



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](#)

공학박사 학위논문

센서 모델 기반 수중트랙로봇의
수중항법 시스템 연구

A Study on Underwater Navigation System of
Sensor Model-based Underwater Track Robot

지도교수 최 형 식

2020년 2월

한국해양대학교 대학원

기계공학과

지 대 형

본 논문을 지대형의 공학박사 학위논문으로 인준함.



위원장	김 준 영
위 원	최 형 식
위 원	유 삼 상
위 원	조 용 성
위 원	주 문 갑



2019년 12월

한국해양대학교 대학원

목 차

List of Tables	iv
List of Figures	v
Abstract	x

1. 서 론

1.1 연구 배경	1
1.1.1 수중트랙로봇	2
1.1.2 항법	5
1.2 연구 목표	7
1.3 논문 구성	8

2. 항법 센서의 수학적 모델

2.1 관성 측정 장치	9
2.1.1 가속도계	10
2.1.2 각속도계	15
2.1.3 자력계	20

3. 융합 항법 알고리즘

3.1 좌표계	25
3.2 INS 항법 알고리즘 설계	26
3.2.1 자세 추정	26
3.2.2 속도 추정	33
3.2.3 위치 추정	34
3.2.4 INS 오차 모델	35
3.3 융합 항법 알고리즘 설계	38
3.3.1 시스템 오차 모델	38
3.3.2 측정 오차 모델	41
3.4 융합 항법 알고리즘 구성	41
3.5 융합 항법 시뮬레이션	42
3.5.1 S자 궤적 항법 시뮬레이션	44
3.5.2 사각형 궤적 항법 시뮬레이션	50
3.5.3 모의 진회수 항법 시뮬레이션	56

4. 수중항법 시스템 및 플랫폼 구성

4.1 수중항법 시스템	63
4.1.1 시스템 구성	63
4.1.2 운용 시스템	67

4.2 수중트랙로봇	69
4.2.1 플랫폼 구성	69
4.2.2 운용 시스템	76
 5. 융합 항법 성능 실험	
5.1 트랜칭 수중트랙로봇에 적용한 융합 항법 성능 실험	77
5.1.1 Heading 추정 성능 실험	79
5.1.2 실효역 융합 항법 성능 실험	80
5.2 수중트랙로봇에 적용한 융합 항법 성능 실험	86
5.2.1 기본 항법 성능 실험	88
5.2.2 미션 궤적 항법 성능 실험	91
5.2.3 사각형 궤적 항법 성능 실험 #1	95
5.2.4 사각형 궤적 항법 성능 실험 #2	98
 6. 결론	103
 부록	106
 참고문헌	114
 감사의 글	119

List of Tables

Table 1.1 Features of Underwater Track Robots	4
Table 2.1 Parameters of accelerometer	10
Table 2.2 Parameters of gyroscope	15
Table 2.3 Parameters of magnetometer	20
Table 3.1 Specifications of the IMU used in the simulation	43
Table 3.2 Specifications of the magnetometer used in the simulation .	43
Table 3.3 Specifications of the GPS used in the simulation	43
Table 4.1 Specifications of underwater navigation system	64
Table 4.2 Specifications of sensors included in navigation system	66
Table 4.3 Specifications of UTR	70
Table 4.4 Power specifications for each component of underwater navigation system	72
Table 4.5 Power specifications of underwater navigation system	72
Table 4.6 Power specifications for each component of UTR	73
Table 4.7 Power specifications of UTR	73
Table 4.8 Total power and parameters of the system	74
Table 5.1 Specification of commercial trenching UTR	78
Table 5.2 Specifications of sensors included in trenching UTR	79
Table 5.3 Specifications of sensors used in ground experiments	87
Table 5.4 Position error result of navigation experiments	90
Table 5.5 Position RMS error result of navigation experiments	102

List of Figures

Fig. 1.1 Various marine industries in the deep sea	1
Fig. 1.2 Types and applications of underwater track robots	2
Fig. 2.1 Typical IMU sensor family(MEMS/RLG type)	9
Fig. 2.2 Block diagram of accelerometer	11
Fig. 2.3 Block diagram of gyroscope	16
Fig. 2.4 Block diagram of magnetometer	21
Fig. 3.1 Coordinate systems	25
Fig. 3.2 Kalman filter for orientation calculation	30
Fig. 3.3 Block diagram of the navigation system	41
Fig. 3.4 Angular velocity/acceleration input of IMU model (Case 1)	44
Fig. 3.5 Orientation input of IMU model (Case 1)	45
Fig. 3.6 Create a path (Case 1)	45
Fig. 3.7 Navigation sensor measurements derived from the IMU model (Case 1)	46
Fig. 3.8 Position result of navigation (Case 1)	47
Fig. 3.9 Result orientation of navigation vs. real orientation (Case 1)	47
Fig. 3.10 Each velocity results of navigation (Case 1)	48
Fig. 3.11 Position result of navigation vs. real position (Case 1)	49
Fig. 3.12 Difference between position result of navigation and real position (Case 1)	49

Fig. 3.13 Angular velocity/acceleration input of IMU model (Case 2) ..	50
Fig. 3.14 Orientation input of IMU model (Case 2)	51
Fig. 3.15 Create a path (Case 2)	51
Fig. 3.16 Navigation sensor measurements derived from the IMU model (Case 2)	52
Fig. 3.17 Position result of navigation (Case 2)	53
Fig. 3.18 Result orientation of navigation vs. real orientation (Case 2)	53
Fig. 3.19 Each velocity results of navigation (Case 2)	54
Fig. 3.20 Position result of navigation vs. real position (Case 2)	55
Fig. 3.21 Difference between position result of navigation and real position (Case 2)	55
Fig. 3.22 Angular velocity/acceleration input of IMU model (Case 3) ..	56
Fig. 3.23 Orientation input of IMU model (Case 3)	57
Fig. 3.24 Create a path (Case 3)	57
Fig. 3.25 Navigation sensor measurements derived from the IMU model (Case 3)	58
Fig. 3.26 Position result of navigation (Case 3)	59
Fig. 3.27 Position result of navigation N-E (Case 3)	59
Fig. 3.28 Position result of navigation N-D (Case 3)	60
Fig. 3.29 Result orientation of navigation vs. real orientation (Case 3)	60
Fig. 3.30 Each velocity results of navigation (Case 3)	61
Fig. 3.31 Position result of navigation vs. real position (Case 3)	62

Fig. 3.32 Difference between position result of navigation and real position (Case 3)	62
Fig. 4.1 Developed underwater navigation system	63
Fig. 4.2 Block diagram of underwater navigation system	64
Fig. 4.3 Printed circuit board of the interface board	65
Fig. 4.4 Inside of underwater navigation system	65
Fig. 4.5 Operation console of total system	67
Fig. 4.6 Operation software of underwater navigation system	68
Fig. 4.7 Developed UTR	69
Fig. 4.8 Block diagram of UTR	70
Fig. 4.9 Battery and communication parts of UTR	71
Fig. 4.10 Motor drive and connector parts of UTR	71
Fig. 4.11 Operating time of each system according to L_f	74
Fig. 4.12 Communication diagram of total system	75
Fig. 4.13 Operation software of UTR	76
Fig. 5.1 Commercial trenching UTR	77
Fig. 5.2 Underwater navigation system without encoder	78
Fig. 5.3 Result of rectangle trajectory experiment	80
Fig. 5.4 Launching underwater track robot	80
Fig. 5.5 Intervals of USBL measurements of navigation experiment ...	81
Fig. 5.6 Position result of navigation #1	82
Fig. 5.7 Velocities estimated by underwater navigation system	83

Fig. 5.8 Difference between USBL and position estimates	84
Fig. 5.9 Position result of navigation #2	85
Fig. 5.10 Location of navigation experiment	86
Fig. 5.11 Environment of navigation experiment	87
Fig. 5.12 Trajectory and navigation position results (Test 1)	88
Fig. 5.13 The error between trajectory and navigation position results (Test 1)	89
Fig. 5.14 Position results of each navigation (Test 2)	92
Fig. 5.15 N-E position result of each navigation (Test 2)	92
Fig. 5.16 N-E velocity results of each navigation (Test 2)	93
Fig. 5.17 Orientation results of navigation (Test 2)	94
Fig. 5.18 Position result of navigation (Test 3)	95
Fig. 5.19 Position result of navigation vs. GPS position (Test 3)	96
Fig. 5.20 Difference between position result of navigation and GPS position (Test 3)	96
Fig. 5.21 Velocity result of navigation (Test 3)	97
Fig. 5.22 Orientation result of navigation (Test 3)	97
Fig. 5.23 Position result of navigation (Test 4)	99
Fig. 5.24 Position result of navigation vs. GPS position (Test 4)	99
Fig. 5.25 Difference between position result of navigation and GPS position (Test 4)	100
Fig. 5.26 Velocity result of navigation (Test 4)	100
Fig. 5.27 Orientation result of navigation (Test 4)	101

Fig. A-1.1 Soft and Hard iron effects in magnetic compass	106
Fig. A-5.1 The power circuit of sensors interface	111
Fig. A-5.2 The communication circuit of sensors interface	112
Fig. A-5.3 The PCB design of sensors interface	113



A Study on Underwater Navigation System of Sensor Model-based Underwater Track Robot

Ji, DaeHyeong

Department of Mechanical Engineering

Graduate School of Korea Maritime and Ocean University

Abstract

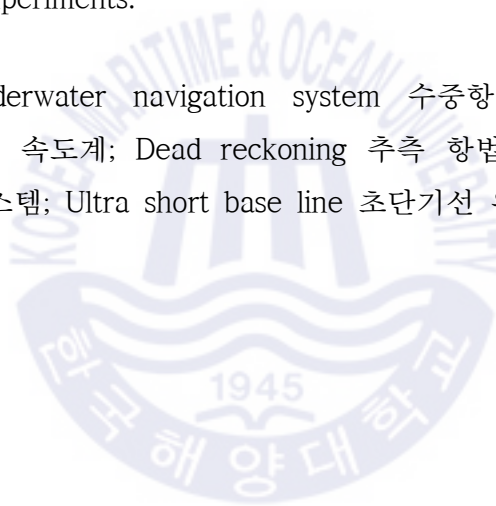
In this study, an underwater navigation algorithm was developed to apply the underwater navigation system to the underwater track robot. Generally, underwater navigation uses a Doppler Velocity Log(DVL) to measure the velocity of underwater vehicles. However, undersea platforms, such as underwater track robots, cannot use DVL due to the distance limitations of sensor operation. As a result, Dead Reckoning(DR) navigation is inevitably used, and which results in severe errors in attitude and position values over long periods of platform operation. To overcome this problem, we developed an underwater navigation system composed of coupled Inertial Navigation System(INS) composed Ultra Short Base Line(USBL) and additional track information.

The INS sensors were modeled using the mathematical model of the accelerometer, the gyroscope, the magnetometer.

Before the experiment, computer simulations were performed to analyze the expected sensor values for specific track missions in unexpected situations. Based on this, we developed an underwater navigation algorithm for a prototype underwater track robot which we developed at the lab and confirmed the effectiveness of the navigation algorithm through experiments.

For the prototype underwater track robot, we developed the navigation system, the electric hardware, the control system, and the operating system. Finally, we applied the developed INS and the underwater navigation algorithm to the platform and verified a good performance through real sea experiments.

KEY WORDS: Underwater navigation system 수중항법 시스템; Doppler velocity log 도플러 속도계; Dead reckoning 추측 항법; Inertial navigation system 관성항법시스템; Ultra short base line 초단기선 위치추적시스템.



제 1 장 서 론

1.1 연구 배경

각종 해양자원의 이용과 개발은 미래의 고도 산업사회 건설과 지속적인 경제 발전에 필수적이며, 다양한 해양공간 이용과 개발 관련 산업으로 발전할 수 있는 무한한 성장 가능성을 가지고 있다. 특히 Fig. 1.1과 같이 심해자원의 채집 작업이나 해양 건설작업, 해양플랜트 작업 등은 세계 각국이 심혈을 기울이는 중요한 해양산업이다.

그러나 심해저의 높은 수압과 산소의 부족 등의 이유로 인해 인간이 투입되기 어려운 실정이다. 수중에서 인간이 직접 수행하기 힘든 작업 현장에 인간을 대체한 로봇이 투입되어 위험하고 어려운 임무를 대행함으로써 많은 분야에서 효과적인 결과를 나타내고 있다. 현재 세계 각국에서는 망간단괴를 비롯해 각종 희소금속뿐만 아니라 심해 광물자원을 찾기 위해 심해 탐사기술을 포함하는 수중로봇 개발의 선두주자가 되고자 수많은 노력과 비용을 투자하고 있다.

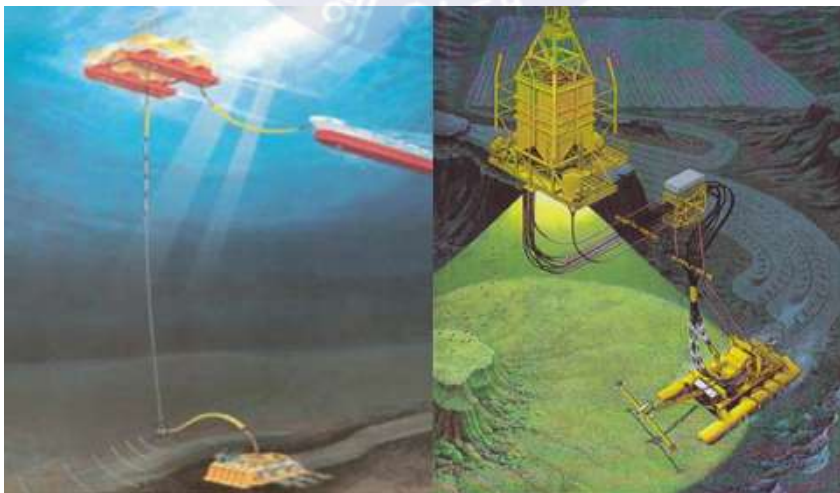


Fig. 1.1 Various marine industries in the deep sea

해양에너지 개발의 활성화 및 해저플랜트 시장의 급격한 증가와 함께 무한한 자원의 보고인 해양을 개발하기 위한 건설시장의 증가로 인해 해양에서 좀 더 열악한 입지에서 더욱 다양한 형태의 작업이 필요하게 되었으며, 수중에서 작업을 수행하는 무인잠수정(Unmanned underwater vehicle, UUV)도 좀 더 다양하고 세분화된 기능이 요구되고 있다.

1.1.1 수중트랙로봇

트랙기반 원격조종 무인잠수정(Remotely operated vehicle, ROV) 즉, 수중트랙로봇(Underwater track robot, UTR)의 일종인 수중건설로봇은 해저 지반이 단단하여 기존의 쟁기식(Plough) 또는 워터젯(Water jet)으로 작업이 어려운 조건일 경우 트랜칭 커터(Trenching cutter)를 장착하여 이를 극복하고, 다양한 작업 툴을 교체하여 활용할 수 있는 다목적 암을 장착한 플랫폼이다.

수중트랙로봇이 수행 가능한 다양한 작업공정은 Fig. 1.2과 같으며, 대표적인 공정으로는 단단한 지반에서의 케이블 매설, 파이프라인 매설을 위한 트랜칭 커팅, 퇴매우기, 토공, 고르기 작업, 암 파쇄 등을 예로 들 수 있다.



Fig. 1.2 Types and applications of underwater track robots

다양한 해양플랜트 및 해양구조물을 잇는 해저 광통신 케이블, 전력 케이블 및 파이프라인을 해저에 매설하는 작업의 경우 매설 로봇의 하단에 부착된 Trenching cutter, Water jet 등의 Tool을 사용한다. 이때 발생하는 반력을 견디기 위한 견인력과 기동성능을 얻는데 있어 트랙방식의 궤도차량이 큰 장점이 있다.

수중트랙로봇의 기술현황은 Table 1.1과 같다. 해외 선진국에서는 70년대부터 개발이 이루어졌으며, 특히 미국의 Forum사가 영국의 SMD(Soil Machine Dynamics)사와 함께 세계적으로 가장 대표적인 두 제작용업체이다. Forum사의 대표적인 모델은 XT1200이며, 작업이 가능한 수심은 2,700 m이며 워터 제팅을 이용하여 해저 지반 및 토사를 트랜칭하여 파이프라인 및 해저케이블 매설 작업을 수행한다. SMD사는 또한 각종 Tool 및 구동 방식을 스키드 방식과 트랙 방식으로 나누어 QTrencher series의 다양한 장비를 개발하여 상용화를 하였고, QTrencher1400은 트랜치커터를 장착한 트랙기반 매설 로봇이다. 국내의 경우 수중트랙로봇을 개발한 사례로는 한국해양과학기술원 부설 선박해양플랜트연구소에서 해저자원 채광로봇인 미네로(Minero)를 개발하였으며(김형우 & 홍섭, 2007; 박성재, 2011; Kim et al, 2013; 윤석민, 2016), 이 로봇은 트랙기반으로 운영되는 국내 최대 규모의 ROV이다. 그리고 한국해양과학기술원에서 창원대학교와 같이 수심 20m급 항만수중공사용 수중사석고르기를 개발하였다(김치효, 2013; Kim et al., 2018). 또한, 국내에서 유일하게 해저 케이블 및 파이프라인 매설장비를 다수 보유하고 있는 (주)케이티서브마린의 경우에도 장비들은 모두 해외에서 직접 구매한 것들이다. 이처럼 열악한 수중트랙로봇 분야에서 해외 임대 의존도를 줄이기 위해 국내의 독자적인 수중트랙로봇의 개발이 필수적이다.

위와 같은 수중트랙로봇들은 수중에서 작업자가 원하는 경로와 위치를 따라 작업을 하기 위해서는 전체의 자세와 위치 정보를 정확하게 취득할 수 있어야 하므로 수중항법 연구가 필수적으로 선행되어야 한다(Lee et al, 2004).

Table 1.1 Features of Underwater Track Robots

모델 / 제작기관 (국가)	제원 및 특징
XT1200 / Forum (미국) 	■ 제원: 9.0m(L)×6.1m(W)×3.9m(H) ■ 동력: 1200hp ■ 최대운용 수심: 2700m ■ Water jet 이용 ■ 파이프라인 및 해저케이블 매설
QTrancher1400 / SMD (영국) 	■ 제원: 7.8m(L)×5.0m(W)×6.0m(H) ■ 동력: 1050kW/1400hp ■ 최대운용 수심: 3000m ■ Trenching cutter 이용 ■ 해저케이블 매설
Minero / Kriko (한국) 	■ 제원: 1.3m(L)×1.8m(W)×2.2m(H) ■ 최대운용 수심: 6000m ■ 해저 자원채광 ■ 7 자유도 매니플레이터 장비
Underwater Leveling Equipment / Kiost (한국) 	■ 제원: 9.3m(L)×3.0m(W)×2.7m(H) ■ 질량: 21000kg ■ 최대운용 수심: 20m ■ 항만 수중공사를 위한 사석 고르기 장비

1.1.2 항법

선체가 궤도를 따라 기동하는 동안 선체의 상태(방향, 속도, 위치 등)를 자동으로 추정하는 과정을 항법(Navigation)이라 한다. 관성 항법 시스템(Inertial navigation system, INS)은 가장 널리 사용되는 추측 항법(Dead reckoning, DR) 시스템 중 하나이다. INS를 이용하여 관성 항법을 수행할 수 있다(Yunchun & Farrell, 2003). 전형적인 INS는 선체의 위치와 방향을 지속적으로 추정하기 위해 가속도계와 자이로스코프에서 가져온 측정 정보를 통합한다. 시간이 지남에 따라 다양한 센서 오류가 통합되므로 INS는 정확하고 높은 주파수(일반적으로 100 ~ 200Hz 범위) 정보를 단기간 동안만 제공할 수 있다. 이러한 이유로 저렴한 센서를 사용할 때 특히 오류가 높다. 반면에, 위성 측위 시스템(Global positioning system, GPS)은 낮은 속도(일반적으로 1 ~ 4Hz 범위)로 전역 참조된 위치 및 속도 추정을 제공한다. INS-GPS의 통합은 두 시스템의 장점을 모두 활용할 수 있는 항법 시스템을 생성할 수 있으며, 개별 시스템에서 나타나는 단점을 보완한다. 따라서 INS-GPS의 통합 시스템은 드리프트가 없이 고주파에서 차량의 전체 상태를 추정할 수 있다.

