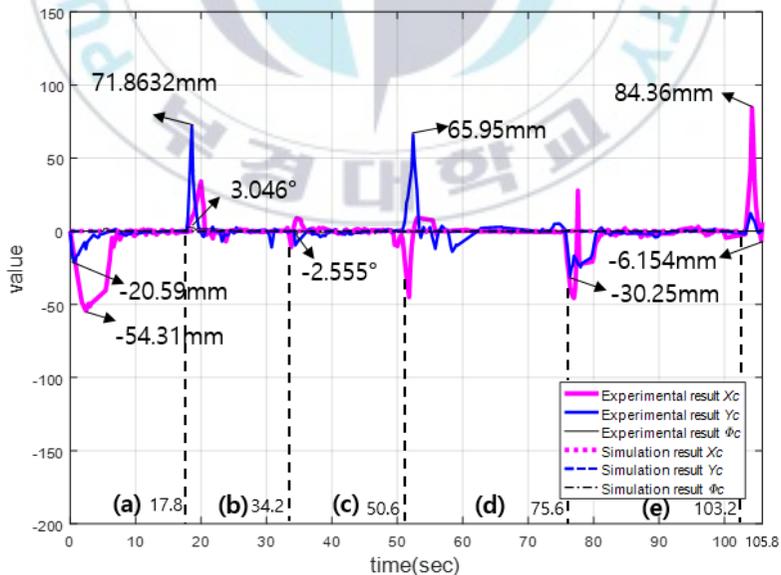


Fig. 5.22 Experimental results of OAGV's tracking posture error vector of OAGV  $z_1$

Fig. 5.23~Fig. 5.25는 OAGV의 추적속도값  $\dot{X}_c$ ,  $\dot{Y}_c$ ,  $\omega_c$  을 시뮬레이션 결과와 비교하여 각 구간별로 도시화한 것이다. Fig. 5.23에서 추적속도값  $\dot{X}_c$  는 (d)구간에서 (e)구간으로 진입시  $-197.2\text{mm/s} \sim 282\text{mm/s}$  의 범위에서 최대 및 최소값을 가지며, 직선구간인 (a), (c), (e)구간에서 각각 약  $27.778\text{mm/s}$ ,  $-29.412\text{mm/s}$ ,  $55.556\text{mm/s}$  으로 시뮬레이션 결과값을 잘 추종한다. 회전구간인 (b), (d)구간에서는  $0\text{mm/s}$ 로 진동하며 잘 추종한다. Fig. 5.24에서 추적속도값  $\dot{Y}_c$  는 (a)구간에서 (b)구간으로 진입시  $-179.22\text{mm/s}$ 로 최소값을 나타내며  $-179.22\text{mm/s} \sim 110\text{mm/s}$  의 범위에서 진동하며  $0\text{mm/s}$ 로 수렴한다. (c)구간에서 (d)구간으로 진입시 최대값  $115.72\text{mm/s}$  을 나타내며  $-81.83\text{mm/s} \sim 115.72\text{mm/s}$  의 범위에서 진동하며  $0\text{mm/s}$ 로 수렴한다. 직선구간인 (a), (c), (e)구간에서 각각 약  $55.556\text{mm/s}$ , 약  $58.824\text{mm/s}$ ,  $18.52\text{mm/s}$  으로 시뮬레이션 결과값을 잘 추종한다. Fig. 5.25에서 추적각속도값  $\omega_c$  는 약 5초(a)구간에서 최대값  $9.061\text{deg/s}$  를 나타내고 약 19초(b)구간에서 최소값  $-14.025\text{deg/s}$  를 나타낸다. 또한 직선구간인 (a), (c), (e)에서 각각  $-0.0075\text{deg/s}$ ,  $-0.001\text{deg/s}$ ,  $-0.006\text{deg/s}$  으로 시뮬레이션 결과값을 잘 추종한다. 회전구간인 (b), (d)구간에서는 시뮬레이션 결과값과 비교하여 각각  $7.87\text{deg/s}$ ,  $3.26\text{deg/s}$  의 오차를 가진다.

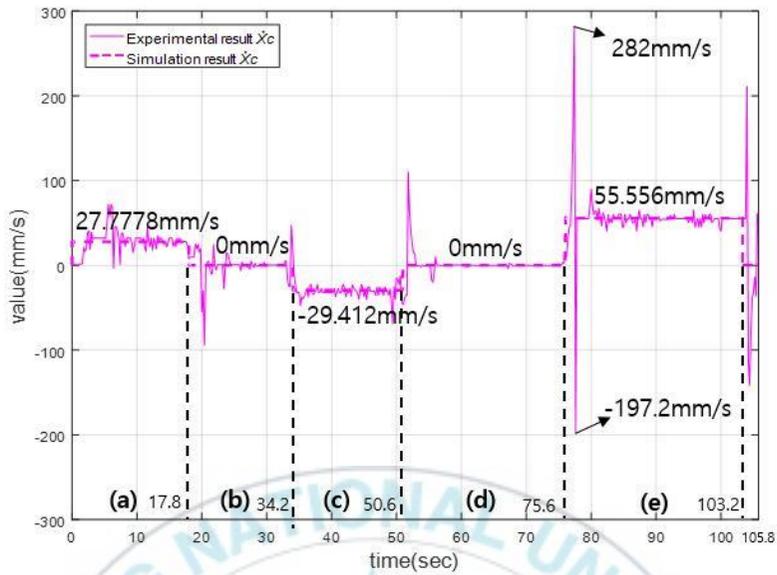


Fig. 5.23 Experimental results of OAGV's tracking linear velocity with respect to  $X$  axis

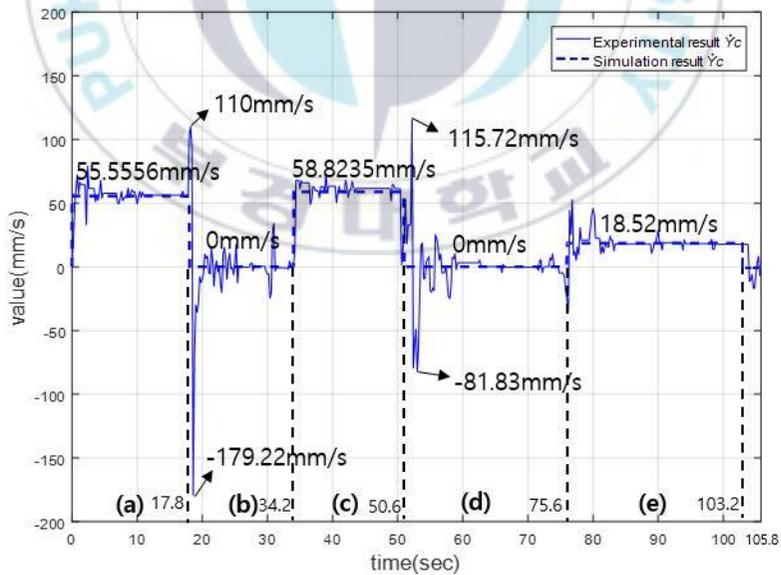


Fig. 5.24 Experimental results of OAGV's tracking linear velocity with respect to  $Y$  axis

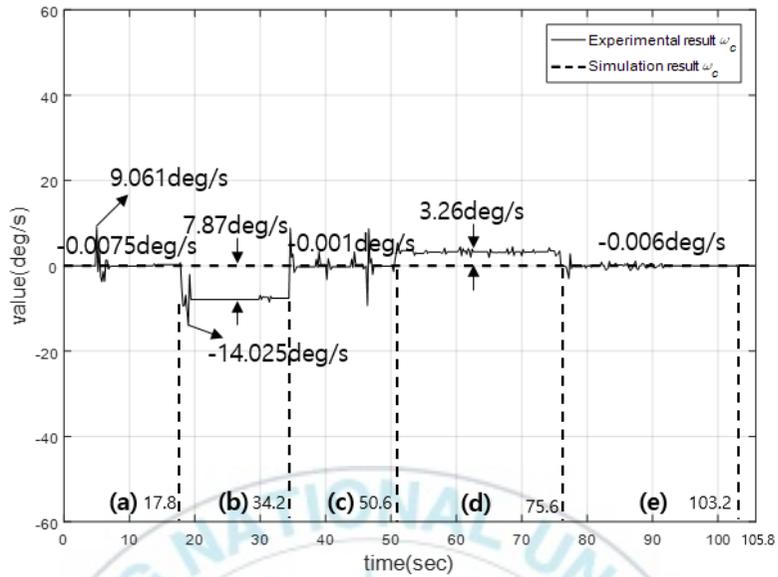


Fig. 5.25 Experimental results of OAGV's tracking angular velocity with respect to orientation  $\phi$

Fig. 5.26은 추적속도오차벡터  $\mathbf{z}_2$  를 시뮬레이션 결과와 비교하여 각 구간별로 도시화한 것이다.  $\dot{X}_c$  의 최대추적속도오차는 약 104초에서 337.54mm/s 로 나타나고  $-32.94\text{mm/s} \sim 337.54\text{mm/s}$  의 범위에서 크게 진동하며 0mm/s 으로 수렴한다. 최소추적속도오차는 (a)구간의 약 2.2초에서  $-166.167\text{mm/s}$  로 나타나고 점차 0mm/s 으로 수렴한다.  $\dot{Y}_c$  의 최대추적속도오차는 (b)구간의 17.8 초에서 약 253.149mm/s 이고, 최소추적속도오차는 (e)구간의 75.6 초에서  $-111.62\text{mm/s}$  로 나타난다. 특히, 각 경로변경 구간에서 크게 진동하며 0mm/s 으로 점차 수렴한다.  $\omega_c$  의 최대추적각속도 오차는 (b)구간의 17.8초에서 약 11.055deg/s 이고, 최소추적각속

도오차는 (c)구간의 34.2초에서  $-10.705\text{deg/s}$  으로 나타났으며 전체적으로  $0\text{deg/s}$  로 잘 수렴한다. 이를 통해 실험결과에서도 시뮬레이션과 유사하게 주로 구간을 이동하는 지점에서 진동이 크게 발생함을 알 수 있다.

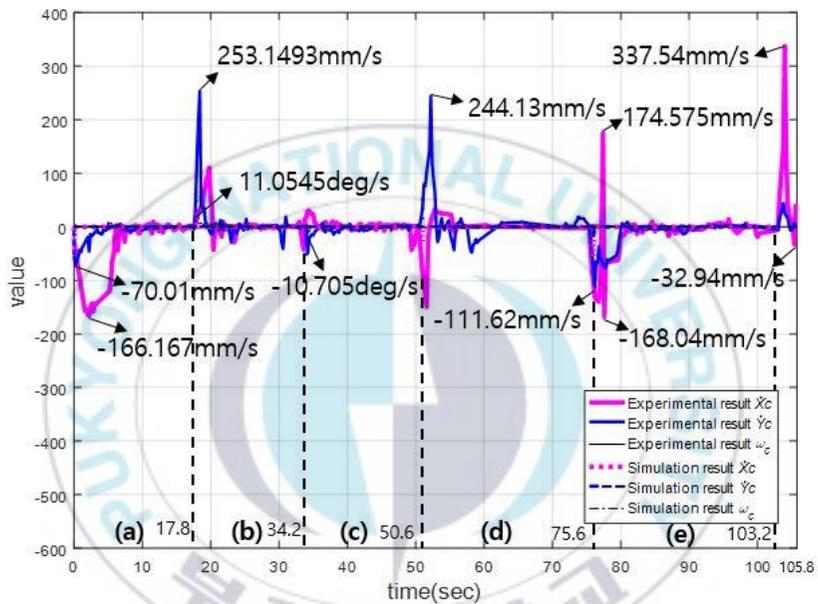


Fig. 5.26 Experimental results of OAGV's tracking velocity error vector of OAGV  $z_2$

# 제 6 장 결 론

## 6.1 결론

본 연구에서는 OAGV의 안전한 주행을 위한 다중목표점 경로 계획법을 제안하고, 제안된 경로계획법에 의해 계획된 경로를 추적하기 위한 시스템 구성 및 제어를 설계하였다. 또한, 주어진 간단한 장애물이 있는 영역에 대하여 다중목표점 경로계획법 및 제어를 구현하기 위해서 OAGV를 이용하여 시뮬레이션 및 실험을 통해 그 유효성을 검증하였다. 본 연구의 결론을 요약하면 다음과 같다.

- ◆ 제 2장에서는 제 3장에서 생성되는 다중목표점 경로를 추적하는 실험을 수행하기 위한 OAGV의 시스템을 구성하였다. OAGV는 기구시스템과 제어시스템으로 구성되어 있으며, 제어시스템은 센서부, 제어부, 구동부 및 전원공급장치로 구성된다. OAGV는 반지름 0.275m, 높이 0.53m로 구성된 전방향 구동 시스템을 사용하였고, OAGV의 중심에서 0.275m거리만큼 떨어진 위치에 120° 각도로 이루어진 더블 유니버설

전방향 구동바퀴(반지름 0.0625m)를 사용하였으며 이를 각각 DC서보모터로 연결되었다. OAGV 위치인식과 장애물 인식을 위해 NAV-200 레이저센서는 OAGV의 중앙에 위치하고 있고, 전원 공급장치로 25.2V의 리튬이온배터리 1개를 사용하였다. 후면부에는 7인치 모니터와 컨트롤 패널이 위치하고 있으며, OAGV 내부에 산업용 PC ARK-5260을 이용하여 제어부를 구성하였다.