INS-GPS 통합은 크게 약 결합 시스템(Loosely coupled system), 강 결합 시스템(Tightly coupled system), 초강 결합 시스템(Ultra-tightly coupled system)으로 분류된다(Zhou, 2010). 약 결합 시스템에서 INS와 GPS는 명시적으로 융합된다(Crassidis, 2006; El-Sheim et al., 2006; Bijker & Steyn, 2008; Noureldin et al., 2008; Syed et al., 2008; Georgy et al., 2009; Liu et al., 2010; Wang & Gao, 2010; Bruggemann, 2011). 이러한 종류의 시스템은 GPS 데이터의 가용성에 크게 의존한다. 강 결합 시스템에서 GPS와 INS는 직접 융합된다(Wang et al., 2006; Bruggemann, 2011; Karamat et al., 2015). 이러한 종류의 방법에서 주요 장점은 위성이 4 개 미만인 경우에도 시스템이 GPS 측정 업데이트를 수행할 수 있다는 것이다. 하지만 이 시스템은 시스템의 구성이 복잡하다는 단점이 있다. 초강 결합 시스템에서의 INS 출력은 GPS 수신기에 대한 외부 입력으로 사용된다(Ohlmeyer, 2006). INS 출력은 빠른 신호 수집을 위한 전치 계산과 신호 추적 중 간섭 제거에 도움이 된다. 이런 종류의 시스템 구현은 종종 GPS 펄스

어에 대한 접근이 필요하기 때문에 시스템 구성이 매우 복잡하고 개발 단가가 비싸진다.

오래전부터 INS-GPS를 통합하기 위해 다양한 상태 추정 기법이 사용되었다. Sukkarieh et al. (1998)와 Qi and Moore (2002)이 제시한 기법은 선형 칼만 필터링 기법을 기반으로 한다. 시스템의 상태 변수를 추정하고 측정 잡음을 억제하는데 일반적으로 사용되는 칼만 필터(Kalman filter, KF)는 가장 강력한 상태 추정 기술 중 하나로 인식된다. KF를 사용하면 다른 센서에서 얻은 정보를 구조적으로 병합할 수 있다. 예를 들어 Tsai et al. (2013)가 제시한 위치 추정을 위한 KF 기반 방법이 있으며, 이 접근법은 시각 데이터와 무선 센서 네트워크 정보를 결합한다. 선형 필터링을 기반으로 하는 방법은 일반적으로 선형화된 모델을 사용한다. 따라서 일부 계산 시간이 절약되지만 성능이 약간 저하된다. 그러나, 현재 이용 가능한 광범위한 프로세싱 장치는 성능을 향상하기 위해 복잡한 알고리즘의 구현이 가능하다.

비선형 특성으로 인해 KF의 비선형 버전인 확장 칼만 필터(Extended Kalman filter, EKF)는 INS-GPS 융합을 계산하는데 널리 사용되는 기술이다. EKF를 구현하는 간접/직접 두 가지 기본 방법이 있다. Bijker and Steyn (2008)은 EKF를 두 번 사용하여 GPS-INS을 약 결합하였다. EKF를 두 번 사용한 이유는 첫 번째 EKF는 시스템의 자세를 추정하고 두 번째 EKF는 속도와 위치를 추정하는데 사용되기 때문이다. Wendel and Trommer (2004)은 GPS-INS의 결합과정을 강 결합을 사용하였는데, 이러한 이유는 강 결합에서 GPS의 의사 거리(pseudo-range) 정보를 사용하여 GPS의 위치 및 속도 정보를 처리하기 위함이며, 약 결합보다 우수하다는 것을 확인하였기 때문이다. Ridao et al. (2011) 및 Ribas et al. (2012)는 초단기선 초음파 위치추적시스템(Ultra-short baseline, USBL) 측정 신호의 시간 지연을 고려한 도플러 속도계(Doppler velocity log, DVL) 융합을 연구하여 자율 무인 수중선(Autonomous Underwater Vehicle, AUV)의 수중항법에 적용했다. Lee et al. (2007)는 관성 항법 알고리즘에 거리 정보를 융합한 복합항법 알고리즘을 제안하였으며, 2015년에는 USBL, DVL 및 관성 측정 장비(Inertial measurement unit, IMU)를 융합하는 연구를 수행했다 (Lee et al., 2015).

일반적으로 수중항법에 사용되는 센서는 IMU, GPS, DVL, USBL, 수심 센서 (Depth sensor) 등이 있다(Alahyari et al., 2011). USBL은 육상의 GPS와 유사한 역할인 수중의 절대 좌표를 측정하는 수중 음향 센서의 일종이다(Penas, 2009). 본 연구에서 항법 대상 플랫폼인 장비는 트랙기반 로봇 형태로 제작되어 운용 특성상 충격 외란이 지속적으로 입력되며 필연적으로 주위의 타도가 높아질 것이다. 작업에 의한 충격 외란이 입력되면 선체의 운동으로서 측정되는 가속도 신호가 불안정하게 평가되어 항법 연산이 제대로 이루어지지 않을 수도 있다. 또한 주변 타도가 높아지면 DVL을 사용할 수 없는 경우도 발생한다(Somers, 2011). 따라서 DVL외에 INS와 USBL을 이용한 수중항법을 적용하여 위치 오차를 줄이는 방법의 연구가 필요하다.

1.2 연구 목표

일반적으로 수중항법에서 DVL은 수중로봇의 속도를 추정하는데 사용된다. 그러나 1.1.1절에서 설명한 본 연구의 항법 대상 플랫폼인 수중트랙로봇과 같은 해저면에서 운용되는 플랫폼들은 센서 운용의 제약으로 DVL을 사용할 수 없다. 그렇기 때문에 고전적 추측 항법을 사용할 수밖에 없어 플랫폼을 장기간 운영 시에 자세 및 위치 값에 심각한 오류를 가지게 된다. 따라서 본 연구에서는 INS와 USBL을 결합한 수중항법 시스템에 대한 연구를 하며, 추가로 트랙의 속도를 측정 가능한 엔코더의 정보를 이용하여 속도를 보정하는 연구를 하고자 한다.

본 연구에서는 가속도계, 각속도계, 자력계의 수학적 모델과 각 센서의 매개변수를 이용하여 관측 값을 분석하여, 항법 알고리즘 설계 시에 완성도를 높이려고 한다. 또한 수중항법 시스템을 수중트랙로봇에 장착하여 실험역 성능 평가를 하기 이전에 컴퓨터 시뮬레이션을 통하여 특정 미션과 돌발 상황에 따른 추정 값을 분석하였고, 적용된 항법 알고리즘의 유효성을 확인하였다.

마지막으로 연구의 대상 플랫폼인 수중트랙로봇에 수중항법 시스템을 적용하여 육상 및 수중 환경에서 개발된 수중항법 시스템의 성능을 검증한다.

1.3 논문 구성

본 논문의 구성은 다음과 같다. 2장에서는 본 연구에서 사용된 항법 센서들에 대한 모델과 센서의 속성을 통해 각 센서들을 모델링하였다. 3장에서는 INS 항법 알고리즘을 설계하기 위한 자세, 속도, 위치 추정과 INS 오차 모델을 설명하였다. 또한 INS와 GPS/USBL이 결합한 융합 항법 알고리즘에 대한 연구내용을 설명하였다. 그리고 제안된 융합 항법 알고리즘의 성능에 대한 실험적 실험 이전에 확인하기 위하여 시뮬레이션을 통하여 확인하였다. 4장에서는 3장에서 제안된 융합 항법 알고리즘을 적용한 수중항법 시스템의 구성 및 운영 시스템을 설명하고, 또한 수중에서 항법 실험을 수행하기 위해 제작된 플랫폼인 수중 트랙로봇에 대해 설명하였다. 5장에서는 수중트랙로봇에 탑재된 수중항법 시스템의 성능을 육상 및 실험적 실험을 통하여 확인하였으며, 이에 대한 결과를 설명하였다. 결론에서는 본 연구의 주요 결과를 나타내었다. 마지막으로 부록에는 본 연구에서 사용된 시스템 행렬들의 특성을 정리하고, 언급되는 중요한 이론에 대하여 정리하였다.

제 2 장 항법 센서의 수학적 모델

2.1 관성 측정 장치

IMU는 일반적으로 3차원 평면에서의 항법 연산을 위하여 설계된 시스템으로 가속도계와 각속도계로 구성이 된다. 추가로 자력계를 이용하여 더욱 정밀한 방향각을 추정한다. Fig. 2.1은 대표적인 IMU 제품들이며, 좌측은 MEMS 타입이며, 우측은 본 연구에서 사용된 링 레이저 자이로(Ring laser gyro, RLG) 타입이다(Kinsey et al., 2006). 위치를 보정하기 위한 INS-GPS를 통합한 시스템의 구성을 위한 IMU의 선택은 통합 시스템의 전체 성능을 결정지어주는 아주 중요한 요소이며, IMU에서 발생하는 오차 요소들에 대한 특성 분석을 통하여 항법 해의 오차 특성 분석이 가능하다.

이 절에서는 IMU에 포함되는 가속도계, 각속도계, 자력계의 수학적 모델과 각 센서의 매개변수를 이용하여 센서를 모델링 한다.



Fig. 2.1 Typical IMU sensor Family(MEMS/RLG type)

2.1.1 가속도계

가속도계 모델은 방향, 가속도 입력, 가속도계 매개변수를 사용하여 가속도계 판독 값을 모델링 한다(Titterton & Weston, 2004; Brunner et al., 2015; Hostettler & Sarkka, 2016; MathWorks, 2018). 가속도계 모델을 유도 시에 사용된 매개변수를 Table 2.1에 정의하였고, Fig. 2.2는 모델 유도 방식에 대한 블록 다이어그램을 나타낸다.

Table 2.1 Parameters of accelerometer

Parameter	Units	Description
R_n^b	.	Orientation of sensor in local navigation coordinate system
f^n	m/s^2	Acceleration of sensor in local navigation coordinate system
f^b	m/s^2	Accelerometer measurement of the sensor in the sensor body coordinate system
$f_{\max}$	m/s^2	Maximum sensor reading
f_{RS}	$(m/s^2)/LSB$	Resolution of sensor measurements
δf	m/s^2	Constant sensor offset bias
f_M	%	Sensor axes skew
f_D	$(m/s^2)/\sqrt{Hz}$	Power spectral density of sensor noise
f_B	m/s^2	Instability of the bias offset
f_W	$(m/s^2)\sqrt{Hz}$	Integrated white noise of sensor
$f_{\delta T}$	$(m/s^2)/^{\circ}C$	Sensor bias from temperature
$f_{T_{SF}}$	$\%/^{\circ}C$	Scale factor error from temperature
f_T	$^{\circ}C$	Operating temperature of the sensor in degrees Celsius
f_{Hz}	Hz	Sample rate of the sensor

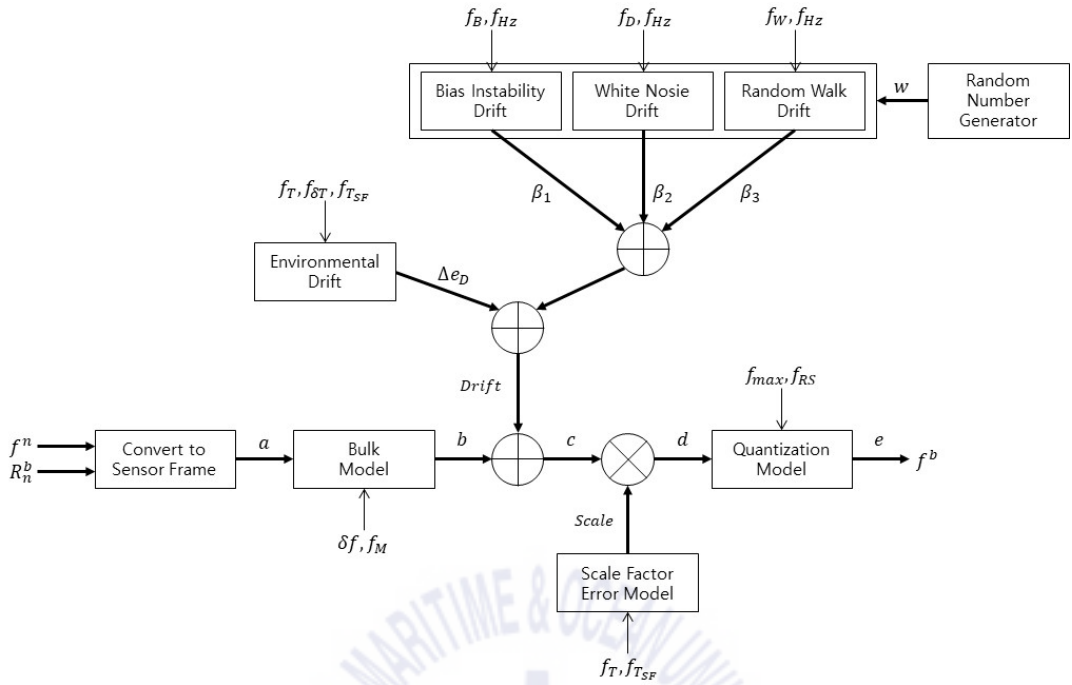


Fig. 2.2 Block diagram of accelerometer

Convert to Sensor Frame

식(2.1)은 지면에 대한 가속도 f^n 을 사용하여 로컬 프레임에서 센서 프레임으로 변환한다.

$$a = R_n^b f^n \quad (2.1)$$

방향이 사원수 형식으로 입력되면 처리 전에 회전 행렬로 변환한다.

Bulk Model

식(2.2)의 센서 프레임의 지면에 대한 가속도 a 는 벌크 모델을 통과하여 축 오정렬 및 바이어스를 추가한다.

$$b = \left(\begin{bmatrix} 1 & \frac{f_{M_2}}{100} & \frac{f_{M_3}}{100} \\ \frac{f_{M_1}}{100} & 1 & \frac{f_{M_3}}{100} \\ \frac{f_{M_1}}{100} & \frac{f_{M_2}}{100} & 1 \end{bmatrix} (a^T) \right)^T + \delta f \quad (2.2)$$

여기서 δf 는 가속도계의 매개변수이며, $f_M = [f_{M_1} f_{M_2} f_{M_3}]$ 은 가속도계 매개변수로 첫 번째, 두 번째 및 세 번째 요소를 입력하게 된다.

Bias Instability Drift

바이어스 불안정성 드리프트(Bias instability drift)는 식(2.3)과 같이 백색 잡음 바이어스로 모델링 된 후 필터링 된다.

$$\beta_1 = h_1^*(w)(f_B) \quad (2.3)$$

여기서 f_B 는 가속도계의 매개변수이며, h_1 은 식(2.4)와 같이 f_{Hz} 속성으로 정의된 필터이다.

$$H_1(z) = \frac{1}{1 + (\frac{2}{f_{Hz}} - 1)z^{-1}} \quad (2.4)$$

White Noise Drift

백색 잡음 드리프트(White noise drift)는 식(2.5)와 같이 백색 잡음 성분의 난수 요소에 표준 편차를 곱하여 모델링 된다.

$$\beta_2 = (w)(\sqrt{\frac{f_{Hz}}{2}})(f_D) \quad (2.5)$$

여기서 f_{Hz} 와 f_D 는 가속도계의 매개변수이며, w 는 속성에 의해 주어진 난수이다.

Random Walk Drift

임의 행로 드리프트(Random walk drift)는 식(2.6)과 같이 백색 잡음 성분의 난수 요소를 바이어스 한 다음 필터링하여 모델링 된다.

$$\beta_3 = h_2^*(w) \left(\frac{f_w}{\sqrt{\frac{f_{Hz}}{2}}} \right) \quad (2.6)$$

여기서 f_w 와 f_{Hz} 는 가속도계의 매개변수이며, h_2 는 식(2.7)과 같이 정의된 필터이다.

$$H_2(z) = \frac{1}{1 + z^{-1}} \quad (2.7)$$

Environmental Drift Noise

환경 드리프트 잡음(Environmental drift noise)은 식(2.8)과 같이 표준 온도와 현재 센서의 작동온도 사이의 차이에 온도 바이어스를 곱하여 모델링 된다.

$$\Delta e_D = (f_T - 25)(f_{\delta T}) \quad (2.8)$$

여기서 f_T 와 $f_{\delta T}$ 는 가속도계의 매개변수이며, 상수 25는 표준 온도에 해당한다.

Scale Factor Error Model

온도에 의한 환산 계수 오류(Scale factor error)는 식 (2.9)과 같이 모델링 된다.

$$\delta f_{sf} = 1 + \left(\frac{f_T - 25}{100} \right) (f_{T_{sf}}) \quad (2.9)$$

여기서 f_T 와 $f_{T_{sf}}$ 는 가속도계의 매개변수이며, 상수 25는 표준 온도에 해당하며, 논 논문에서는 f_T 를 25 °C로 사용하였다.

Quantization Model

양자화(Quantization)는 식(2.10)과 같이 연속 신호 모델을 먼저 포화시켜 모델링 한다.

$$e = \begin{cases} if & d > f_{\max}, d = f_{\max} \\ else if & d < -f_{\max}, d = -f_{\max} \\ else & d \end{cases} \quad (2.10)$$

그런 다음 식(2.11)과 같이 f_{RS} 를 설정한다.

$$f^b = f_{RS} \left(\text{round} \left(\frac{e}{f_{RS}} \right) \right) \quad (2.11)$$

여기서 $f_{\max}$ 는 가속도계의 매개변수이다.

2.1.2 각속도계

각속도계 모델은 방향, 가속도 및 각속도 입력, 각속계 매개변수를 사용하여 각속도계 판독 값을 모델링 한다(Titterton & Weston, 2004; Brunner et al., 2015; Hostettler & Sarkka, 2016; MathWorks, 2018). 각속도계 모델을 유도 시에 사용된 매개변수를 Table 2.2에 정의하였고, Fig. 2.3은 모델 유도 방식에 대한 블록다이어그램을 나타낸다.