- ◆ 제 3장에서는 기지의 환경에서 장애물의 날카로운 모서리를 지나지 않고 안전한 주행을 하기 위한 다중목표점 경로계획 알고리즘을 제시하였다. 주어진 환경의 영역을 정사각형 셀로 나누어 셀의 중심을 노드로 정의하며 노드값들의 거리비용계산을 통하여 장애물을 회피하고 시작점에서 목표점까지 최단경로로 이동하는 D\* Lite 알고리즘을 제시하였고, 이를 이용하여 추가적으로 해밀턴 경로법을 이용한 다중목표점 경로계획 알고리즘을 제시하였다. A\* 알고리즘과 D\* Lite 알고리즘과의 시뮬레이션 비교를 통해 다중목표점 경로계획 알고리즘이 다른 알고리즘들에 비해 경로는 길지만, 장애물의 날카로운 모서리를 지나지 않는 안전성을 증명하였으며, 제 5장의 시뮬레이션 및 실험에 사용된 전체 목표점 경유 경로를 생성하였다.

- ◆ 제 4장에서는 OAGV의 시스템 모델링 및 백스텝핑을 이용한 경로 추적제어기를 제시하였다. 먼저 OAGV의 기구학적 모델링을 제시하였고 DC서보모터로부터 생성된 바퀴에 발생하는 힘 및 마찰이나 미끄러짐으로 생기는 힘외란벡터를 가지는 OAGV의 동역학적 모델링을 제시하였다. 이 방정식을 이용하여 OAGV가 목표 경로를 추적하기 위한 백스텝핑 비선형 경로 추적제어기를 제시하였다. 또한, Lyapunov 함수를 이용하여 시스템의 안정성을 판별하였다.
- ◆ 제 5장에서는 제 3장에서 생성되는 전체 목표점 경유 경로를 추적하는 제 4장에서 제시된 경로 추적제어기의 유효성을 검증하기 위해 OAGV 시뮬레이션 및 실험결과를 제시하였다. OAGV의 목표위치값  $X_r$  은 (a), (b)구간에서 500mm 이고 (c), (d)구간은 0mm 이며, (e)구간은 1500mm 이다. 목표위치값  $Y_r$  은 (a), (b)구간에서 1000mm 이고 (c), (d)구간은 2000mm 이며, (e)구간은 2500mm 이다. 목표위치값  $\phi_r$  은 (a) 구간에서  $63^\circ$  이고 (b), (c)구간은  $-63^\circ$  이며, (d), (e)구간은  $18.4^\circ$  이다. 이에 대한 시뮬레이션 결과 최대 추적위치오차 벡터는  $\mathbf{z}_{1,\max} = [0.118\text{mm} \ 0.059\text{mm} \ 0^\circ]^T$  를 나타냈으며, 최소 추적위치오차벡터는  $\mathbf{z}_{1,\min} = [-0.118\text{mm} \ -0.111\text{mm} \ 0^\circ]^T$  를 나타냈다. 여기서, 추적위치오차벡터의  $\phi_c$  값은 전구간에서

0deg/s 로 안정적이게 수렴하여 목표값을 잘 추종했다. 그리고 최대 추적속도오차벡터는  $\mathbf{z}_{2,\max} = [60.12\text{mm/s} \ 30.06\text{mm/s} \ 0\text{deg/s}]^T$  를 나타냈으며, 최소 추적속도오차벡터는  $\mathbf{z}_{2,\min} = [-60.12\text{mm/s} \ -56.78\text{mm/s} \ 0\text{deg/s}]^T$  를 나타냈다. 추적오차벡터들을 통해 OAGV가 계획된 경로를 잘 추적하고 있음을 확인했다. 특히, 시뮬레이션 결과값이 목표값과 비교해 주로 구간변경지점에서 추적위치오차벡터와 추적속도오차벡터가 크게 발생하였지만, 해당 경로를 지난 후 오차는 0으로 점차 수렴함을 알 수 있었다.

- ◆ 실험결과에서 OAGV의 최대 추적위치오차벡터는  $\mathbf{z}_{1,\max} = [84.36\text{mm} \ 71.863\text{mm} \ 3.046^\circ]^T$  를 나타냈으며, 최소 추적위치오차벡터는  $\mathbf{z}_{1,\min} = [-54.31\text{mm} \ -30.25\text{mm} \ -2.555^\circ]^T$  를 나타냈다. 그리고 최대 추적속도오차벡터는  $\mathbf{z}_{2,\max} = [337.54\text{mm/s} \ 253.149\text{mm/s} \ 11.055\text{deg/s}]^T$  를 나타냈으며, 최소 추적속도오차벡터는  $\mathbf{z}_{2,\min} = [-168.04\text{mm/s} \ -111.62\text{mm/s} \ -10.705\text{deg/s}]^T$  를 나타냈다. 실험결과에서 구간변경지점에서 시뮬레이션과 비교적 큰 오차를 나타내었다. 그러나 곧 제어입력으로 인해 다시 시뮬레이션 결과값을 따라 진동하며 잘 추적하였다.

그러므로 본 연구에서 연구된 다중목표점 경로계획 알고리즘과 경로 추적제어기는 기지의 환경에서 전방향구동을 하는 AGV들의 개발 시 유용한 참조학문으로 적용될 수 있을 것이다.



## 6.2 향후 연구

본 연구에 대한 향후 계획은 다음과 같다.

- ◆ SLAM을 이용하여 미지의 환경에서 경로계획 연구
- ◆ 각종 외란에 대한 고장진단에 대한 연구
- ◆ 전방향 구동가능한 메카넘휠에 대한 연구
- ◆ AGV에 카메라 장착하여 실시간 환경 촬영가능 로봇 설계



## 감사의 글

이 논문이 완성되기까지 수많은 조언과 격려로 지도하여 주신 김상봉 교수님께 감사드리며, 항상 연구실 학생들을 자식같이 아껴주시고 사랑으로 대해 주신 점 너무나 감사드립니다. 논문 심사위원으로서 지적하여 주신 내용과 많은 조언을 해주셔서 완벽한 논문이 되게하여 주신 최연욱 교수님과 서진호 교수님께도 감사의 말씀을 전해드립니다. 그리고 한참 부족한 제 논문을 완성시켜 주시고 학문적 지식뿐만 아니라 인생의 선배로서 많은 조언을 해주신 김학경 박사님께 깊은 감사드립니다. 마음의 고향인 남해에 놀러갈 때마다 어머니의 마음으로 학생들을 살뜰히 챙겨주신 사모님께도 감사드립니다.

석사과정 중 저의 발전가능성을 믿고 채용해주신 (주)엔스퀘어 임혁 사장님께도 감사의 말씀을 드립니다. 사회의 첫 발을 내딛을 수 있게 큰 도움을 주시고, 어디서나 기본만 잘 지키면 된다는 조언을 해주신 김대환 박사님께도 감사드립니다. 항상 회사의 발전을 위해 힘쓰는 회사 식구들 모두 존경합니다. ISAMPE학회 때 처음 만나서 잊지 못할 좋은 추억을 공유하며 덕담 많이 해주시는 김현중 박사님께도 감사드립니다. URAI학회 때 지쳐있던 저에게 격려를 많이 주셨던 충북대학교 김근우 교수님께도 감사드립니다.

제가 석사과정을 무사히 마칠 수 있었던 것은 CIMEC 연구실의 앞선 많은 선배님들이 있기에 가능했습니다. 이 영광을 선배님들께 돌립니다. CIMEC 연구실에서 어언 3년간 생활하면서

함께 동고동락했던 선배님, 동기, 후배님들에게 감사드립니다. 처음 시작한 학부시절에 큰 도움을 주신 김진욱, 정재훈, 김진일, 이정웅, Doung Van Tu, Sheng Dong Bo, Gao Tian Shui 선배님들께 감사드리고, 작년까지 함께 고생하며 추억이 많은 Nguyen Trong Hai, Nguyen Huy Hung 선배님들께 감사드립니다. 특히 오랜시간 함께 생활했던 동기 오종민, 장환열님께 감사드리고 건승을 기원합니다. 졸업동기인 오종민, Tran Minh Thien님께 축하드리며, 같이 졸업하지 못해 아쉽지만 스스로의 열정으로 한 학기 더 공부하기로 한 Jotje Rantung님께도 감사드립니다. 남은 기간 박사과정으로 열심히 공부 중이며 도움을 많이 준 Nguyen Van Lanh님께 감사드리고, 저의 뒤를 이어 차기방장 역할을 수행하고 있는 석사과정 김성원님께도 감사드립니다. 또한 기간은 짧지만, 연구실에 들어와서 열심히 공부중인 학부생 김성락, 김동영 후배님께 감사드립니다. 이외에도 잠시 연구실 생활을 함께했던 손정림, 이지은 후배님께 감사드립니다. 항상 연구실의 든든한 버팀목이 되어주시고 저를 노심초사 걱정해주시며 정신적으로도 많은 도움을 주신 김희영 선생님께 감사의 말씀을 드립니다.

주위에서 서로 힘이 되어주고 정신적으로 큰 도움을 준 안광보, 권정욱, 정진호, 김종완, 송평훈, 김호철, 이봉우, 박진우, 김현수, 유근호, 구경웅, 공정희, 이하나님께도 감사의 말씀을 드립니다.

마지막으로 항상 뒤에서 묵묵히 뒷바라지 해주시느라 고생하시는 부모님, 그 은혜 평생 갚아도 모자랄 것 같습니다. 그리고 어릴적 저를 많이 챙겨주시고 예뻐해주신 할머니와 외할머니,

최근 몸이 많이 편찮으신데 모두 건강하셨으면 좋겠습니다.  
마지막으로 세상에서 제일 귀여운 우리 동생들 김지수, 김지은  
님께 감사의 마음을 전하면서 이만 줄이겠습니다.

2018년 12월

김 창 규



## 참고문헌

- [1] A. G. Ozkil, Fan Zhun, S. Dawids, H. Aanes, J. K. Kristensen and K. H. Christensen, “Service Robots for Hospitals: A Case Study of Transportation Tasks in a Hospital”, IEEE International Conference on Automation & Logistics, pp. 289-294, 2009
- [2] J. Sankari and R. Imtiaz, “Automated Guided Vehicle (AGV) for Industrial Sector”, IEEE 10<sup>th</sup> International Conference on Intelligent Systems & Control (ISCO), pp. 1-5, 2016
- [3] M. Troesch, S. Chien, Y. Chao and J. Farrara, “Planning and Control of Marine Floats in the Presence of Dynamic, Uncertain Currents”, 26<sup>th</sup> International Conference on Automated Planning and Scheduling (ICAPS), pp. 431-439, 2016
- [4] N. S. Flann, S. L. Hansen and S. A. Gray, “Point-to-Point Path Planning”, US Patent, No. 10/680,307, 2003

- [5] R. N. De Carvalho, H. A. Vidal, P. Vieira and M. I. Ribeiro, “Complete Coverage Path Planning and Guidance for Cleaning Robots”, IEEE International Symposium on Industrial Electronics Industrial Electronics, pp. 677-682, 1997
- [6] 김시중, 강정원, 정명진, “효율적 커버리지 경로 계획 및 동적 환경에서의 경로주행”, 한국로봇학회, pp. 304-309, 2007
- [7] S. Koenig, M. Likhachev and D. Furcy, “Lifelong Planning A\*”, Artificial Intelligence, pp. 93-146, 2004
- [8] K. I. Trovato, “Differential A\*: An Adaptive Search Method Illustrated with Robot Path Planning for Moving Obstacles and Goals, and an Uncertain Environment”, Journal of Pattern Recognition and Artificial Intelligence, Vol. 4, No. 2, 1990
- [9] A. Stentz, “The Focussed D\* Algorithm for Real-Time Replanning”, International Joint Conference on Artificial Intelligence(IJCAI), 1995

- [10] D. Mackay, “Path Planning with D\* Lite”, DRDC Suffield TM. 2005
- [11] D. Belanova, M. Mach, P. Sincak and K. Yoshida, “Path Planning on Robot Based on D\* Lite Algorithm”, IEEE World Symposium on Digital Intelligence for Systems and Machines (DISA), pp. 125-130, 2018
- [12] T. D. Viet, P. T. Doan, N. Hung, H. K. Kim and S. B. Kim, “Tracking Control of a Three-Wheeled Omni-directional Mobile Manipulator System with Disturbance and Friction”, Journal of Mechanical Science & Technology, pp. 2197-2211, 2012
- [13] B. T. Luan, “Trajectory Tracking Control of Omni-directional Automated Guided Vehicles Using Laser Sensor”, 박사학위논문, 부경대학교, 2014
- [14] P. P. Sandi, “Path Planning and Fault Detection for Automatic Guided Vehicle Based on Multiple Positioning Modules”, 박사학위논문, 부경대학교, 2015