Table 2.2 Parameters of gyroscope

Parameter	Units	Description
R_n^b	.	Orientation of sensor in local navigation coordinate system
f^b	m/s^2	Accelerometer measurement of the sensor in the sensor body coordinate system
ω^n	rad/s	Angular velocity of sensor in local navigation coordinate system
ω^b	rad/s	Gyroscope measurement of the sensor in the sensor body coordinate system
$\omega_{\max}$	rad/s	Maximum sensor reading
ω_{RS}	$(rad/s)/LSB$	Resolution of sensor measurements
$\delta\omega$	rad/s	Constant sensor offset bias
ω_M	%	Sensor axes skew
ω_D	$(rad/s)/\sqrt{Hz}$	Power spectral density of sensor noise
ω_B	rad/s	Instability of the bias offset
ω_W	$(rad/s)\sqrt{Hz}$	Integrated white noise of sensor
$\omega_{\delta T}$	$(rad/s)/^{\circ}C$	Sensor bias from temperature
$\omega_{T_{SF}}$	$\%/^{\circ}C$	Scale factor error from temperature
ω_T	$^{\circ}C$	Operating temperature of the sensor in degrees Celsius
ω_{Hz}	Hz	Sample rate of the sensor
$\omega_{\delta f}$	$(rad/s)/(m/s^2)$	Sensor bias from linear acceleration

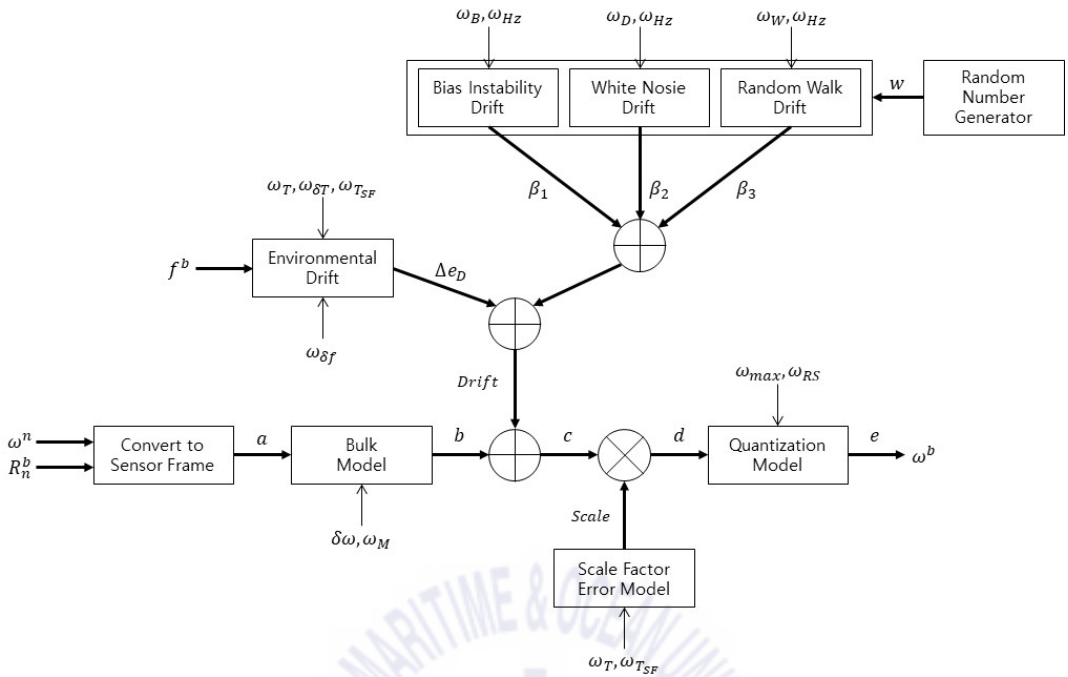


Fig. 2.3 Block diagram of gyroscope

Convert to Sensor Frame

식(2.12)는 지면에 대한 각속도 ω^n 을 사용하여 로컬 프레임에서 센서 프레임으로 변환한다.

$$a = R_n^b \omega^n \quad (2.12)$$

방향이 사원수 형식으로 입력되면 처리 전에 회전 행렬로 변환한다.

Bulk Model

식(2.13)의 센서 프레임의 지면에 대한 각속도 a 는 벌크 모델을 통과하여 축 오정렬 및 바이어스를 추가한다.

$$b = \left(\begin{bmatrix} 1 & \frac{\omega_{M_2}}{100} & \frac{\omega_{M_3}}{100} \\ \frac{\omega_{M_1}}{100} & 1 & \frac{\omega_{M_3}}{100} \\ \frac{\omega_{M_1}}{100} & \frac{\omega_{M_2}}{100} & 1 \end{bmatrix} (a^T) \right)^T + \delta\omega \quad (2.13)$$

여기서 $\delta\omega$ 는 각속도계의 매개변수이며, $\omega_M = [\omega_{M_1} \ \omega_{M_2} \ \omega_{M_3}]$ 은 각속도계 매개변수로 첫 번째, 두 번째 및 세 번째 요소를 입력하게 된다.

Bias Instability Drift

바이어스 불안정성 드리프트(Bias instability drift)는 식(2.14)와 같이 백색 잡음 바이어스로 모델링 된 후 필터링 된다.

$$\beta_1 = h_1^*(w)(\omega_B) \quad (2.14)$$

여기서 ω_B 는 각속도계의 매개변수이며, h_1 은 식(2.15)와 같이 ω_{Hz} 속성으로 정의된 필터이다.

$$H_1(z) = \frac{1}{1 + (\frac{2}{\omega_{Hz}} - 1)z^{-1}} \quad (2.15)$$

White Noise Drift

백색 잡음 드리프트(White noise drift)는 식(2.16)과 같이 백색 잡음 성분의 난수 요소에 표준 편차를 곱하여 모델링 된다.

$$\beta_2 = (w)(\sqrt{\frac{\omega_{Hz}}{2}})(\omega_D) \quad (2.16)$$

여기서 ω_{Hz} 와 ω_D 는 각속도계의 매개변수이며, w 는 속성에 의해 주어진 난수이다.

Random Walk Drift

임의 행로 드리프트(Random walk drift)는 식(2.17)과 같이 백색 잡음 성분의 난수 요소를 바이어스 한 다음 필터링하여 모델링 된다.

$$\beta_3 = h_2^*(w) \left(\frac{\omega_w}{\sqrt{\frac{\omega_{Hz}}{2}}} \right) \quad (2.17)$$

여기서 ω_w 와 ω_{Hz} 는 각속도계의 매개변수이며, h_2 는 식(2.18)과 같이 정의된 필터이다.

$$H_2(z) = \frac{1}{1 + z^{-1}} \quad (2.18)$$

Environmental Drift Noise

환경 드리프트 잡음(Environmental drift noise)은 식(2.19)와 같이 표준 온도와 현재 센서의 작동온도 사이의 차이에 온도 바이어스를 곱하여 모델링 된다.

$$\Delta e_D = (\omega_T - 25)(\omega_{\delta T}) \quad (2.19)$$

여기서 ω_T 와 $\omega_{\delta T}$ 는 각속도계의 매개변수이며, 상수 25는 표준 온도에 해당한다.

Scale Factor Error Model

온도 환산 계수 오류(Scale factor error)는 식(2.20)과 같이 모델링 된다.

$$\delta\omega_{sf} = 1 + \left(\frac{\omega_T - 25}{100}\right)(\omega_{T_{sf}}) \quad (2.20)$$

여기서 ω_T 와 $\omega_{T_{sf}}$ 는 각속도계의 매개변수이며, 상수 25는 표준 온도에 해당하며, 논 논문에서는 ω_T 를 25 °C로 사용하였다.

Quantization Model

양자화(Quantization)는 식(2.21)과 같이 연속 신호 모델을 먼저 포화시켜 모델링 한다.

$$e = \begin{cases} \text{if } d > \omega_{\max}, d = \omega_{\max} \\ \text{else if } d < -\omega_{\max}, d = -\omega_{\max} \\ \text{else } d \end{cases} \quad (2.21)$$

그런 다음 식(2.22)와 같이 ω_{RS} 를 설정한다.

$$\omega^b = \omega_{RS}(\text{round}(\frac{e}{\omega_{RS}})) \quad (2.22)$$

여기서 $\omega_{\max}$ 는 각속도계의 매개변수이다.

2.1.3 자력계

자력계 모델은 방향, 가속도 입력, 자력계의 매개변수를 사용하여 자력계 관독 값을 모델링 한다(Brunner et al., 2015; Hostettler & Sarkka, 2016; MathWorks, 2018). 자력계 모델을 유도 시에 사용된 매개변수를 Table 2.3에 정의하였고, Fig. 2.4는 모델 유도 방식에 대한 블록다이어그램을 나타낸다. 연철과 경철 효과(Soft and Hard Iron effects)는 부록에 서술하였다.

Table 2.3 Parameters of magnetometer

Parameter	Units	Description
R_n^b	.	Orientation of sensor in local navigation coordinate system
m^n	μT	Magnetic field vector of sensor in local navigation coordinate system
m^b	μT	Magnetometer measurement of the sensor in the sensor body coordinate system
$m_{\max}$	μT	Maximum sensor reading
m_{RS}	$\mu T/LSB$	Resolution of sensor measurements
δm	μT	Constant sensor offset bias
m_M	%	Sensor axes skew
m_D	$\mu T/\sqrt{Hz}$	Power spectral density of sensor noise
m_B	μT	Instability of the bias offset
m_W	$\mu T/\sqrt{Hz}$	Integrated white noise of sensor
$m_{\delta T}$	$\mu T/^{\circ}C$	Sensor bias from temperature
$m_{T_{SF}}$	%/^{\circ}C	Scale factor error from temperature
m_T	$^{\circ}C$	Operating temperature of the sensor in degrees Celsius
m_{Hz}	Hz	Sample rate of the sensor

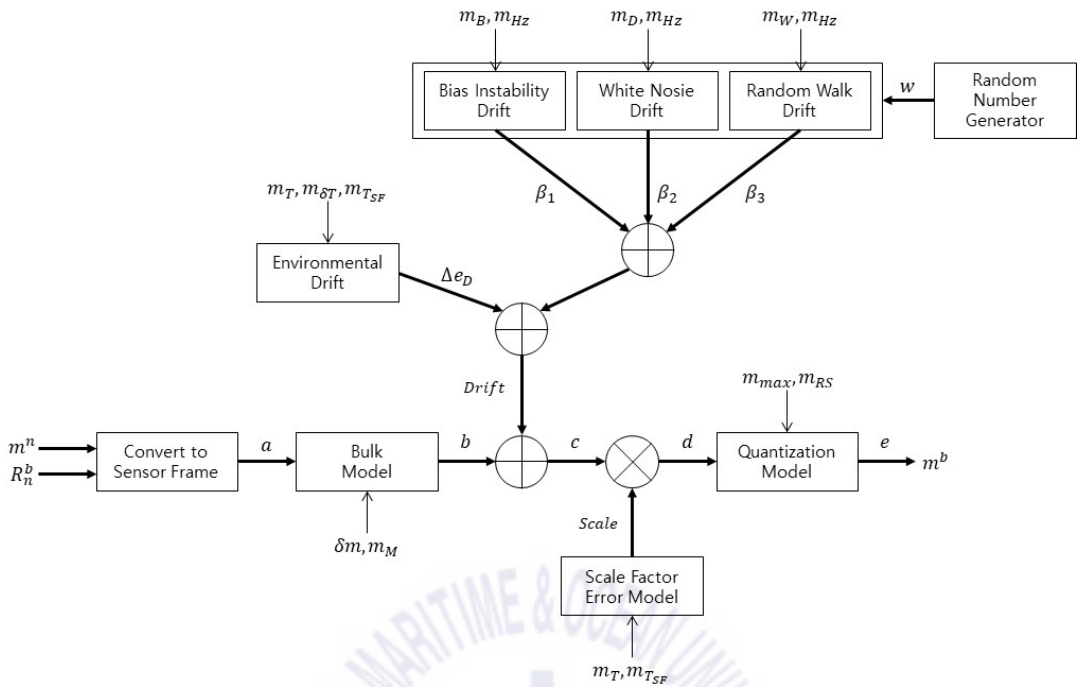


Fig. 2.4 Block diagram of magnetometer

Convert to Sensor Frame

식(2.23)은 지면에 대한 자력 m^n 을 사용하여 로컬 프레임에서 센서 프레임으로 변환한다.

$$a = R_n^b m^n \quad (2.23)$$

방향이 사원수 형식으로 입력되면 처리 전에 회전 행렬로 변환한다.

Bulk Model

식(2.24)의 센서 프레임의 지면에 대한 자력 a 는 벌크 모델을 통과하여 축 오정렬 및 바이어스를 추가한다.

$$b = \left(\begin{bmatrix} 1 & \frac{m_{M_2}}{100} & \frac{m_{M_3}}{100} \\ \frac{m_{M_1}}{100} & 1 & \frac{m_{M_3}}{100} \\ \frac{m_{M_1}}{100} & \frac{m_{M_2}}{100} & 1 \end{bmatrix} (a^T) \right)^T + \delta m \quad (2.24)$$

여기서 δm 는 자력계의 매개변수이며, $m_M = [m_{M_1} m_{M_2} m_{M_3}]$ 은 자력계 매개변수로 첫 번째, 두 번째 및 세 번째 요소를 입력하게 된다.

Bias Instability Drift

바이어스 불안정성 드리프트(Bias instability drift)는 식(2.25)와 같이 백색 잡음 바이어스로 모델링 된 후 필터링 된다.

$$\beta_1 = h_1^*(w)(m_B) \quad (2.25)$$

여기서 m_B 는 자력계의 매개변수이며, h_1 은 식(2.26)과 같이 m_{Hz} 속성으로 정의된 필터이다.

$$H_1(z) = \frac{1}{1 + (\frac{2}{m_{Hz}} - 1)z^{-1}} \quad (2.26)$$

White Noise Drift

백색 잡음 드리프트(White noise drift)는 식(2.27)과 같이 백색 잡음 성분의 난수 요소에 표준 편차를 곱하여 모델링 된다.

$$\beta_2 = (w)(\sqrt{\frac{m_{Hz}}{2}})(m_D) \quad (2.27)$$

여기서 m_{Hz} 와 m_D 는 자력계 매개변수 속성이며, w 는 속성에 의해 주어진 난수이다.

Random Walk Drift

임의 행로 드리프트(Random walk drift)는 식(2.28)과 같이 백색 잡음 성분의 난수 요소를 바이어스 한 다음 필터링하여 모델링 된다.

$$\beta_3 = h_2^*(w) \left(\frac{m_w}{\sqrt{\frac{m_{Hz}}{2}}} \right) \quad (2.28)$$

여기서 m_w 와 m_{Hz} 는 자력계의 매개변수이며, h_2 는 식(2.29)와 같이 정의된 필터이다.

$$H_2(z) = \frac{1}{1 + z^{-1}} \quad (2.29)$$

Environmental Drift Noise

환경 드리프트 잡음(Environmental drift noise)는 식(3.30)과 같이 표준 온도와 현재 센서의 작동온도 사이의 차이에 온도 바이어스를 곱하여 모델링 된다.

$$\Delta e_D = (m_T - 25)(m_{\delta T}) \quad (2.30)$$

여기서 m_T 와 $m_{\delta T}$ 는 자력계의 매개변수이며, 상수 25는 표준 온도에 해당한다.

Scale Factor Error Model

온도 환산 계수 오류(Scale factor error)는 식(2.31)과 같이 모델링 된다.

$$\delta m_{sf} = 1 + \left(\frac{m_T - 25}{100} \right) (m_{T_{sf}}) \quad (2.31)$$

여기서 m_T 와 $m_{\delta T}$ 는 자력계의 매개변수이며, 상수 25는 표준 온도에 해당하며, 논 논문에서는 m_T 을 25 °C로 사용하였다.

Quantization Model

양자화(Quantization)는 식(2.32)와 같이 연속 신호 모델을 먼저 포화시켜 모델링 한다.

$$e = \begin{cases} \text{if } d > m_{\max}, d = m_{\max} \\ \text{else if } d < -m_{\max}, d = -m_{\max} \\ \text{else } d \end{cases} \quad (2.32)$$

그런 다음 식(2.33)과 같이 m_{RS} 를 설정한다.

$$m^b = m_{RS} \left(\text{round} \left(\frac{e}{m_{RS}} \right) \right) \quad (2.33)$$

여기서 $m_{\max}$ 는 자력계의 매개변수이다.

제 3 장 융합 항법 알고리즘

3.1 좌표계

지면좌표계(North-East-Down, NED)의 원점은 지표면에 있으며 지구와 함께 자전운동을 한다. Fig. 3.1과 같이 X축은 북쪽, Y축은 동쪽, Z축은 지구의 중심 방향을 향한다.

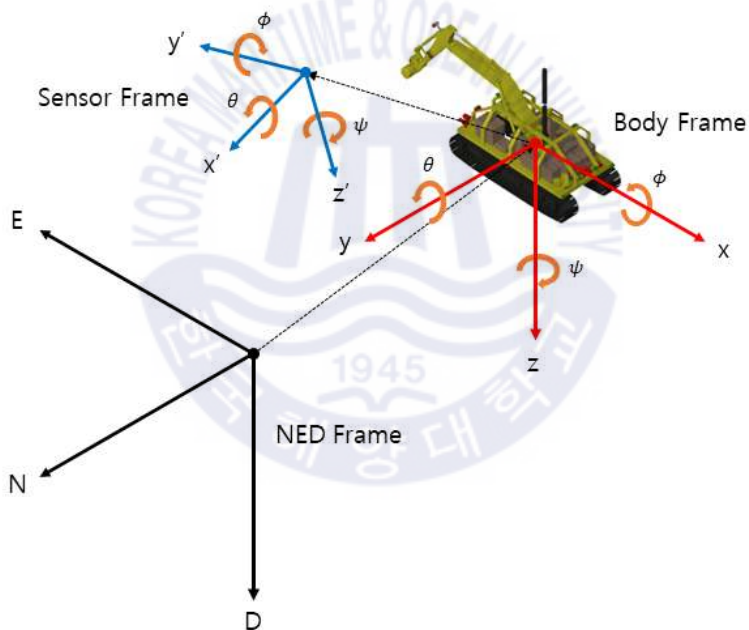


Fig. 3.1 Coordinate systems

지구가 타원형임을 고려하면 Z축이 적도면과 이루는 각은 지구 중심 위도 (Geocentric latitude) λ 가 아니라 지표 위도(Geodetic latitude) μ 이다. 그러나 일반적으로 항공기 운동특성을 해석하기 위해 사용하는 지구모델은 지구를 완

전한 구로 가정하기 때문에 Z 가 λ 가 될 수 있다(Siouris, 2004).

X , Y 및 Z 축에 대한 회전 부호는 축의 증가 방향을 따라 볼 때 시계 방향 회전이 양의 것으로 간주되는 일반적인 수학적 의미로 정의된다. 롤, 피치 및 요 회전은 일반적인 의미를 가지므로 각각 X , Y 및 Z 축에 대한 회전에 해당한다.

지면좌표계는 중력이(가속이 아님) 양수이며, 아래쪽을 향하고 중력과 정렬될 때 모든 가속도계 채널의 출력이 양수임을 의미한다. 따라서 시스템이 평평하고 똑바로 있을 때 가속도계 z 축 판독 값은 $+1g$ 이다.