- [15] S. H. Park and G. W. Kim, “Expanded Guide Circle-based Obstacle Avoidance for the Remotely Operated Mobile Robot”, Journal of Electrical Engineering and Technology, Vol. 9, pp. 1034-1042, 2014
- [16] 김곤우, 차영엽, “실외 지형에서의 GPS/IMU 센서 기반 이동로봇의 효율적인 위치 추정”, 제어로봇 시스템학회 합동학술대회, pp. 126-129, 2009
- [17] <https://www.naverlabs.com>
- [18] <http://www.lgv.co.kr>
- [19] D. Ferguson and A. Stentz, “The Field D\* Algorithm for Improved Path Planning and Replanning in Uniform and Non-Uniform Cost Environments”, Tech. Report, CMU-RI-TR-05-19, Robotics Institute, Carnegie Mellon University, 2005
- [20] [https://en.wikipedia.org/wiki/Hamiltonian\\_path](https://en.wikipedia.org/wiki/Hamiltonian_path)
- [21] R. L. Williams, B. E. Carter, P. Gallina and G. Rosati, “Dynamic Model with Slip for Wheeled Omni-

directional Robots”, IEEE Transactions on Robotics and Automation, Vol. 18, pp. 285–293, 2002

- [22] J. W. Kim, “Coverage Path Planning and Its Tracking Controller Design for an Underwater Mining Robot”, 공학석사학위논문, 부경대학교, 2016
- [23] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar and D. Hull, “Morse Decompositions for Coverage Tasks”, The International Journal of Robotics Research, Vol. 21, pp. 331–344, 2002
- [24] Y. M. Park, “A Study on Trajectory Tracking Control of Three Wheeled Omnidirectional Mobile Robot”, 공학석사학위논문, 부경대학교, 2013
- [25] H. H. Lin and C. C. Tsai, “Laser Pose Estimation and Tracking Using Fuzzy Extended Information Filtering for an Autonomous Mobile Robot”, Journal of Intelligent Robotic system, Vol. 53, No. 2, pp. 119–143, 2008
- [26] S. Zhang and W. Qian, “Dynamic Backstepping Control for Pure-Feedback Nonlinear Systems”,

Computational Aerodynamics Institution, pp. 1-20,  
2017

- [27] R. Freeman and L. Praly, “Integrator Backstepping for Bounded Controls and Control Rates”, IEEE Transactions on Automatic Control, Vol. 43, No. 2, pp. 258-262, 1998
- [28] Y. Zhang, B. Fidan and P. A. Loannou, “Backstepping Control of Linear Time-Varying Systems With Known and Unknown Parameters”, IEEE Transactions on Automatic Control, Vol. 48, No. 11, pp. 1908-1925, 2003
- [29] A. Jadbabaie, J. Yu and J. Hauser, “Stabilizing Receding Horizon Control of Nonlinear Systems: A Control Lyapunov Function Approach”, IEEE American Control Conference, 1999
- [30] Y. Lin and E. D. Sontag, “Control-Lyapunov Universal Formulas for Restricted Inputs”, Control Theory Adv. Technol., Vol. 10, No. 4, pp. 1981-2004, 1995.

## 학술논문 발표 및 게재

- [1] Hung Nguyen Huy, Chang Kyu Kim, Van Cuu Ho and Sang Bong Kim, “Tracking Controller Design for Omni-directional Automated Guided Vehicles Using Backstepping and Model Reference Adaptive Control”, International Conference on Advanced Engineering Theory and Applications (AETA), pp. 715-725, 2017
- [2] Thien Minh Tran, Chang Kyu Kim, Hak Kyeong Kim, Dae Hwan Kim and Sang Bong Kim, “Determination of Injury Rate on Fish Surface Based on Fuzzy C-means Clustering Algorithm and L\*a\*b\* Color Space Using ZED Stereo Camera”, IEEE 15th International Conference on Ubiquitous Robots (UR), pp. 472-477, 2018
- [3] Chang Kyu Kim, Sung Won Kim, Hung Nguyen Huy and Sang Bong Kim, “Path Planning for Automatic Guided Vehicle with Multiple Target Points in Known Environment”, International Conference on Advanced Engineering Theory and Applications (AETA),

pp. 726-735, 2017

- [4] Chang Kyu Kim, Hung Nguyen Huy, Dae Hwan Kim and Sang Bong Kim, "Path Planning for Automatic Guided Vehicle with Multiple Target Points in Dynamic Environment", The 2<sup>nd</sup> International Joint Conference on Advanced Engineering and Technology (IJCAET) and International Symposium on Advanced Mechanical and Power Engineering (ISAMPE), Vol. 159, 2018
- [5] Hung Nguyen Huy, Dae Hwan Kim, Chang Kyu Kim and Sang Bong Kim, "A Simple Path Planning for Automatic Guided Vehicle in Unknown Environment", IEEE 14<sup>th</sup> International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2017
- [6] Hung Nguyen Huy, Chang Kyu Kim, Hyuk Yim, Dae Hwan Kim and Sang Bong Kim, "A Simple Path Planning for Automatic Guided Vehicle in Unknown Environment", IEEE 14<sup>th</sup> International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), 2017

# 부 록

## 부록 A

식 (4.8)에 대한 증명:

식 (4.7)에 주어진 OAGV의 기구학적 방정식은 다음과 같다:

$$\omega_i = \frac{1}{r} [\dot{\mathbf{p}}_C^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{w_i} + \mathbf{p}_{w_i}^T \dot{\mathbf{R}}_1^T(\Phi_C) \mathbf{R}_1(\Phi_C) \mathbf{d}_{w_i}] \quad (\text{A.1})$$

식 (4.1)~(4.3) 으로부터, 식 (A.1)은 다음과 같이 좀더 명확하게 표현가능하다:

$$\begin{aligned} \omega_i &= \frac{1}{r} [\dot{\mathbf{p}}_C^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{w_1} + \mathbf{p}_{w_1}^T \dot{\mathbf{R}}_1^T(\Phi_C) \mathbf{R}_1(\Phi_C) \mathbf{d}_{w_1}] \\ &= \frac{1}{r} \left[ \begin{array}{c} \begin{bmatrix} \dot{X}_C & \dot{Y}_C \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ + \begin{bmatrix} L & 0 \end{bmatrix} \begin{bmatrix} -\Phi_C \sin(\Phi_C) & -\Phi_C \cos(\Phi_C) \\ \Phi_C \cos(\Phi_C) & -\Phi_C \sin(\Phi_C) \end{bmatrix}^T \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{array} \right] \\ &= \frac{1}{r} \left[ \begin{array}{c} -\sin(\Phi_C) \dot{X}_C + \cos(\Phi_C) \dot{Y}_C \\ + \dot{\Phi}_C \begin{bmatrix} L & 0 \end{bmatrix} \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) \\ -\cos(\Phi_C) & -\sin(\Phi_C) \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{array} \right] \\ &= \frac{1}{r} \left[ \begin{array}{c} -\sin(\Phi_C) \dot{X}_C + \cos(\Phi_C) \dot{Y}_C + \dot{\Phi}_C \begin{bmatrix} L & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \end{array} \right] \end{aligned}$$

$$= \frac{1}{r} \left[ -\sin(\Phi_C) \dot{X}_C + \cos(\Phi_C) \dot{Y}_C + L \dot{\Phi}_C \right] \quad (\text{A.2})$$

$$\begin{aligned} \omega_2 &= \frac{1}{r} \left[ \dot{\mathbf{p}}_C^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{w2} + \mathbf{p}_{w2}^T \dot{\mathbf{R}}_1^T(\Phi_C) \mathbf{R}_1(\Phi_C) \mathbf{d}_{w2} \right] \\ &= \frac{1}{r} \left[ \begin{array}{c} \left[ \dot{X}_C \quad \dot{Y}_C \right] \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \\ + L \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} -\Phi_C \sin(\Phi_C) & -\Phi_C \cos(\Phi_C) \\ \Phi_C \cos(\Phi_C) & -\Phi_C \sin(\Phi_C) \end{bmatrix}^T \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\ &= \frac{1}{r} \left[ \begin{array}{c} \left[ \dot{X}_C \quad \dot{Y}_C \right] \begin{bmatrix} -\cos(\Phi_C) \sin(\frac{\pi}{3}) + \sin(\Phi_C) \cos(\frac{\pi}{3}) \\ \sin(\Phi_C) \sin(\frac{\pi}{3}) - \cos(\Phi_C) \cos(\frac{\pi}{3}) \end{bmatrix} \\ + L \dot{\Phi}_C \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) \\ -\cos(\Phi_C) & -\sin(\Phi_C) \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\ &= \frac{1}{r} \left[ \begin{array}{c} \left[ \dot{X}_C \quad \dot{Y}_C \right] \begin{bmatrix} -\sin(\frac{\pi}{3} - \Phi_C) \\ -\cos(\frac{\pi}{3} - \Phi_C) \end{bmatrix} \\ + L \dot{\Phi}_C \begin{bmatrix} -\frac{1}{2} & \frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\ &= \frac{1}{r} \left[ -\sin(\frac{\pi}{3} - \Phi_C) \dot{X}_C - \cos(\frac{\pi}{3} - \Phi_C) \dot{Y}_C + L \dot{\Phi}_C \right] \quad (\text{A.3}) \end{aligned}$$

$$\begin{aligned}
\omega_3 &= \frac{1}{r} [\dot{\mathbf{p}}_C^T \mathbf{R}_1(\Phi_C) \mathbf{d}_{W3} + \mathbf{p}_{W3}^T \dot{\mathbf{R}}_1^T(\Phi_C) \mathbf{R}_1(\Phi_C) \mathbf{d}_{W3}] \\
&= \frac{1}{r} \left[ \begin{array}{c} \begin{bmatrix} \dot{X}_C & \dot{Y}_C \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \\ +L \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} -\Phi_C \sin(\Phi_C) & -\Phi_C \cos(\Phi_C) \\ \Phi_C \cos(\Phi_C) & -\Phi_C \sin(\Phi_C) \end{bmatrix}^T \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\
&= \frac{1}{r} \left[ \begin{array}{c} \begin{bmatrix} \dot{X}_C & \dot{Y}_C \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) \sin(\frac{\pi}{3}) + \sin(\Phi_C) \cos(\frac{\pi}{3}) \\ \sin(\Phi_C) \sin(\frac{\pi}{3}) - \cos(\Phi_C) \cos(\frac{\pi}{3}) \end{bmatrix} \\ +L \dot{\Phi}_C \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) \\ -\cos(\Phi_C) & -\sin(\Phi_C) \end{bmatrix} \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\
&= \frac{1}{r} \left[ \begin{array}{c} \begin{bmatrix} \dot{X}_C & \dot{Y}_C \end{bmatrix} \begin{bmatrix} \sin(\frac{\pi}{3} + \Phi_C) \\ -\cos(\frac{\pi}{3} + \Phi_C) \end{bmatrix} \\ +L \dot{\Phi}_C \begin{bmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \end{array} \right] \\
&= \frac{1}{r} \left[ \sin(\frac{\pi}{3} + \Phi_C) \dot{X}_C - \cos(\frac{\pi}{3} + \Phi_C) \dot{Y}_C + L \dot{\Phi}_C \right] \quad (A.4)
\end{aligned}$$

식 (A.2)~(A.4)로부터, 다음과 같은 행렬식이 얻어진다:

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{r} \begin{bmatrix} -\sin(\Phi_C) & \cos(\Phi_C) & L \\ -\sin(\frac{\pi}{3} - \Phi_C) & -\cos(\frac{\pi}{3} - \Phi_C) & L \\ \sin(\frac{\pi}{3} + \Phi_C) & -\cos(\frac{\pi}{3} + \Phi_C) & L \end{bmatrix} \begin{bmatrix} \dot{X}_C \\ \dot{Y}_C \\ \dot{\Phi}_C \end{bmatrix} \quad (A.5)$$

## 부록 B

식 (4.17)에 대한 증명:

DC모터와 전방향 구동바퀴가 결합된 회로 구성도는 Fig. B.1과 같다.

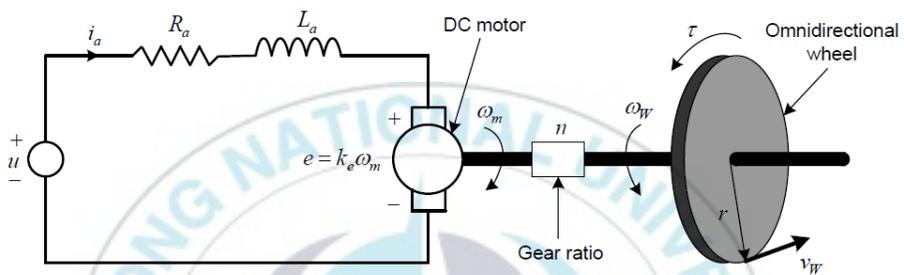


Fig. B.1 Circuit diagram combining DC motor and omni-directional drive wheels

Fig. B.1에서 각 기호들의 의미;  $u$ 는 각 DC모터에 걸리는 전기자 전압,  $e$ 는 각 DC모터의 역기전력,  $i_a$ 는 전기자 전류,  $R_a$ 는 전기자 저항,  $L_a$ 는 전기자 인덕턴스,  $\omega_m$ 은 DC모터의 각속도,  $v_w$ 는 바퀴의 선속도,  $\omega_w$ 는 바퀴의 각속도,  $\tau$ 는 전자기 토크,  $k_e$ 는 역기전력상수,  $r$ 은 각 바퀴의 반경이고,  $n$ 은 모터의 감속비이다.