3.2 INS 항법 알고리즘 설계

3.2.1 자세 추정

INS는 자이로스코프 센서의 각속도 관측 값을 활용하여 초기 자세로부터 자세 변화를 지속적으로 계산한다. 또한, 가속도 센서의 출력주기가 매우 짧을 경우 이전 관측 값으로 계산한 자세로부터 오차에 해당하는 가속도 관측 값을 제거하여 현재의 중력 벡터를 알 수 있고, 현재의 중력 벡터로부터 현재의 자세를 측정할 수 있다(Wang et al., 2015).

관성 센서를 활용하여 지상에서 움직이는 물체의 항법 시스템을 구현할 때 차량의 움직임에 의한 각속도, 차량의 가속, 감속으로 인한 가속도, 지구 자전에 의한 각속도 성분이 자이로 센서 측정치에 포함된다. 이러한 외부 요인은 관성항법시스템을 이용하여 항법 시스템을 구현할 때 오차 요인이 되므로 제거해야 한다. 먼저 자이로 센서에서 측정되는 지구 자전에 의한 각속도는 아래의 식(3.1)로 나타낼 수 있다. 여기서 ω_e 는 지구의 자전 각속도이며, φ 는 물체가 지구에서 위치한 위도를 의미한다.

$$\omega_{ie}^n = \begin{bmatrix} \omega_e \cos \varphi \\ 0 \\ -\omega_e \sin \varphi \end{bmatrix} \quad (3.1)$$

구의 형태를 갖는 지구 위에서 운동하는 물체는 지구 중심축을 기준으로 각속도 성분이 생긴다. 물체의 운동에 의한 각속도는 아래의 식(3.2)로 나타낼 수 있다.

$$\omega_{en}^n = \begin{bmatrix} \dot{\lambda} \cos \varphi \\ -\dot{\varphi} \\ -\dot{\lambda} \sin \varphi \end{bmatrix} \quad (3.2)$$

자이로 센서 측정치에서 지구 자전 및 차량의 움직임에 의한 각속도 성분을 제거하여 동체의 회전 각속도만을 계산하면 아래의 식(3.3)으로 나타낼 수 있다. 이때 ω 는 실제 각속도, ω_{ib}^b 는 자이로 센서의 측정치, ω_{bias} 는 자이로 센서의 바이어스 오차, C_n^b 는 동체 좌표계에서 항법 좌표계로의 변환행렬이다.

$$\omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \omega_{ib}^b - \omega_{bias} - C_n^b(\omega_{ie}^n + \omega_{en}^n) \quad (3.3)$$

자세를 계산하기 위한 방법으로는 오일러 각(Euler angle)을 이용하는 방법과 방향코사인 행렬(Direction cosine matrix, DCM)을 이용하는 방법, 사원수(Quaternion)를 이용하는 방법이 있다. 항법 좌표계와 동체 좌표계가 정의되었을 때, 동체에 부착된 가속도계와 각속도계를 측정하여 항법 좌표계로 변환하면 항법 좌표계에서 자세 변화를 알 수 있다. 이때 세 회전 각도를 사용하는데, 세 가지 각도를 오일러 각으로 정의하며, 동체에서 측정한 회전을 활용하여 오일러 각의 변화를 알 수 있다. 방향코사인 행렬은 세 축에 대한 회전 행렬을 활용하여 이전 자세로부터 현재 자세로의 변환을 계산할 수 있다. 사원수는 회전량을 의미하는 하나의 스칼라와 회전축을 의미하는 3개의 벡터로 이루어져 자세 변화를 계산할 수 있다(김용호, 2005). 본 논문에서는 자세 변화 오차가 적고 짐벌 락(Gimbal lock) 현상이 일어나지 않는 사원수 방식으로 자세를 계산하였다. 사원수의 구성은 아래의 식(3-4)와 같다.

$$q = [q_0 \ q_1 \ q_2 \ q_3]^T \quad (3.4)$$

계산한 동체의 각속도로부터 자세의 변화량을 계산하기 위한 사원수 미분방정식은 아래의 식(3.5)과 같다. $\dot{q}$ 는 사원수의 변량을 의미한다.

$$\dot{q}_{k+1} = \frac{1}{2} \Omega(\omega) q_k, \quad \Omega(\omega) = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_x & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \quad (3.5)$$

계산한 사원수의 변화량으로부터 현재의 사원수 자세를 계산하면 아래의 식(3-6)으로 표현할 수 있다(Wang et al., 2015).

$$q_{k+1} = q_k \left[I_{3 \times 3} \left(1 - \Delta \frac{\theta_k^2}{8} \right) + \frac{1}{2} \Omega(\omega) T_s \right] q_k \quad (3.6)$$

가속도 센서의 출력주기가 충분히 짧은 경우 이전시간의 자세와 가속도 센서의 측정치가 출력되는 시간의 자세는 큰 변화가 없다. 따라서 현재 자세로부터 측정치의 중력 벡터를 계산할 수 있고, 중력 벡터로부터 현재의 롤, 피치 각을 알 수 있다. 먼저 가속도 측정치로부터 중력 벡터를 계산할 때 발생하는 오차를 알기 위한 오차 함수를 활용한다. 식(3.7)는 오차 함수를 나타내며 g 는 중력 벡터, a_b 는 동체에서 측정한 가속도 측정치를 정규화시킨 가속도 벡터, a_{bias} 는 가속도 센서의 바이어스 오차로 식(3.8)과 같다.

$$f_{error}(q, g, a) = C_b^n(q)g = (C_b^n(q))^T g - (a_b - a_{bias}) \quad (3.7)$$

$$g = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad a_b = \frac{1}{\sqrt{a_x^2 + a_y^2 + a_z^2}} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \quad (3.8)$$

측정된 오차를 현재 자세로부터 제거하면 현재의 자세에 대해 알 수 있다. 오차 함수로부터 측정한 가속도 오차를 활용하여 사원수의 오차를 제거하기 위해 경사 하강법(Gradient descent method)을 활용한다(Wang et al., 2015). 이전 시간의 사원수 자세에서 현재 시간의 오차를 제거하는 식은 아래의 식(3.9), (3.10), (3.11)으로 나타낼 수 있다

$$q_{z(k+1)} = q_k - \mu \left(\frac{\nabla f_{error}(q, g, a)}{\| \nabla f_{error}(q, g, a) \|} \right) \quad (3.9)$$

$$\nabla f_{error}(q, g, a) = J^T(q) f_{err}(q, g, a) \quad (3.10)$$

$$J(q) = 2 \begin{bmatrix} -q_2 & q_3 & -q_0 q_1 \\ q_1 & q_0 & q_3 q_2 \\ q_0 & -q_1 & -q_2 q_3 \end{bmatrix} \quad (3.11)$$

식(3.9)의 μ 는 경사 하강법을 위한 결절점(step size)이다. 식(3.10)은 오차 함수의 경사(Gradient)를 계산한 식이며, 식(4-11)의 $J(q)$ 는 사원수에 대한 자코비안 행렬이다.

자이로 센서로 계산한 자세 정보는 오차가 지속적으로 누적되어 장시간 사용 시 발산하는 단점이 있다. 가속도 센서로 계산한 자세 정보는 순간적인 가속도 상황에서 부정확해지는 단점이 있다. 상호 보완적인 두 센서의 자세 관측 값을 칼만 필터를 활용하여 융합한다. 두 관측 값을 융합하기 위해 짧은 시간동안 정확한 자이로 센서로 계산한 자세를 예측값, 발산하지 않는 가속도 센서로 계

산한 자세를 관측 값으로 사용한다(이기원, 2002). 아래의 Fig. 3.2는 자세를 계산하는 칼만 필터의 구조를 나타낸 식이다.

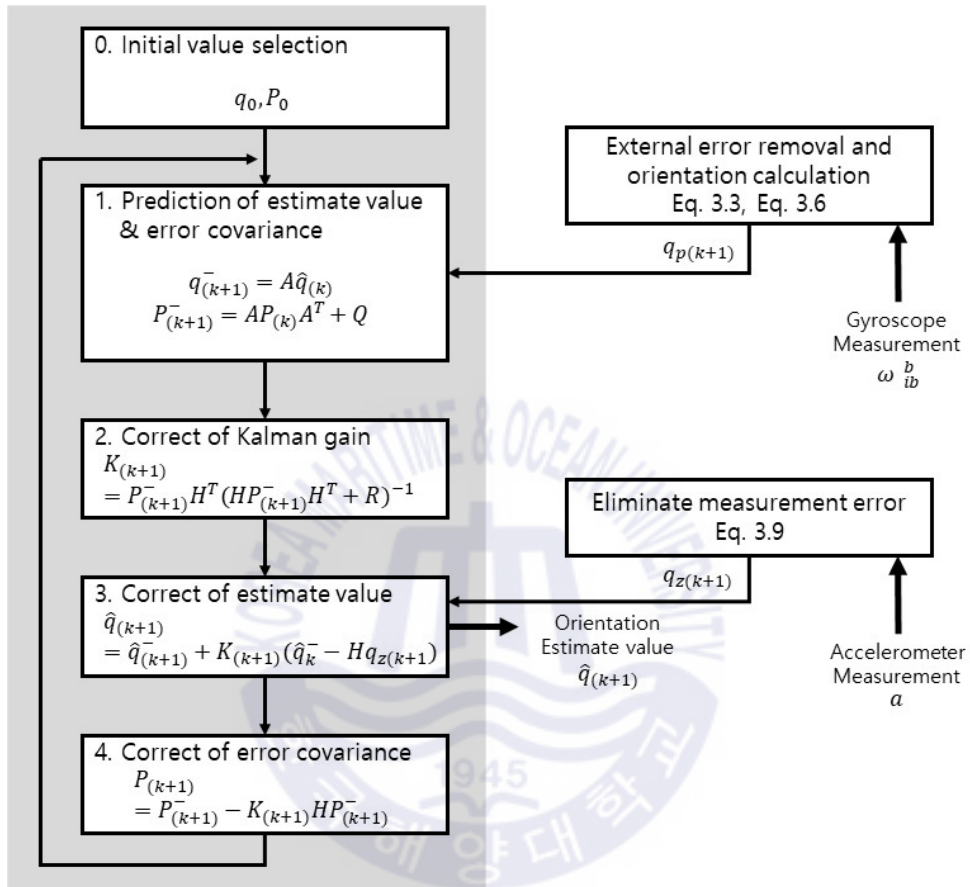


Fig. 3.2 Kalman filter for orientation calculation

칼만 필터에서 추정하는 상태 변수는 사원수 자세로 설정하였고, 이때 측정치 모델 H 는 아래의 식(3.12)과 같이 나타낼 수 있다. 상태전이 행렬과 관측 값에 대한 공분산 행렬은 각 자세 관측 값의 오차 공분산으로 설정하였다.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.12)$$

갱신된 자세에 대하여 수치적 안정성을 얻기 위해서는 식(3.13)과 같이 정규화 과정을 수행한다.

$$q^* = \frac{q}{\|q\|} \quad (3.13)$$

최종적으로 계산된 사원수로부터 현재의 오일러 각 자세와 좌표변환행렬을 계산하기 위해서는 아래의 식(3.14)~(3.17)과 같은 좌표변환 행렬을 활용해 계산해야 한다.

$$C_b^n = \begin{bmatrix} q_0^2 + q_1^2 + q_2^2 + q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (3.14)$$

$$\theta = -\tan^{-1}\left(\frac{-\sin\theta}{\cos\theta}\right) \quad (3.15)$$

$$= -\tan^{-1}\left(\frac{2(q_1q_3 - q_0q_2)}{\sqrt{(2(q_2q_3 + q_0q_1))^2 + (q_0^2 - q_1^2 - q_2^2 + q_3^2)^2}}\right)$$

$$\phi = \tan^{-1}\left(\frac{\sin\phi}{\cos\phi}\right) = \tan^{-1}\left(\frac{2(q_2q_3 + q_0q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\right) \quad (3.16)$$

$$\psi = \tan^{-1}\left(\frac{\sin\psi}{\cos\psi}\right) = \tan^{-1}\left(\frac{2(q_1q_2 - q_0q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2}\right) \quad (3.17)$$

관성 항법으로 동체의 자세를 구하기 위해서는 동체의 초기 자세에 대한 정보가 필요하다. 초기 자세를 구하는 초기 정렬은 개략 정렬(Coarse alignment)과 칼만 필터와 같은 최적의 추정자를 이용하여 오차를 추정 및 보정하는 정밀 정렬(Fine alignment)로 구분된다[26]. 본 논문에서는 개략 정렬방법으로 초기 자세를 계산하였다.

개략 정렬은 정지 상태에서 가속도 센서 및 자이로 센서의 측정치를 활용하여 초기 자세를 구하는 과정이다. 정지 상태에서 가속도는 중력가속도 성분만을 가지고 있다. 식(3.18)과 식(3.19)와 같이 중력의 방향은 항상 지면을 향하므로 중력 벡터의 방향으로부터 롤, 피치 각에 대한 초기 자세를 계산할 수 있다 (김용호, 2005).

$$\phi = \tan^{-1}\left(\frac{a_y}{a_z}\right) \quad (3.18)$$

$$\theta = \tan^{-1}\left(\frac{a_x}{\sqrt{a_y^2 + a_z^2}}\right) \quad (3.19)$$

중력 벡터는 지면을 향하므로 가속도 관측 값만으로는 요 각에 대한 정보를 계산할 수 없다. 정지 상태의 요 각을 계산하기 위해서 지구 자전 각속도를 각속도계로부터 측정한다. 각속도계에서 측정되는 지구 자전 각속도 ω_{ib}^b 는 항법 좌표계의 지구 자전 각속도 ω_{ib}^n 로부터 좌표변환 과정을 통해 다음과 같이 계산한다. C_1 은 롤의 좌표변환 행렬, C_2 는 피치의 좌표변환 행렬이다.

$$\omega_{ib}^b = C_n^b \omega_{ib}^n = C_1 C_2 \omega_{ie}^n \quad (3.20)$$

식(3.20)을 정리하면 아래의 식(3.21)로 나타낼 수 있다.

$$C_1^T \omega_{ib}^b = C_2 \omega_{ib}^n \quad (3.21)$$

C 의 성분은 앞에서 구한 롤, 피치 각을 이용해 구할 수 있으며, 이때 식 (3.22)로부터 계산한 요 각은 식(3.23)와 같다.

$$C_1^T \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = C_2 \omega_{ib}^n \quad (3.22)$$

$$\varphi = -\tan^{-1}\left(\frac{\omega_2}{\omega_1}\right) \quad (3.23)$$

3.2.2 속도 추정

INS에서 식(3.24)과 같이 속도를 계산할 수 있다.

$$\dot{V}^n = C_b^n a_b - \left[2(w_{ie}^n \times) + (w_{en}^n \times) \right] V^n + g^n \quad (3.24)$$

위 식에서 V^n 은 항법 좌표계에서 차량의 속도, g^n 은 항법 좌표계에서 중력 가속도, $(w_{ie}^n \times)$ 와 $(w_{en}^n \times)$ 은 각각 w_{ie}^n 와 w_{en}^n 의 반대칭 행렬이다. 가속도계는 지표면 방향으로 중력가속도를 받기 때문에 정확한 속도 측정을 위해서는 중력 가속도 성분을 제거한 가속도 관측 값이 필요하다. 지구의 단면은 완벽한 구가 아닌 타원형이고, 지구 자전 각속도로 인한 원심력으로 인해 위도에 따라 중력 가속도의 차이가 존재한다. 위도에 맞는 중력가속도를 알기 위해 중력 모델로부터 차량의 중력가속도를 계산한다(Madden, 2006). 위도에 따른 현재 중력에 대한 중력 모델을 지구에 관한 변수들을 대입하여 정리하면 아래의 식(3.25)로 나타낼 수 있다.

$$g^n = -9.7803253359 \frac{1 + 0.00193185265241 \sin^2 \varphi}{\sqrt{1 - 0.00669437999014 \sin^2 \varphi}} \quad (3.25)$$

x_1, x_2, x_3 을 원소로 갖는 3차원 벡터 X 에 대한 반대칭 행렬은 아래의 수식 (3.26)과 같이 만들 수 있다.

$$(X \times) = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix} \quad (3.26)$$

3.2.3 위치 추정

지표면 근처를 이동하는 항체의 이동은 항법 좌표계에서 각 축으로의 속력으로부터 구할 수 있다. 앞에서 계산한 속도로부터 현재의 위도, 경도, 고도에 대한 변화량을 계산하면 아래의 식(3.27)로 나타낼 수 있다.

$$\dot{\varphi} = \frac{v_n}{M+h}, \quad \dot{\lambda} = \frac{v_e}{(N+h)\cos\varphi}, \quad \dot{h} = -v_d \quad (3.27)$$

식(3.27)에서 v_N, v_E, v_D 는 지면좌표계에서 각 축으로의 속력을 의미하고, h 는 고도를 의미한다. M 은 위도에 따른 지구의 평균 곡률반경으로 아래의 식(3.28)로부터 계산할 수 있다.

$$M = \frac{r_e(1-e^2)}{(1-e^2\sin^2\varphi)^{3/2}} \quad (3.28)$$

식(3.28)에서 r_e 는 지구 장반경을 의미하고, e 는 지구의 이심률로 식(3.29)와 같이 지구의 장반경과 단반경으로부터 계산할 수 있다. 이때 r_p 는 지구 단반경을 의미한다.

$$e = \left(1 - \frac{r_p^2}{r_e^2}\right)^{1/2} \quad (3.29)$$

N 은 지구의 횡단면 곡률반경으로 아래의 식(3.30)으로 나타낼 수 있다.

$$N = \frac{r_e}{(1 - e^2 \sin^2 \varphi)^{1/2}} \quad (3.30)$$

3.2.4 INS 오차 모델

INS는 구현에 활용된 센서들의 오차, 측정한 초기 자세의 오차, 계산상의 오차 등으로 인해 항법 오차가 발생한다. 항법 오차를 보상하기 위해서는 오차들에 대한 정보가 필요하다. 따라서 오차들을 정의하고 보정하기 위한 오차 모델은 정밀한 INS 항법 결과를 위해 필수적이다.

위치 오차 모델

위치 변화에 대한 미분방정식인 식(3.27)을 1차 테일러 급수로 미분 선형화하여 나타낼 수 있다. 식(3.27)에서 오차에 대한 성분을 분류하면 속도에 대한 오차, 위도에 대한 오차, 고도에 대한 오차, 지구의 평균 곡률반경의 오차, 지구의 횡단면 곡률반경의 오차로 구분할 수 있다. 오차 요인들에 대하여 정리한 결과는 아래의 식(3.31), 식(3.32), 식(3.33)과 같다(Sveinsson, 2012).

$$\dot{\delta\varphi} = \frac{-v_n M'}{(M+h)^2} \delta\varphi - \frac{v_n}{(M+h)^2} \delta\varphi + \frac{1}{M+h} \delta v_n \quad (3.31)$$

$$\delta\dot{\lambda} = \left(\tan\varphi - \frac{N'}{N+h}\right) \delta\varphi + \frac{v_e}{(N+h)^2 \cos\varphi} \delta h + \frac{1}{(N+h)^2 \cos\varphi} \delta v_e \quad (3.32)$$

$$\delta\dot{h} = -\delta v_d \quad (3.33)$$

식(3.31)에서 M' 은 위도 오차에 의한 지구의 평균 곡률반경 오차로 식(3.34)과 같이 나타낼 수 있다.

$$M' = \frac{3r_e(1-e^2)e^2\sin\varphi\cos\varphi}{(1-e^2\sin^2\varphi)^{5/2}} \quad (3.34)$$

식(3.32)의 N' 은 위도 오차에 의한 지구의 횡단면 곡률반경의 오차로 식(3.35)와 같이 나타낼 수 있다.

$$N' = \frac{r_e e^2 \sin\varphi \cos\varphi}{(1-e^2\sin^2\varphi)^{3/2}} \quad (3.35)$$

속도 오차 모델

식(3.24)의 속도 계산식에서 오차 요인은 현재 자세의 비틀림 오차, 가속도 측정치의 바이어스 오차, 지구 자전 벡터와 항체 운동에 의한 각속도 벡터의 오차, 속도 오차로 구분할 수 있다. 속도 오차 모델은 아래의 식(3.36)과 같다.

$$\begin{aligned} \delta \dot{V}^n = & C_b^n \nabla + C_b^n f_c \delta \Phi \\ & - \left[2(\delta \omega_{ie}^n \times) + (\delta \omega_{en}^n \times) \right] V^n - \left[2(\omega_{ie}^n \times) + (\omega_{en}^n \times) \right] \delta V^n \end{aligned} \quad (3.36)$$

식(3.36)에서 ∇ 는 가속도 센서의 랜덤 바이어스 오차, $\delta \Phi$ 는 자세에 대한 비틀림 오차, δV^n 은 속도 오차, $\delta \omega_{ie}^n$ 와 $\delta \omega_{en}^n$ 은 각각 지구 자전 벡터와 항체 운동에 의한 각속도 벡터의 오차를 의미한다. 이때 $\delta \omega_{ie}^n$ 와 $\delta \omega_{en}^n$ 에 대한 오차는 아래의 식(3.37)과 식(3.38)로 표현할 수 있다.

$$\omega_{ie}^n = \begin{bmatrix} -\omega_e \sin \varphi \delta \varphi \\ 0 \\ -\omega_e \cos \varphi \delta \varphi \end{bmatrix} \quad (3.37)$$

$$\omega_{en}^n = \begin{bmatrix} \lambda \left(\frac{-N' \cos \varphi}{N+h} \right) \delta \varphi + \frac{\dot{\lambda} \cos \varphi}{N+h} + \frac{1}{(N+h) \cos \varphi} \delta v_e \\ \frac{v_n M'}{(M+h)^2} \delta \varphi + \frac{v_n}{(M+h)^2} \delta h - \frac{1}{(N+h) \cos \varphi} \delta v_n \\ - \lambda \left(\tan \varphi - \frac{N'}{N+h} + \cos \varphi \right) \delta \varphi - \lambda \frac{\sin \varphi}{M+h} \delta h - \frac{\tan \varphi}{(N+h)^2} \delta v_e \end{bmatrix} \quad (3.38)$$

자세 오차 모델

자세에 대한 오차는 기준 자세로부터 발생한 비틀림 오차의 미분방정식 형태로 아래의 식(3.39)로 표현할 수 있다.

$$\dot{\Phi} = - \left(\left[\omega_{ie}^n + \omega_{en}^n \right] \times \right) \Phi - C_b^n \epsilon + \delta \left(\omega_{ie}^n + \omega_{en}^n \right) \quad (3.39)$$

관성센서 오차 모델

가속도계와 각속도계의 오차는 불규칙한 오차로 아래의 식(3.40)과 식(3.41)로 표현할 수 있다. 아래의 식에서 ∇ 는 가속도 센서의 랜덤 바이어스 오차, ϵ 은 자이로 센서의 랜덤 바이어스 오차, ω_a 와 ω_g 는 각각 가속도계와 각속도계의 백색 잡음이다.

$$\delta a_b = \nabla + \omega_a \quad (3.40)$$

$$\delta \omega_{ib}^b = \epsilon + \omega_g \quad (3.41)$$

3.3 융합 항법 알고리즘 설계

3.3.1 시스템 오차 모델

시스템 모델은 앞에서 계산한 INS의 오차 모델을 활용한다. 두 항법 시스템을 융합하기 위해 확장 칼만 필터를 활용한다. 칼만 필터에서 추정하고자 하는 상태 변수는 아래의 식(3.42)와 같다(Munguia, 2014).

$$X = [\delta P_{INS} \ \delta V_{INS} \ \delta \Phi \ \nabla \ \epsilon]^T \quad (3.42)$$

이때 δP_{INS} 는 INS의 위치 오차 벡터, δV_{INS} 는 INS의 속도 오차 벡터, $\delta \Phi$ 는 INS 자세의 비틀림 오차, ∇ 는 가속도 센서의 랜덤 바이어스 오차, ϵ 은 자이로 센서의 랜덤 바이어스 오차를 의미한다. 칼만 필터의 상태 방정식은 아래의 식(3.43)과 식(3.44)와 같다(박준기, 2014; Karamat et al., 2015).

$$X(k) = \dot{F}_{INS}(k) + w_{INS} \quad (3.43)$$

$$F_{INS}(k) = \begin{bmatrix} F_{pp} & F_{pv} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ F_{vp} & F_{vv} & F_{va} & F_{vb_a} & 0_{3 \times 3} \\ F_{ap} & F_{av} & F_{aa} & 0_{3 \times 3} & F_{ab_g} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \quad (3.44)$$

이때 공정잡음 w_{INS} 는 평균이 0이며, 공분산이 Q_{INS} 인 각속도계와 가속도계의 백색 잡음으로 아래의 식(3.45)와 같이 표현할 수 있다.

$$w_{INS} = [0_{1 \times 3} \ w_{a_n} \ w_{a_e} \ w_{a_d} \ w_{g_n} \ w_{g_e} \ w_{g_d} \ 0_{1 \times 6}] \quad (3.45)$$

F_{INS} 는 앞에서 유도한 오차 모델로부터 식(3.46) ~ 식(3.55)와 같이 나타낼 수 있다.

$$F_{pp} = \begin{bmatrix} \frac{-v_n M'}{(M+M'+h)^2} & 0 & \frac{-v_n}{(M+h)^2} \\ \frac{-v_e}{(N+h)^2} \left(\frac{N'}{\cos\varphi} - (N+h)\tan\varphi \right) & 0 & \frac{-v_e}{(N+h)^2 \cos\varphi} \\ 0 & 0 & 0 \end{bmatrix} \quad (3.46)$$

$$F_{pv} = \begin{bmatrix} \frac{1}{M+h} & 0 & 0 \\ 0 & \frac{1}{(N+h)\cos\varphi} & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad (3.47)$$

$$F_{vp} = \begin{bmatrix} 2w_e v_e \cos\varphi - \frac{v_e^2}{(N+h)\cos\varphi} \left(\frac{\sin\varphi}{N+h} N' - \frac{1}{\cos\varphi} \right) + \frac{v_n v_d M'}{M+h} & 0 & \frac{v_n v_d}{(M+h)^2} - \frac{v_e^2 \tan\varphi}{(N+h)^2} \\ 2w_e (v_d \sin\varphi - v_n \cos\varphi) + \frac{v_n v_e}{(N+h)\cos\varphi} \left(\frac{\sin\varphi}{N+h} N' - \frac{1}{\cos\varphi} \right) + \frac{v_e v_d N'}{(N+h)^2} & 0 & \frac{v_e (v_n \tan\varphi + v_d)}{(N+h)^2} \\ -\frac{v_n^2}{(m+h)^2} - 2w_e v_e \sin\varphi - \frac{v_e^2 N'}{(N+h)^2} & 0 & \frac{v_e v_d}{(N+h)^2} - v \end{bmatrix} \quad (3.48)$$

$$F_{vv} = \begin{bmatrix} -\frac{v_d}{M+h} & -\frac{v_e \tan\varphi}{N+h} & 0 \\ 0 & \frac{-v_n \tan\varphi - v_d}{N+h} & 0 \\ \frac{v_n}{M+h} & \frac{v_e}{N+h} & 0 \end{bmatrix} \quad (3.49)$$

$$F_{va} = C_b^n a_b \quad (3.50)$$

$$F_{vb_a} = C_n^b \quad (3.51)$$

$$F_{ap} = \begin{bmatrix} -w_e \sin \varphi - \frac{v_e N'}{(N+h)^2} & 0 & \frac{-v_e}{N+h} \\ \frac{v_n M'}{M+h} & 0 & \frac{v_n}{(M+h)^2} \\ -w_e \cos \varphi + \frac{v_e}{(N+h) \cos \varphi} \left(\frac{\sin \varphi}{N+h} N' - \frac{1}{\cos \varphi} \right) & 0 & \frac{v_e \tan \varphi}{(N+h)^2} \end{bmatrix} \quad (3.52)$$

$$F_{av} = \begin{bmatrix} 0 & \frac{1}{N+h} & 0 \\ -\frac{1}{M+h} & 0 & 0 \\ 0 & -\frac{\tan \varphi}{N+h} & 0 \end{bmatrix} \quad (3.53)$$

$$F_{aa} = \begin{bmatrix} 0 & w_e \sin \varphi + v_e \frac{\tan \varphi}{N+h} & \frac{-v_n}{M+h} \\ -w_e \sin \varphi - v_e \frac{\tan \varphi}{N+h} & 0 & -w_e \cos \varphi + \frac{v_e}{N+h} \\ \frac{v_n}{M+h} & w_e \cos \varphi + \frac{v_e}{N+h} & 0 \end{bmatrix} \quad (3.54)$$

$$F_{ab_g} = C_b^n \quad (3.55)$$

3.3.2 측정 오차 모델

약 결합 방식에서 두 항법 시스템의 측정 오차는 위치와 속도 정보의 차이로 구할 수 있다. 이때 H 는 측정행렬로 식(4-57)과 같이 나타낼 수 있다.

$$z = \begin{bmatrix} P_{INS} \\ V_{INS} \end{bmatrix} - \begin{bmatrix} P_{GPS \parallel USBL} \\ V_{GPS \parallel USBL} \end{bmatrix} = HX + w_m \quad (3.56)$$

$$H = \begin{bmatrix} I_{6 \times 6} & 0_{6 \times 9} \end{bmatrix} \quad (3.57)$$

3.4 융합 항법 알고리즘 구성

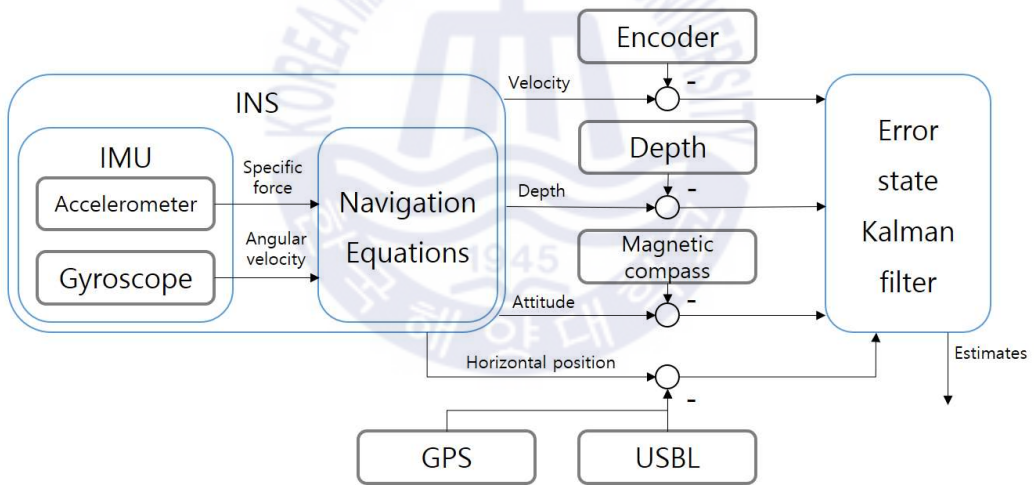


Fig. 3.3 Block diagram of the navigation system

제안된 융합 항법 알고리즘의 블록선도는 Fig. 3.3과 같다. 앞선 설명과 같이 IMU를 통해 INS를 구성하며, INS에서 도출되는 속도는 트랙의 엔고더로 보정, 위치는 GPS 또는 USBL(수평)과 수심 센서(수직)로 보정, 방향각은 자력계로 보정한다. USBL의 위치 보정 단계에서 측정 위치의 정확도가 낮으면 엔코더 정보로 속도를 보정하는 단계의 가중치를 가변적으로 높여준다. 반면 USBL의 위

치 정확도가 높아지면 위치 보정 단계의 가중치를 높이고 속도 보정 단계의 가중치를 가변적으로 낮추어준다. 이와 같이 보정 단계는 USBL의 위치 정확도에 따라 위치/속도 보정 단계의 가중치를 가변적으로 변경해준다. 마지막으로 확장 칼만필터를 거쳐 최종 결과 값(속도, 위치, 방향각)을 도출한다.

추가적으로 융합 항법의 수평 위치 드리프트는 추정된 지구 고정 속도 (즉, 북 및 동 속도)의 오차에 의해 결정된다. 이 오류의 주요 원인은 선체 고정 속도 오류 및 헤딩 오류이다. 선체 고정 속도 오류 중에 고주파 속도 에러는 IMU에 의해 추정된다. 엔코더를 통한 속도 보정이 없이는 가장 정확한 INS 조차도 짧은 시간 후에 큰 속도 불확실성을 가질 것이다. 이는 일반적 수중항법에서 사용되는 DVL의 저주파 오차의 단점을 해소할 수 있다.

3.5 융합 항법 시뮬레이션

융합 항법의 성능에 대한 실험적 실험 이전에 확인하기 위하여 시뮬레이션을 통하여 확인하고자 한다. 시뮬레이터는 Matlab을 활용하여 개발되었다. 이 절에서 설명하는 시뮬레이션에 사용된 센서는 IMU의 가속도계와 각속도계, 자력계, GPS, 엔코더 총 5개다. Table 3.1은 시뮬레이션에 사용된 IMU의 주요 파라미터이며, Table 3.2는 자력계의 주요 파라미터이며, Table 3.3은 GPS의 주요 파라미터이다. 모든 센서의 파라미터는 본 논문에서 개발한 수중항법 시스템에 사용된 센서의 제원을 바탕으로 선정되었다. 사용된 센서의 세부 제원은 Table 4.4와 같다.

첫 번째로 시뮬레이션의 입력은 선체의 가속도, 각속도, 방향각이 사용되며, 이를 바탕으로 선체가 이동하게 될 경로를 생성하며 이는 나중에 항법 연산의 위치 결과와 비교하게 된다. 두 번째로 2장에서 설명한 IMU의 모델을 이용하여 선체의 가속도, 각속도, 방향각을 입력하여 센서에서 측정되는 측정치를 생성한다. 추가로 GPS의 모델을 이용하여 생성된 이동 경로를 입력으로 GPS의 위치 측정치와 엔코더에서 측정되는 속도 측정치를 생성한다. 마지막으로 3장에서 설명한 융합 항법을 이용하여 앞서 생성된 가속도계, 각속도계, 자력계, GPS, 엔코더 정보를 입력으로 최종 결과(위치, 속도, 방향각)를 도출한다. 시뮬레이션은 서로 다른 입력으로 3번 실시하여 융합 항법의 성능을 확인하였다.

Table 3.1 Specifications of the IMU used in the simulation

Index	Accelerometer	Gyroscope
Measurement Range	362.846 m/s^2	18.7448 rad/s
Resolution	3.6576e-12 m/s^2	1.2e-16 rad/s
Constant Bias	0.0098 m/s^2	0.8727 rad/s
Bias Instability	0.0001 m/s^2	4.48e-6 rad/s
Noise Density	8.3357e-4 $m/s^2/\sqrt{Hz}$	0.0785 $rad/s/\sqrt{Hz}$
Update Rate	100 Hz	

Table 3.2 Specifications of the magnetometer used in the simulation

Index	Magnetometer
Measurement Range	80 μT
Resolution	0.05 μT
Constant Bias	1 μT
Noise Density	0.0239 $\mu T/\sqrt{Hz}$
Update Rate	100 Hz

Table 3.3 Specifications of the GPS used in the simulation

Index	GPS
Horizontal Position Accuracy	1.3 m
Vertical Position Accuracy	1.9 m
Horizontal Velocity Accuracy	0.8 cm/s
Vertical Velocity Accuracy	1.3 cm/s
Global Position Noise Decay Factor	0.5
Update Rate	1 Hz

3.5.1 S자 궤적 항법 시뮬레이션 (Case 1)

본 시뮬레이션을 수행하는 이유는 임의의 이동 궤적에 따라 항법 결과가 어느 정도의 오차를 갖는지 확인하기 위해 수행하였다. Fig. 3.4는 IMU 모델의 가속도, 각속도 입력으로 사용될 선체의 각속도, 가속도 정보이다. 각속도는 초기에 $[0 \ 0 \ 0.2] \text{ rad/s}$ 이며, 12초에 $[0 \ 0 \ 0] \text{ rad/s}$ 으로 변경되며, 36초에 $[0 \ 0 \ -0.2] \text{ rad/s}$ 로 변경되며, 최종 각속도도 이전과 같다. 가속도는 초기에 $[0.1 \ 0 \ 0] \text{ m/s}^2$ 이며, 12초에 $[-0.035 \ 0.035 \ 0] \text{ m/s}^2$ 으로 변경되며, 등가속 운동 후 36초에 가속도가 변경되어 최종 가속도는 $[-0.07 \ -0.07 \ 0] \text{ m/s}^2$ 이 된다. 이에 따라 선체의 방향각은 Fig. 3.5와 같이 산출되며, 이는 항법 연산의 방향각 결과와 비교될 것이다. 방향각이 초기 $[0 \ 0 \ 0]^\circ$ 에서 12초까지 Z축이 증가하며 36초까지 $[0 \ 0 \ 135]^\circ$ 를 유지한 뒤에 Z축이 감소하여 최종 방향각은 $[0 \ 0 \ -135]^\circ$ 가 된다. 앞서 설명한 데이터로 선체의 이동 경로를 산출하면 Fig. 3.6과 같으며, S자 형상으로 선체가 이동하였다.