각 DC모터의 동력전달은 다음 식 (B.1)과 같이 표현가능하다:

$$u = R_a i_a + L_a \frac{di_a}{dt} + k_e \omega_m \quad (\text{B.1})$$

DC모터의 전기적 시간상수는 기계적 시간상수와 비교했을 때, 매우 작기 때문에  $L_a \frac{di_a}{dt} = 0$  으로 봐도 무방하다.

따라서, 식 (B.1)을 다음과 같이 쓰여진다.

$$i_a = (u - k_e \omega_m) / R_a \quad (\text{B.2})$$

한편, 모터의 각속도와 바퀴의 각속도사이의 관계식은 식 (B.3)과 같다.

$$\omega_m = n \omega_w \quad (\text{B.3})$$

DC모터의 토크는 토크상수  $k_\tau$  를 고려해보면 식 (B.4)와 같다.

$$\tau = k_\tau i_a \quad (\text{B.4})$$

위의 식 (B.2)~(B.3)을 식 (B.4)에 대입하면 다음 식 (B.5)와 같은 토크방정식을 얻을 수 있다.

$$\tau = k_\tau(u - k_e n \omega_w) / R_a = (k_\tau / R_a)u - (k_\tau k_e / R_a)n \omega_w \quad (\text{B.5})$$

바퀴의 각속도  $\omega_w$  는 다음과 같다.

$$\omega_w = v_w / r \quad (\text{B.6})$$

식 (B.5)~(B.6)을 종합해보면, DC모터가 바퀴에 작용하는 힘은 식 (B.7)~(B.8)과 같이 표현할 수 있다:

$$f = \frac{\tau}{r} = [k_\tau / (rR_a)]u - [k_\tau k_e n / (r^2 R_a)]v_w \quad (\text{B.7})$$

$$\Leftrightarrow f = \rho_1 u - \rho_2 v_w \quad (\text{B.8})$$

식 (B.7)과 식 (B.8)로부터 DC모터의 변수에 따라 변하는 모터특성계수는 다음과 같이 얻어진다.

$$\rho_1 = k_\tau / (rR_a) \text{ and } \rho_2 = k_\tau k_e n / (r^2 R_a) \quad (\text{B.9})$$

실험에서 사용된 DC모터의 변수와 바퀴의 반경을 모터특성계수들을 식 (B.9)에 대입한 계산 결과는 다음과 같다.

$$\rho_1 = k_\tau / (rR_a) = \frac{0.56368}{0.0625 \times 6} \approx 1.5 \text{ N/V}$$

$$\rho_2 = k_\tau k_e n / (r^2 R_a) = 0.56368 \times 6.67 \times 10^{-3} \times \frac{12}{0.0625^2 \times 6} \approx 1.924 \text{ Kg/s}$$



## 부록 C

식 (4.18)에 대한 증명:

식 (4.14)~(4.16)에서 주어지는 OAGV의 동역학적 방정식은 다음과 같다.

$$\sum_{i=1}^3 (f_i - f_{Mi}) \mathbf{R}_1(\Phi_C) \mathbf{d}_{wi} - \mathbf{f}_A = m_1 \ddot{\mathbf{p}}_C \quad (\text{C.1})$$

$$\mathbf{f}_A = \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \quad (\text{C.2})$$

$$L \sum_{i=1}^3 (f_i - f_{Mi}) = I \ddot{\Phi}_C \quad (\text{C.3})$$

식 (C.2)를 식 (C.1)에 대입한 후 전개해 보면 다음과 같다.

$$\begin{aligned} & (f_1 - f_{M1}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} d_{w1} + (f_2 - f_{M2}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} d_{w2} \\ & + (f_3 - f_{M3}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} d_{w3} - \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \\ & = m \begin{bmatrix} \ddot{X}_C \\ \ddot{Y}_C \end{bmatrix} \end{aligned}$$

$$\begin{aligned}
&\Leftrightarrow (f_1 - f_{M1}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + (f_2 - f_{M2}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} -\frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} \\
&+ (f_3 - f_{M3}) \begin{bmatrix} \cos(\Phi_C) & -\sin(\Phi_C) \\ \sin(\Phi_C) & \cos(\Phi_C) \end{bmatrix} \begin{bmatrix} \frac{\sqrt{3}}{2} \\ -\frac{1}{2} \end{bmatrix} - \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \\
&= m \begin{bmatrix} \ddot{X}_C \\ \ddot{Y}_C \end{bmatrix} \\
&\Leftrightarrow (f_1 - f_{M1}) \begin{bmatrix} -\sin(\Phi_C) \\ \cos(\Phi_C) \end{bmatrix} + (f_2 - f_{M2}) \begin{bmatrix} -\frac{\sqrt{3}}{2} \cos(\Phi_C) + \frac{1}{2} \sin(\Phi_C) \\ -\frac{\sqrt{3}}{2} \sin(\Phi_C) - \frac{1}{2} \cos(\Phi_C) \end{bmatrix} \\
&+ (f_3 - f_{M3}) \begin{bmatrix} \frac{\sqrt{3}}{2} \cos(\Phi_C) + \frac{1}{2} \sin(\Phi_C) \\ \frac{\sqrt{3}}{2} \sin(\Phi_C) - \frac{1}{2} \cos(\Phi_C) \end{bmatrix} - \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \\
&= m \begin{bmatrix} \ddot{X}_C \\ \ddot{Y}_C \end{bmatrix} \tag{C.4}
\end{aligned}$$

식 (C.4)에서,  $\sin\left(\frac{\pi}{3}\right) = \frac{\sqrt{3}}{2}$ ,  $\cos\left(\frac{\pi}{3}\right) = \frac{1}{2}$  이므로 삼각함수 덧셈 공식에 의해 식 (C.5)와 같이 정리된다.

$$\begin{aligned}
&\Leftrightarrow (f_1 - f_{M1}) \begin{bmatrix} -\sin(\Phi_C) \\ \cos(\Phi_C) \end{bmatrix} + (f_2 - f_{M2}) \begin{bmatrix} -\sin\left(\frac{\pi}{3} - \Phi_C\right) \\ -\cos\left(\frac{\pi}{3} - \Phi_C\right) \end{bmatrix} \\
&+ (f_3 - f_{M3}) \begin{bmatrix} \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ -\cos\left(\frac{\pi}{3} + \Phi_C\right) \end{bmatrix} - \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \end{bmatrix} \\
&= m \begin{bmatrix} \ddot{X}_C \\ \ddot{Y}_C \end{bmatrix} \tag{C.5}
\end{aligned}$$

식 (C.3)을 전개하면 다음과 같다.

$$L(f_1 - f_{M1}) + L(f_2 - f_{M2}) + L(f_3 - f_{M3}) = I\ddot{\Phi}_C \quad (C.6)$$