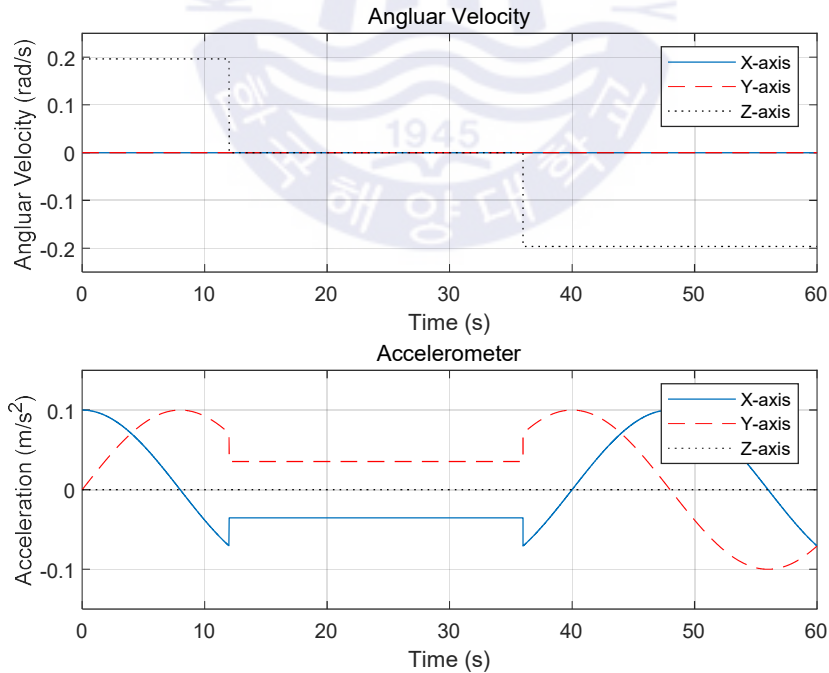


Fig. 3.4 Angular velocity/acceleration input of IMU model (Case 1)

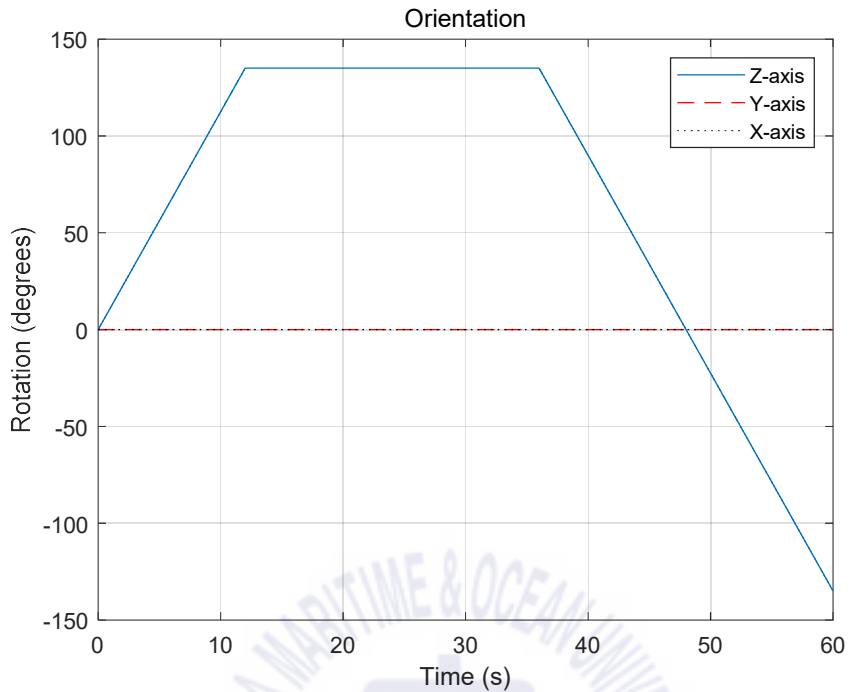


Fig. 3.5 Orientation input of IMU model (Case 1)

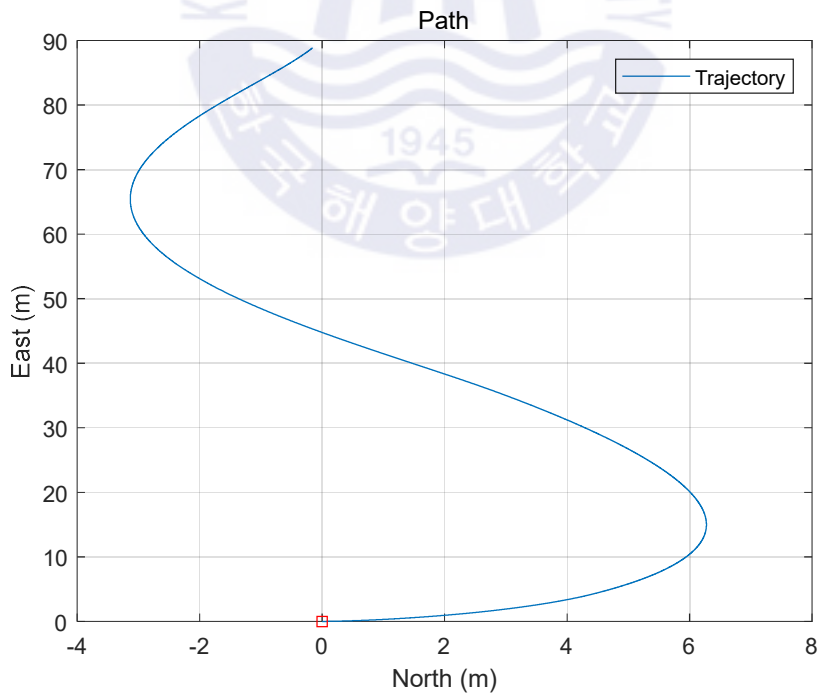


Fig. 3.6 Create a path (Case 1)

Fig. 3.7은 앞서 2장에서 설명한 IMU 센서 모델을 이용하여 생성된 각속도와 가속도, 방향각을 입력으로 출력된 센서 측정치 결과이다. 각 센서들의 파라미터에 따라 센서에서 계측되는 측정치를 산출하며, 가속도에는 Z축으로 중력 성분이 추가되며, 센서들에 센서 노이즈가 추가된 것이 확인된다.

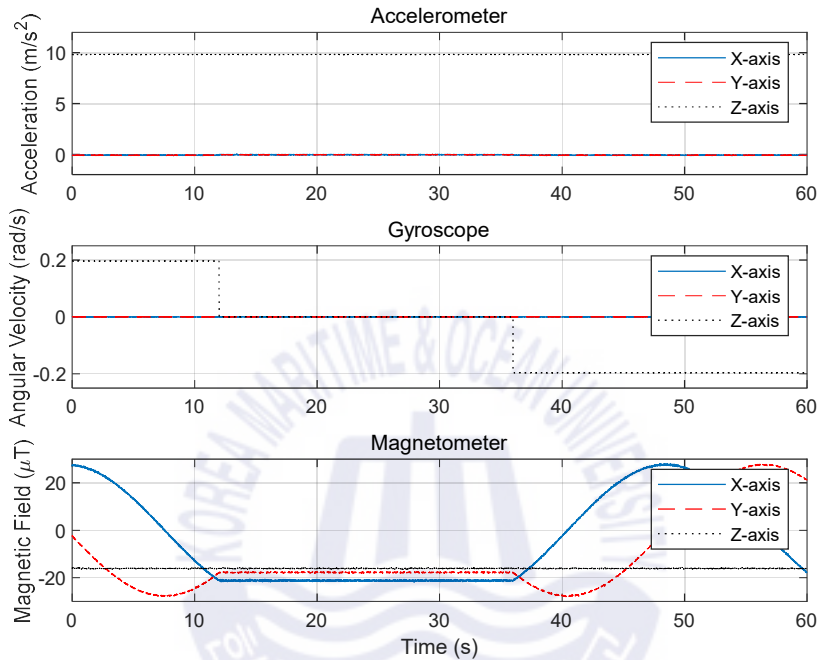


Fig. 3.7 Navigation sensor measurements derived from the IMU model (Case 1)

항법 연산의 위치 결과는 Fig. 3.8과 같다. 선체의 초기 위치는 $[0 \ 0 \ 0]$ m이며, 총 두 번의 회전을 수행하고 최종 위치는 $[0 \ 90 \ 0]$ m이다. Trajectory는 앞서 생성된 선체의 이동 경로를 나타내며 위치 참조값이다. GPS는 위치 참조값을 입력으로 GPS 모델에서 생성된 위치 측정치이다. Estimate는 항법 연산의 위치 결과를 나타낸다. 항법 연산은 가속도계와 각속도계, 자력계를 이용하여 100 Hz로 위치, 속도, 방향각을 산출하며, 확장 칼만 필터를 통해 GPS와 엔코더로 위치, 속도 정보를 1 Hz로 보정한다. 그렇기 때문에 Fig. 3.8과 같이 GPS의 위치 정보가 다소 큰 에러를 가지더라도 항법 연산의 결과는 큰 에러를 갖지 않는다. Fig. 3.9는 항법 연산의 방향각 결과이며, 실제 방향각과 유사하다.

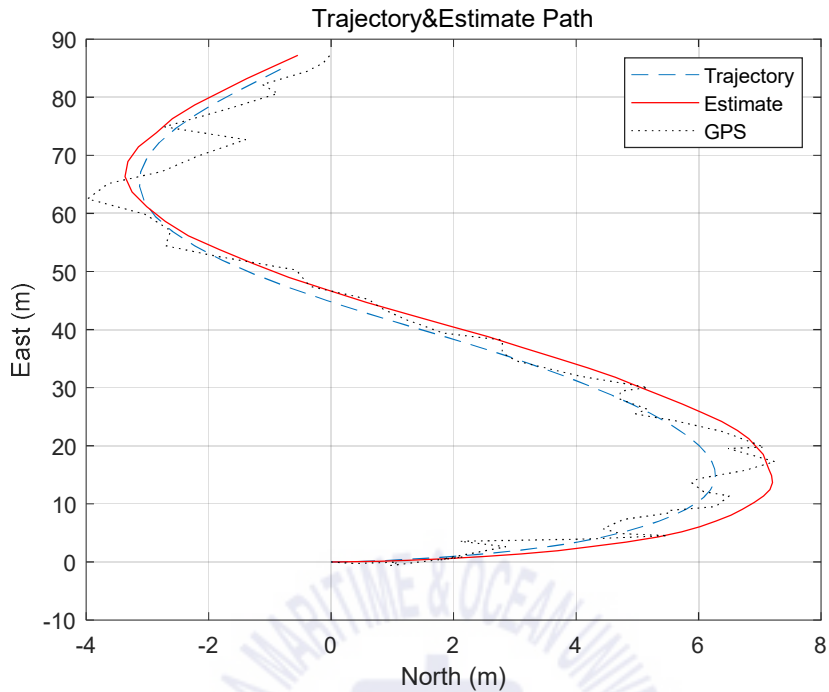


Fig. 3.8 Position result of navigation (Case 1)

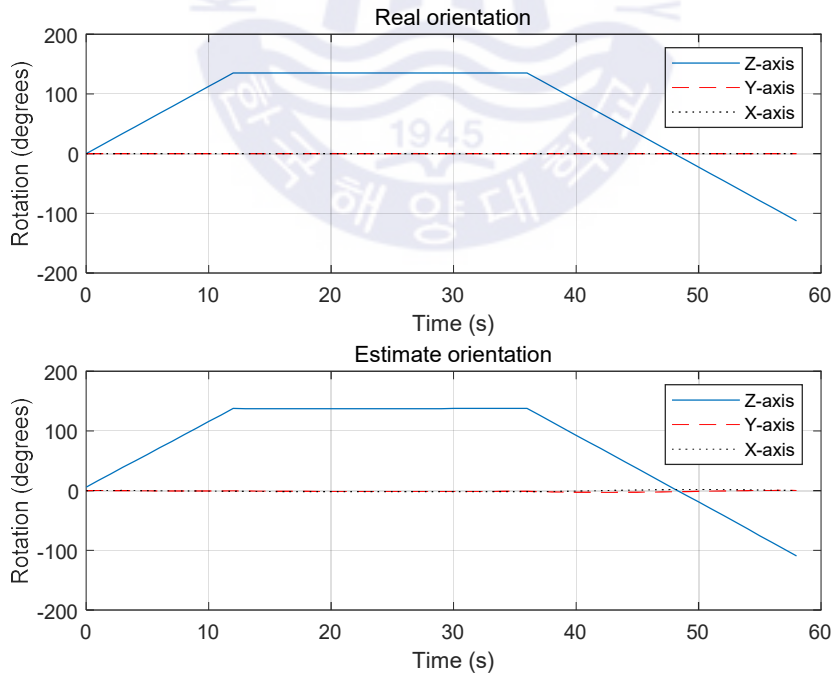


Fig. 3.9 Result orientation of navigation vs. real orientation (Case 1)

항법 연산의 속도 결과는 Fig. 3.10과 같다. Real은 앞서 생성된 가속도계를 통해 산출된 실제 선체의 이동 속도이며, Estimate는 항법 연산의 속도 결과이다. 가속도계를 통한 속도는 한번 적분하여 속도로 표현되었으며, 속도 정보는 100 Hz로 산출되었다. 항법 연산의 속도 결과는 엔코더 정보를 통해 1 Hz로 보정하여 산출되기 때문에 그래프에서 보이듯이 갱신 주기에 따라 속도가 변화되는 것을 볼 수 있다. Fig. 3.10을 통해 항법 연산의 속도 결과가 잘 추정되는 것을 확인할 수 있다.

Fig. 3.11은 항법 연산의 위치 결과와 실제 위치를 비교한 것이다. 각 축별로 Trajectory는 실제 위치, Estimate는 항법 연산의 위치 정보를 나타낸다. Fig. 3.12는 항법 연산의 위치 결과와 실제 위치의 차이를 각 축별로 나타낸 것이다. 각 축별로 가장 큰 에러 정도는 X축이 약 -1.4 m, Y축이 약 -2 m, Z축이 약 1.2 m로 확인되며, 위치 실효치(Root mean square, RMS) 오차는 [0.7 1.46 0.71] m로 산출된다.

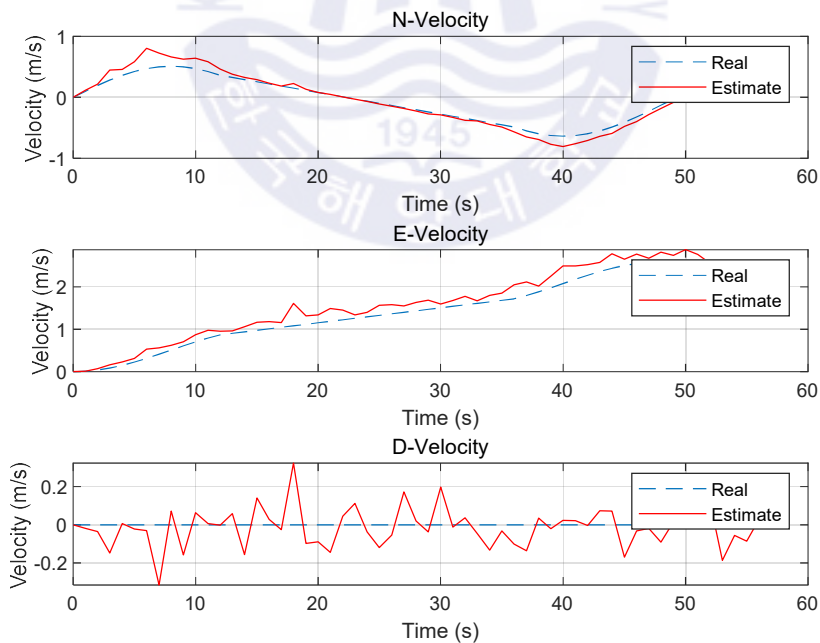


Fig. 3.10 Each velocity results of navigation (Case 1)

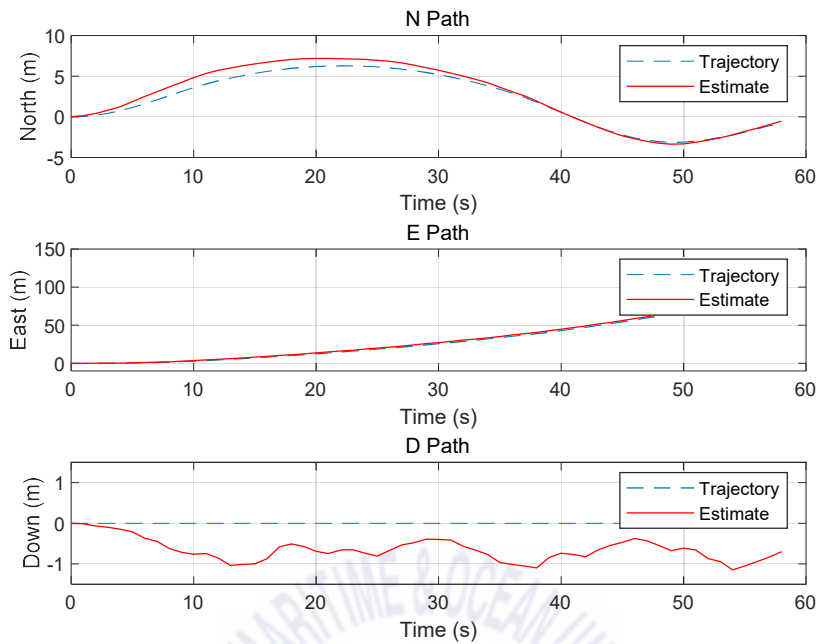


Fig. 3.11 Position result of navigation vs. real position (Case 1)

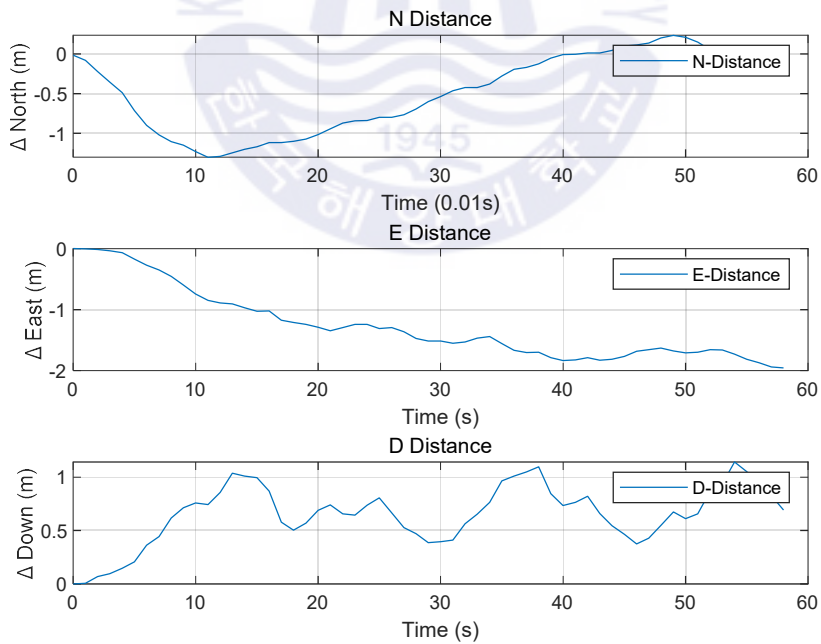


Fig. 3.12 Difference between position result of navigation and real position (Case 1)

3.5.2 사각형 궤적 항법 시뮬레이션 (Case 2)

본 시뮬레이션을 수행하는 이유는 임의의 이동 궤적에 따라 항법 결과가 어느 정도의 오차를 갖는지 확인하기 위해 수행하였으며, Case 1과 다른 사각형 궤적이다. Fig. 3.13은 IMU 모델의 가속도, 각속도 입력으로 사용될 선체의 각속도, 가속도 정보이다. 각속도는 초기에 $[0 \ 0 \ 0] \text{ rad/s}$ 이며, 28초와 36초, 63초에 순간적으로 값이 증가하여 $[0 \ 0 \ 2] \text{ rad/s}$ 으로 변경된 후 다시 $[0 \ 0 \ 0] \text{ rad/s}$ 로 변경되며, 최종 각속도도 이전과 같다. 가속도는 초기에 $[0 \ 0 \ 0] \text{ m/s}^2$ 이며, 각속도와 같은 경우로 28초와 36초, 63초에 순간적으로 값이 증감하여 $[6 \ 4 \ 0] \text{ m/s}^2$ 와 $[-6 \ -4 \ 0] \text{ m/s}^2$ 으로 진동하듯 변경되며, 최종 가속도는 $[0 \ 0 \ 0] \text{ m/s}^2$ 이 된다. 이에 따라 선체의 방향각은 Fig. 3.14와 같이 산출되며, 이는 항법 연산의 방향각 결과와 비교될 것이다. 방향각이 초기 $[0 \ 0 \ 0]^\circ$ 에서 28초까지 변함이 없다가 Z축이 증가하며 $[0 \ 0 \ 90]^\circ$ 로 변한 뒤, 36초에 Z축이 증가하여 $[0 \ 0 \ 180]^\circ$ 로 변한 뒤, 63초에 Z축이 증가하여 $[0 \ 0 \ -90]^\circ$ 로 변한 후 최종 방향각은 $[0 \ 0 \ 0]^\circ$ 가 된다. 앞서 설명한 데이터로 선체의 이동 경로를 산출하면 Fig. 3.15와 같으며, 사각형 형상으로 선체가 이동하였다.

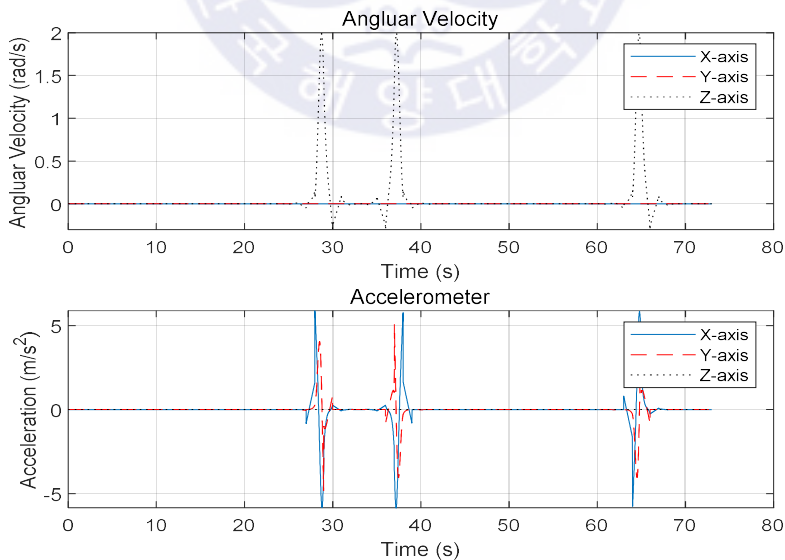


Fig. 3.13 Angular velocity/acceleration input of IMU model (Case 2)

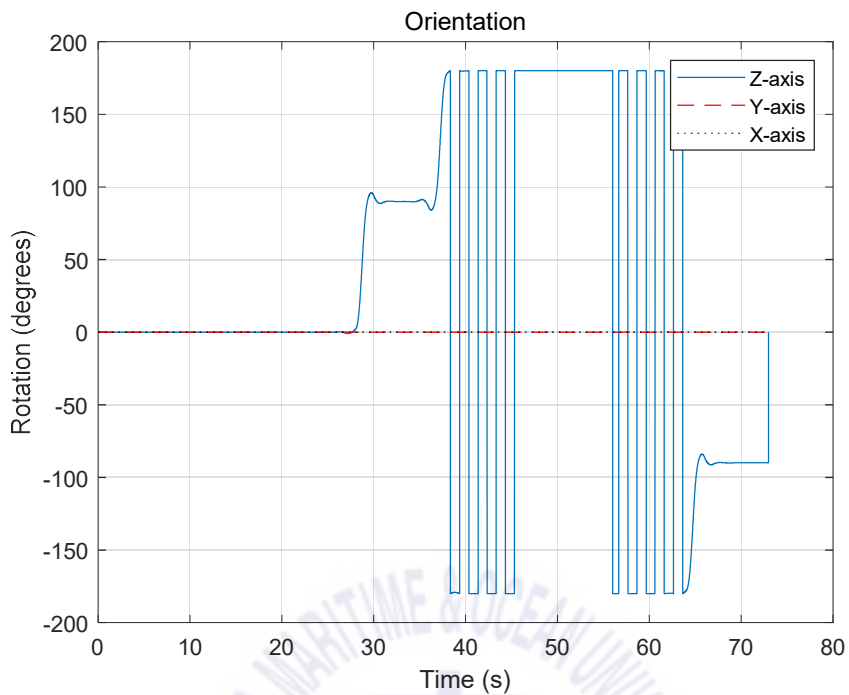


Fig. 3.14 Orientation input of IMU model (Case 2)

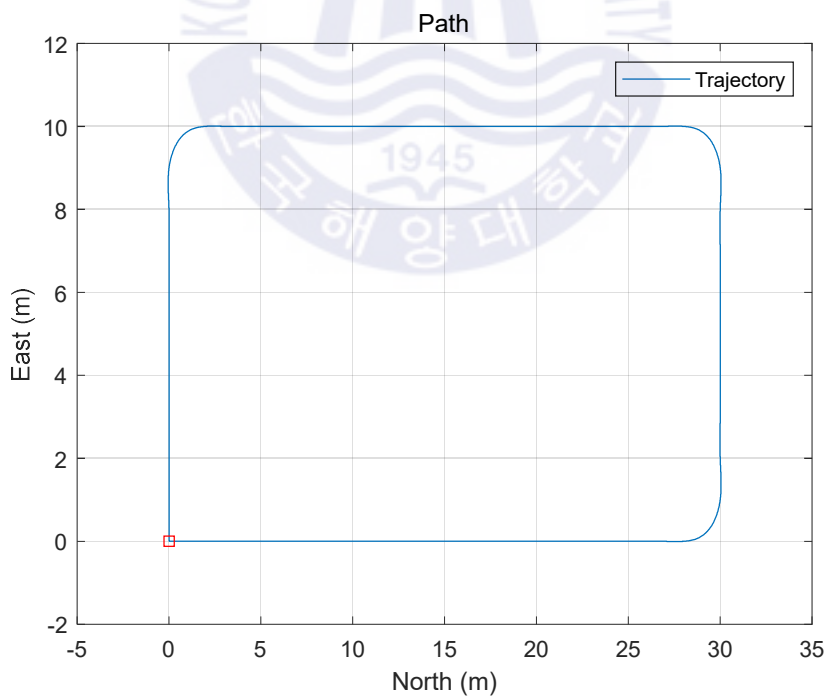


Fig. 3.15 Create a path (Case 2)

Fig. 3.16은 앞서 2장에서 설명한 IMU 센서 모델을 이용하여 생성된 각속도와 가속도, 방향각을 입력으로 출력된 센서 측정치 결과이다. 각 센서들의 파라미터에 따라 센서에서 계측되는 측정치를 산출하며, 가속도에는 Z축으로 중력 성분이 추가되며, 센서들에 센서 노이즈가 추가된 것이 확인된다.

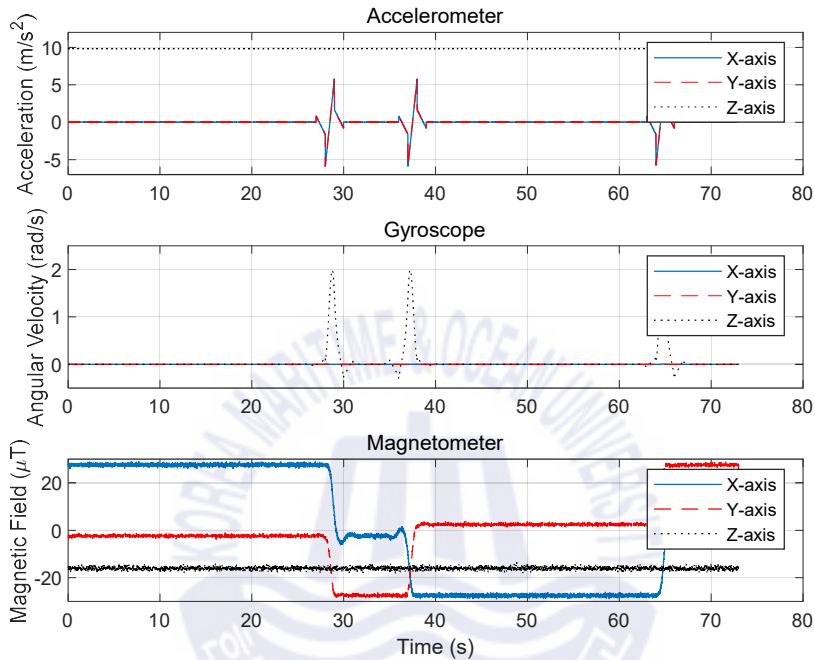


Fig. 3.16 Navigation sensor measurements derived from the IMU model (Case 2)

항법 연산의 위치 결과는 Fig. 3.17과 같다. 선체의 초기 위치는 $[0 \ 0 \ 0]$ m이며, 총 세 번의 회전을 수행하고 최종 위치는 $[-1 \ 1.3 \ 0]$ m이다. Trajectory는 앞서 생성된 선체의 이동 경로를 나타내며 위치 참조값이다. GPS는 위치 참조값을 입력으로 GPS 모델에서 생성된 위치 측정치이다. Estimate는 항법 연산의 위치 결과를 나타낸다. 항법 연산은 가속도계와 각속도계, 자력계를 이용하여 100 Hz로 위치, 속도, 방향각을 산출하며, 확장 칼만 필터를 통해 GPS와 엔코더로 위치, 속도 정보를 1 Hz로 보정한다. 그렇기 때문에 Fig. 3.17과 같이 GPS의 위치 정보가 다소 큰 에러를 가지더라도 항법 연산의 결과는 큰 에러를 갖지 않는다. Fig. 3.18은 항법 연산의 방향각 결과이며, 실제 방향각과 유사하다.

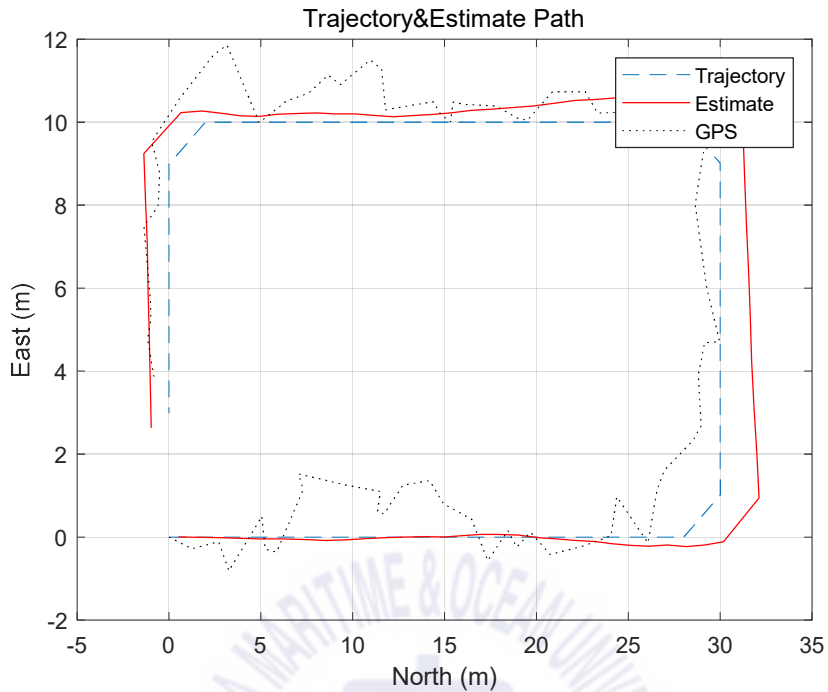


Fig. 3.17 Position result of navigation (Case 2)

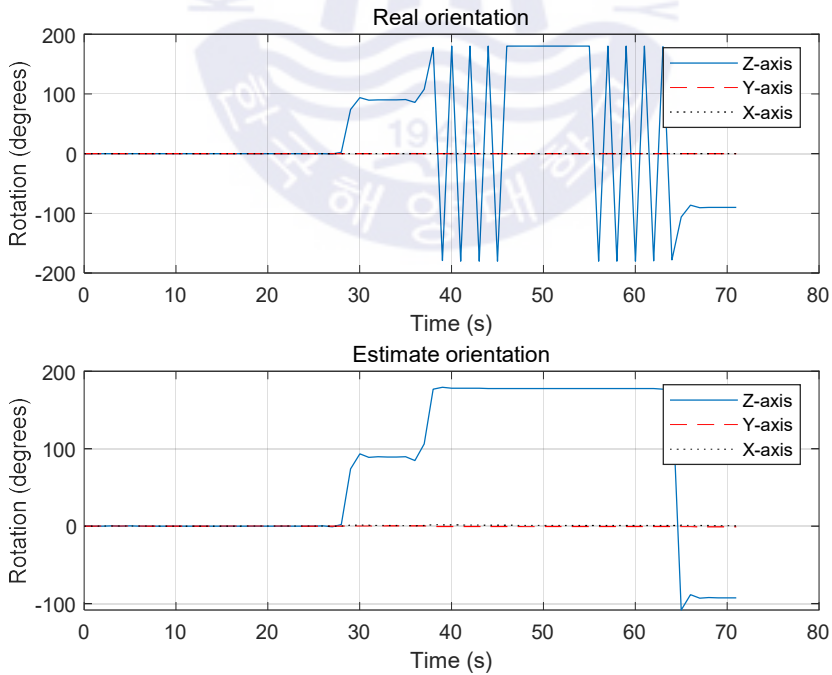


Fig. 3.18 Result orientation of navigation vs. real orientation (Case 2)

항법 연산의 속도 결과는 Fig. 3.19와 같다. Real은 앞서 생성된 가속도계를 통해 산출된 실제 선체의 이동 속도이며, Estimate는 항법 연산의 속도 결과이다. 가속도계를 통한 속도는 한번 적분하여 속도로 표현되었으며, 속도 정보는 100 Hz로 산출되었다. 항법 연산의 속도 결과는 엔코더 정보를 통해 1 Hz로 보정하여 산출되기 때문에 그래프에서 보이듯이 갱신 주기에 따라 속도가 변화되는 것을 볼 수 있다. Fig. 3.19를 통해 항법 연산의 속도 결과가 잘 추정되는 것을 확인할 수 있다.

Fig. 3.20은 항법 연산의 위치 결과와 실제 위치를 비교한 것이다. 각 축별로 Trajectory는 실제 위치, Estimate는 항법 연산의 위치 정보를 나타낸다. Fig. 3.21은 항법 연산의 위치 결과와 실제 위치의 차이를 각 축별로 나타낸 것이다. 각 축별로 가장 큰 에러 정도는 X축이 약 -2 m, Y축이 약 -0.8 m, Z축이 약 1.2 m로 확인되며, 위치 RMS 오차는 [1.46 0.31 0.67] m로 산출된다. Case 1의 위치 RMS 오차와 비교하면 X와 Y는 다소 변경되었으며, Z는 큰 차이가 없다. 하지만 두 경우의 항법 연산의 RMS 오차는 [2 2 1] m 이내로 확인된다.

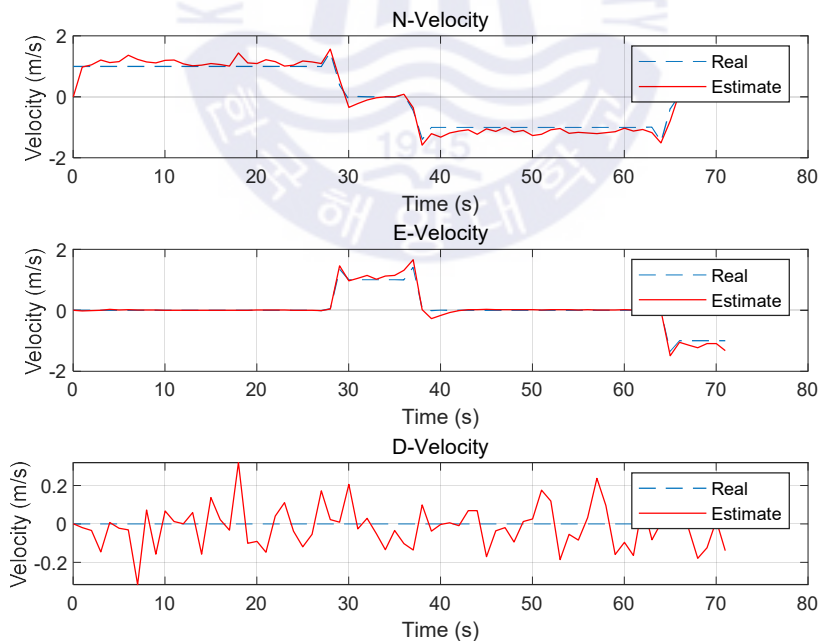


Fig. 3.19 Each velocity results of navigation (Case 2)

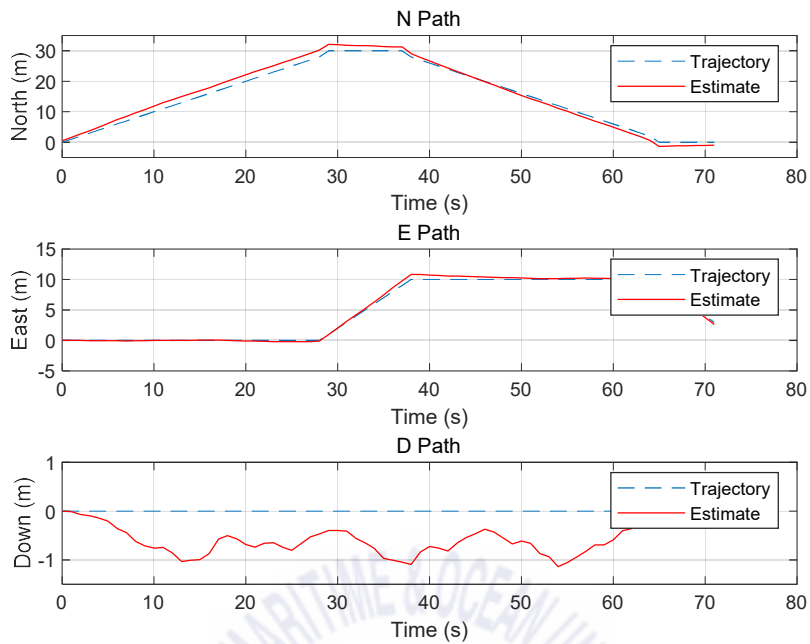


Fig. 3.20 Position result of navigation vs. real position (Case 2)

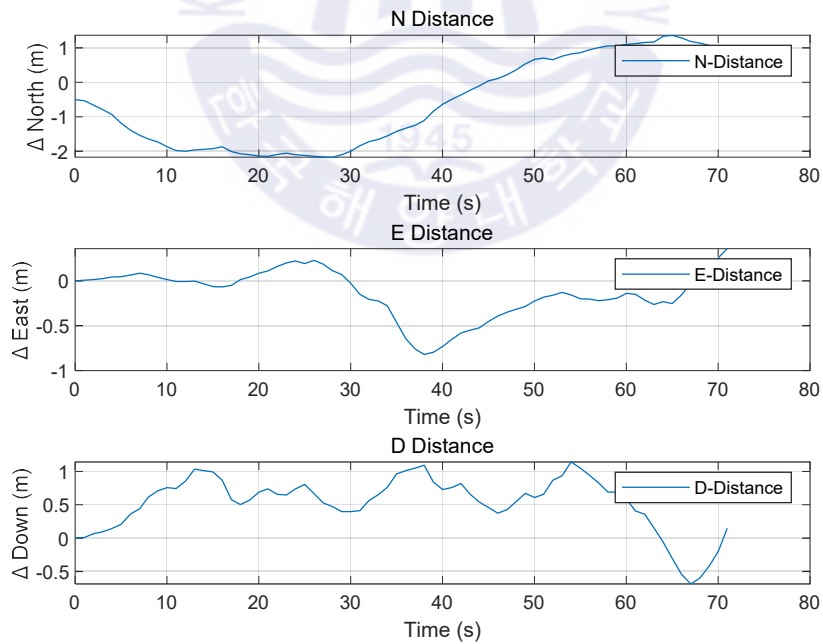


Fig. 3.21 Difference between position result of navigation and real position (Case 2)

3.5.3 모의 진회수 항법 시뮬레이션 (Case 3)

본 시뮬레이션을 수행하는 이유는 수중트랙로봇을 운용 시에 선박에서 진회수할 때 수직 방향으로 움직이게 되며, 엄빌리컬 케이블의 꼬임 정도를 확인할 수 있다. Fig. 3.22는 IMU 모델의 가속도, 각속도 입력으로 사용될 선체의 각속도, 가속도 정보이다. 각속도는 초기에 $[0 \ 0 \ 0.2] \text{ rad/s}$ 이며, 변동 없이 최종까지 그대로 유지된다. 가속도는 초기에 $[0 \ 2 \ 0] \text{ m/s}^2$ 이며, 변동 없이 최종까지 그대로 유지된다. 여기에 방향각의 초기를 $[0 \ 0 \ 90]^\circ$ 으로 설정하고 방향각을 산출하면 Fig. 3.23과 같이 산출되며, 이는 항법 연산의 방향각 결과와 비교될 것이다. 방향각이 초기 $[0 \ 0 \ 90]^\circ$ 에서 계속해서 Z축이 증가하게 되는데 8초에 $[0 \ 0 \ 180]^\circ$, 39초에 $[0 \ 0 \ 180]^\circ$, 71초에 $[0 \ 0 \ 180]^\circ$ 이며, 최종까지도 계속 방향각은 증가된다. 앞서 설명한 데이터로 선체의 이동 경로를 산출하면 Fig. 3.24와 같으며, 스파이럴 형상으로 선체가 원을 그리며 Z축으로 상승하며 이동하였다.