식 (C.5)와 식 (C.6)을 행렬(Matrix)로 표현하면 다음과 같다.

$$\begin{bmatrix} m\ddot{x}_C \\ m\ddot{y}_C \\ I\ddot{\Phi}_C \end{bmatrix} = \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \\ - \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} f_{M1} \\ f_{M2} \\ f_{M3} \end{bmatrix} \\ - \begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \\ 0 \end{bmatrix} \quad (C.7)$$

여기서,

$$f_{id} = \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} = \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} f_{M1} \\ f_{M2} \\ f_{M3} \end{bmatrix}$$

$$\begin{bmatrix} f_{A1} \cos(\Phi_C) + f_{A2} \cos\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \cos\left(\frac{4\pi}{3} + \Phi_C\right) \\ f_{A1} \sin(\Phi_C) + f_{A2} \sin\left(\frac{2\pi}{3} + \Phi_C\right) + f_{A3} \sin\left(\frac{4\pi}{3} + \Phi_C\right) \\ 0 \end{bmatrix}$$

식 (C.7)은 (C.8)과 같이 정리된다.

$$\begin{bmatrix} m\ddot{X}_C \\ m\ddot{Y}_C \\ I\ddot{\Phi}_C \end{bmatrix} = \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} - \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} \quad (C.8)$$

다음, 식 (4.17)의  $f_i (i=1,2,3)$  을 식 (C.8)에 대입하면 다음과 같이 얻어진다.

$$\begin{bmatrix} m\ddot{X}_C \\ m\ddot{Y}_C \\ I\ddot{\Phi}_C \end{bmatrix} = \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} \rho_1 u_1 - \rho_2 v_{w1} \\ \rho_1 u_2 - \rho_2 v_{w2} \\ \rho_1 u_3 - \rho_2 v_{w3} \end{bmatrix} - \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} \quad (C.9)$$

OAGV의 기구학적 모델링으로부터  $\mathbf{v}_w = \mathbf{r}\boldsymbol{\omega} = \mathbf{H}^{-1}(\Phi_C)\dot{\mathbf{q}}_C$  이므로 식 (C.9)에 대입하여 정리하면 다음과 같이 얻어진다.

$$\begin{aligned}
\begin{bmatrix} m\ddot{X}_c \\ m\dot{Y}_c \\ I\ddot{\Phi}_c \end{bmatrix} &= \rho_1 \begin{bmatrix} -\sin(\Phi_c) & -\sin\left(\frac{\pi}{3}-\Phi_c\right) & \sin\left(\frac{\pi}{3}+\Phi_c\right) \\ \cos(\Phi_c) & -\cos\left(\frac{\pi}{3}-\Phi_c\right) & -\cos\left(\frac{\pi}{3}+\Phi_c\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \\
&- \rho_2 \begin{bmatrix} -\sin(\Phi_c) & -\sin\left(\frac{\pi}{3}-\Phi_c\right) & \sin\left(\frac{\pi}{3}+\Phi_c\right) \\ \cos(\Phi_c) & -\cos\left(\frac{\pi}{3}-\Phi_c\right) & -\cos\left(\frac{\pi}{3}+\Phi_c\right) \\ L & L & L \end{bmatrix} \\
&\times \begin{bmatrix} -\sin(\Phi_c) & \cos(\Phi_c) & L \\ -\sin\left(\frac{\pi}{3}-\Phi_c\right) & -\cos\left(\frac{\pi}{3}-\Phi_c\right) & L \\ \sin\left(\frac{\pi}{3}+\Phi_c\right) & -\cos\left(\frac{\pi}{3}+\Phi_c\right) & L \end{bmatrix} \begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ \dot{\Phi}_c \end{bmatrix} - \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} \quad (C.10)
\end{aligned}$$

식 (C.10)을 삼각함수 곱셈공식에 의해 정리해보면 다음과 같이 얻어진다.

$$\begin{aligned}
\begin{bmatrix} m\ddot{X}_c \\ m\dot{Y}_c \\ I\ddot{\Phi}_c \end{bmatrix} &= \rho_1 \begin{bmatrix} -\sin(\Phi_c) & -\sin\left(\frac{\pi}{3}-\Phi_c\right) & \sin\left(\frac{\pi}{3}+\Phi_c\right) \\ \cos(\Phi_c) & -\cos\left(\frac{\pi}{3}-\Phi_c\right) & -\cos\left(\frac{\pi}{3}+\Phi_c\right) \\ L & L & L \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \\
&- \rho_2 \begin{bmatrix} \frac{3}{2} & 0 & 0 \\ 0 & \frac{3}{2} & 0 \\ 0 & 0 & 3L^2 \end{bmatrix} \begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ \dot{\Phi}_c \end{bmatrix} - \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} \quad (C.11)
\end{aligned}$$

$$\Leftrightarrow \begin{bmatrix} m\ddot{X}_c \\ m\dot{Y}_c \\ I\ddot{\Phi}_c \end{bmatrix} = \rho_1 \mathbf{H}^T \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} - \frac{3}{2} \rho_2 \begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ 2L^2 \dot{\Phi}_c \end{bmatrix} - \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix} \quad (C.12)$$

$$\Leftrightarrow \begin{bmatrix} m\ddot{X}_c \\ m\dot{Y}_c \\ I\ddot{\Phi}_c \end{bmatrix} = \rho_1 \mathbf{H}^T u_i - \frac{3}{2} \rho_2 \begin{bmatrix} \dot{X}_c \\ \dot{Y}_c \\ 2L^2 \dot{\Phi}_c \end{bmatrix} - f_{id} \quad (C.13)$$

$$\Leftrightarrow \begin{bmatrix} m_1 \ddot{X}_C \\ m_1 \ddot{Y}_C \\ I \ddot{\Phi}_C \end{bmatrix} + \frac{3}{2} \rho_2 \begin{bmatrix} \dot{X}_C \\ \dot{Y}_C \\ 2L^2 \dot{\Phi}_C \end{bmatrix} = \rho_1 \mathbf{H}^{-T} u_i - f_d \quad (\text{C.14})$$

여기서,  $\mathbf{H}^{-T} = \begin{bmatrix} -\sin(\Phi_C) & -\sin\left(\frac{\pi}{3} - \Phi_C\right) & \sin\left(\frac{\pi}{3} + \Phi_C\right) \\ \cos(\Phi_C) & -\cos\left(\frac{\pi}{3} - \Phi_C\right) & -\cos\left(\frac{\pi}{3} + \Phi_C\right) \\ L & L & L \end{bmatrix}$ ,  $\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$ ,  $\mathbf{f}_d = \begin{bmatrix} f_{1d} \\ f_{2d} \\ f_{3d} \end{bmatrix}$ .