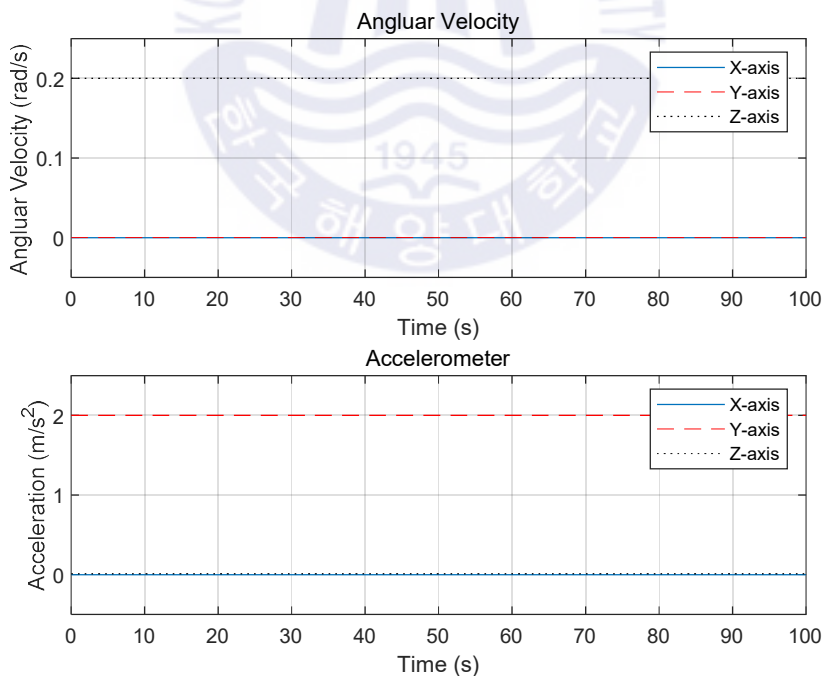


Fig. 3.22 Angular velocity/acceleration input of IMU model (Case 3)

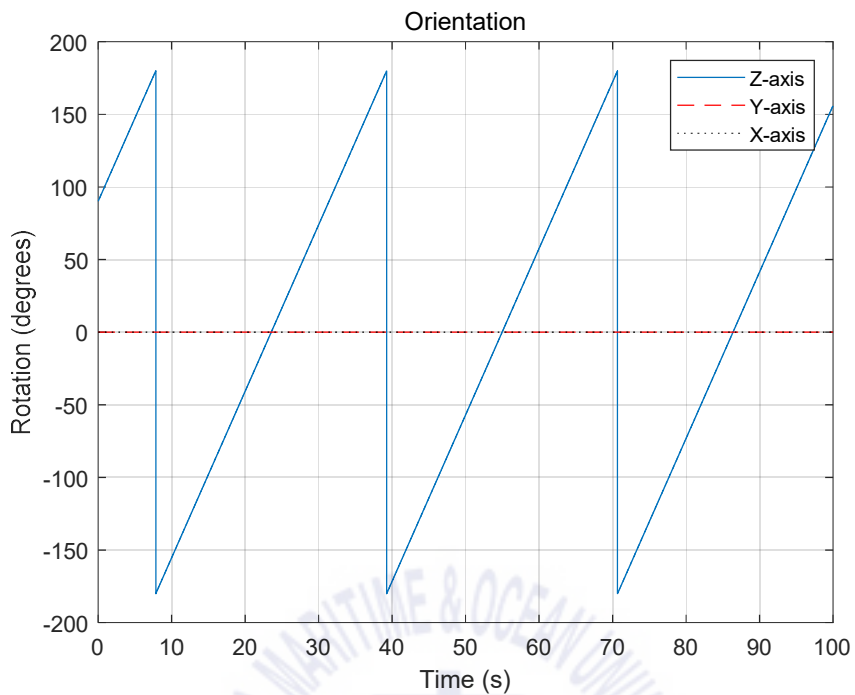


Fig. 3.23 Orientation input of IMU model (Case 3)

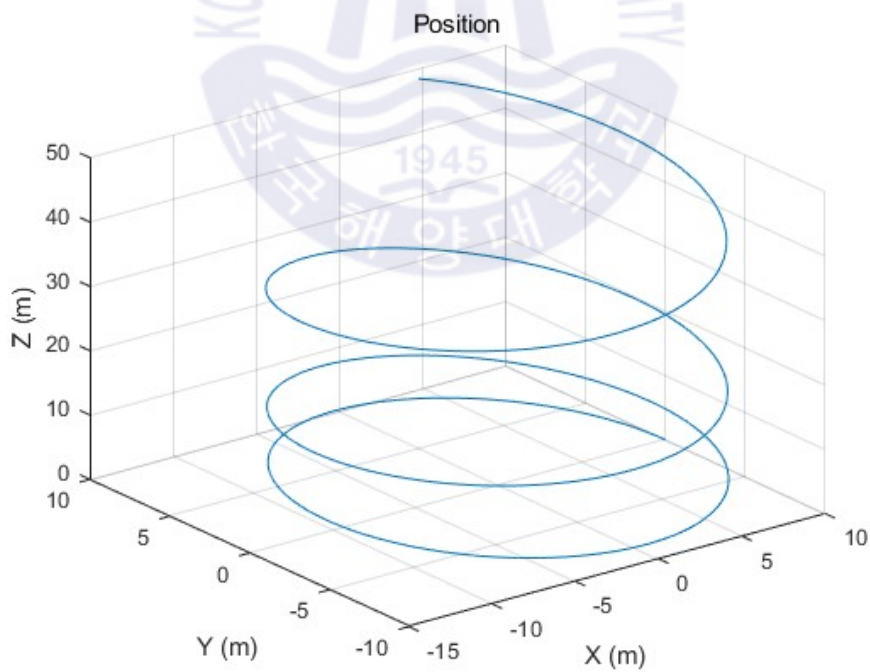


Fig. 3.24 Create a path (Case 3)

Fig. 3.25는 앞서 2장에서 설명한 IMU 센서 모델을 이용하여 생성된 각속도와 가속도, 방향각을 입력으로 출력된 센서 측정치 결과이다. 각 센서들의 파라미터에 따라 센서에서 계측되는 측정치를 산출하며, 가속도에는 Z축으로 중력 성분이 추가되며, 센서들에 센서 노이즈가 추가된 것이 확인된다.

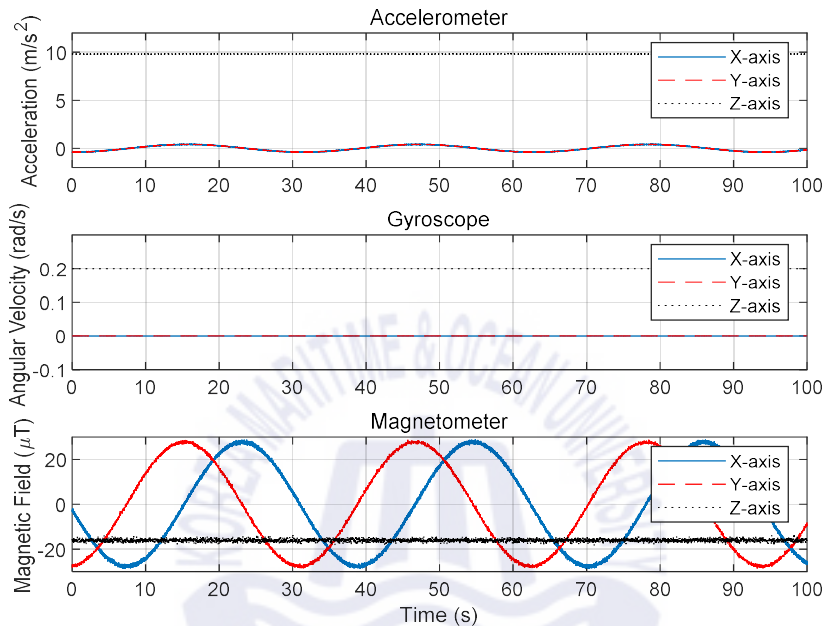


Fig. 3.25 Navigation sensor measurements derived from the IMU model (Case 3)

항법 연산의 위치 결과는 Fig. 3.26과 같다. 선체의 초기 위치는 $[50 \ 0 \ 0]$ m이며, 총 3.5 번의 회전을 수행하고 최종 위치는 $[19.5 \ 45.6 \ 50]$ m이다. Trajectory는 앞서 생성된 선체의 이동 경로를 나타내며 위치 참조값이다. GPS는 위치 참조값을 입력으로 GPS 모델에서 생성된 위치 측정치이다. Estimate는 항법 연산의 위치 결과를 나타낸다. 항법 연산은 가속도계와 각속도계, 자력계를 이용하여 100 Hz로 위치, 속도, 방향각을 산출하며, 확장 칼만 필터를 통해 GPS와 엔코더로 위치, 속도 정보를 1 Hz로 보정한다. 그렇기 때문에 Fig. 3.26과 같이 GPS의 위치 정보가 다소 큰 에러를 가지더라도 항법 연산의 결과는 큰 에러를 갖지 않는다. Fig. 3.27은 항법 연산의 North-East 위치 결과, Fig. 3.28은 항법 연산의 North-Down 위치 결과이다. Fig. 3.29는 항법 연산의 방향각 결과이며, 실제 방향각과 유사하다.

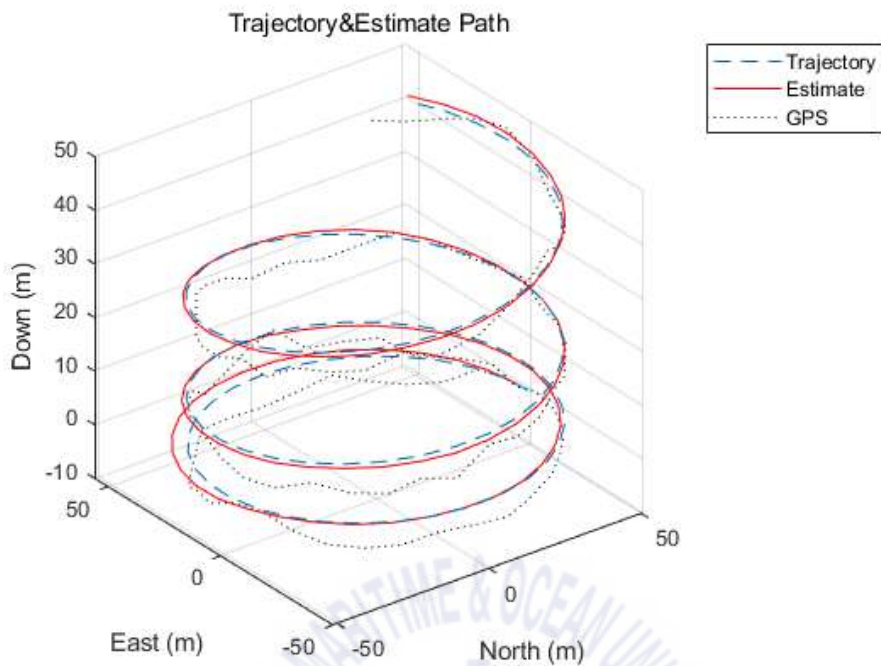


Fig. 3.26 Position result of navigation (Case 3)

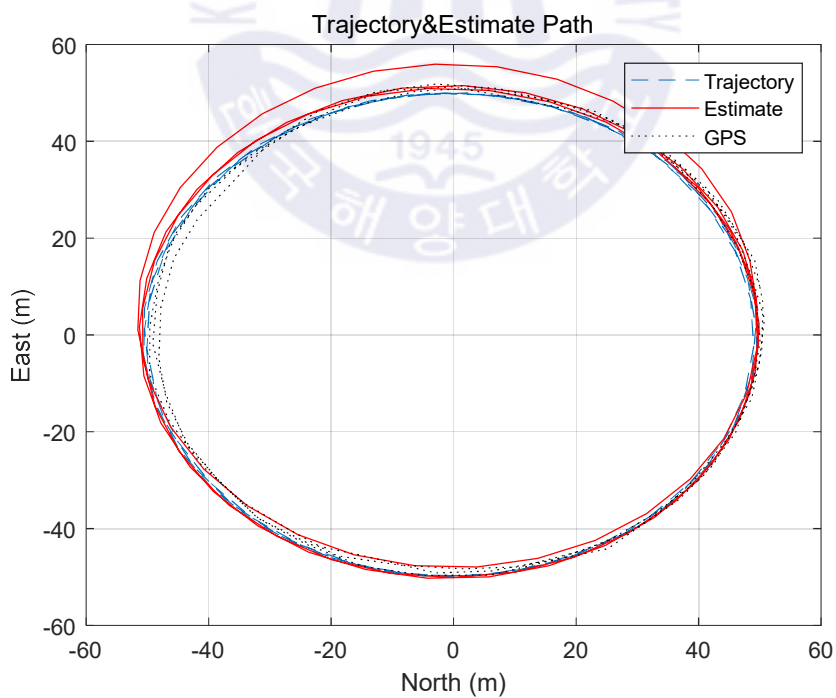


Fig. 3.27 Position result of navigation N-E (Case 3)

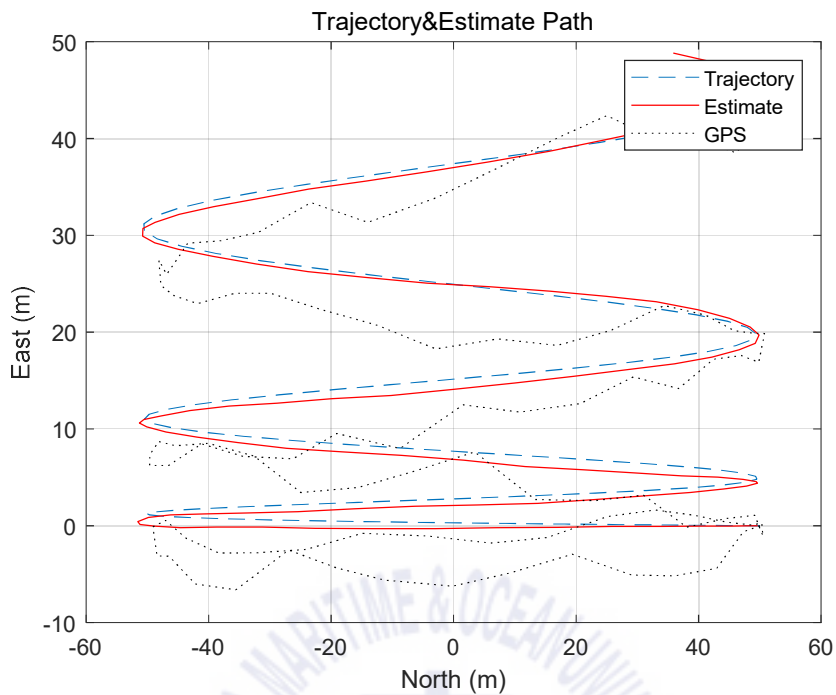


Fig. 3.28 Position result of navigation N-D (Case 3)

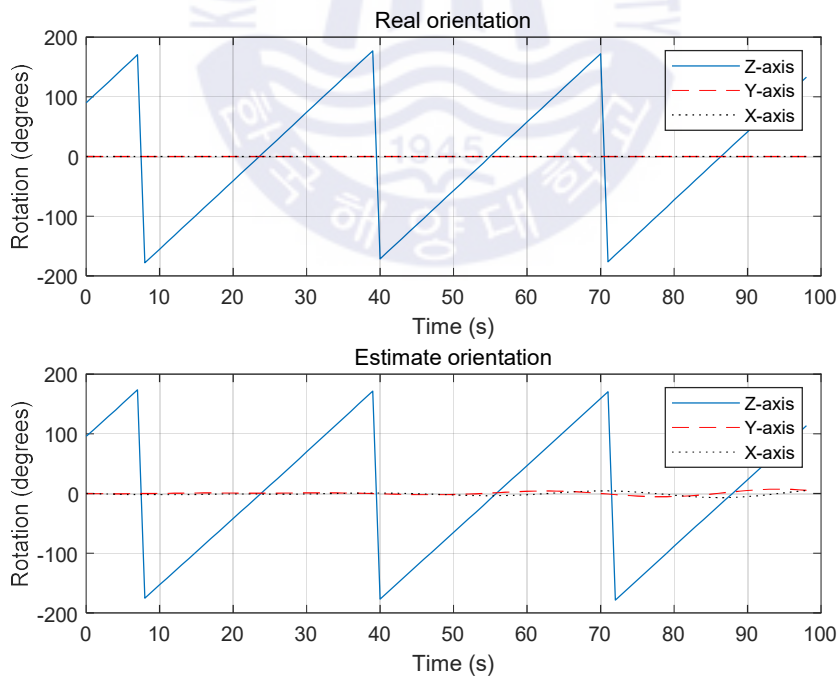


Fig. 3.29 Result orientation of navigation vs. real orientation (Case 3)

항법 연산의 속도 결과는 Fig. 3.30과 같다. Real은 앞서 생성된 가속도계를 통해 산출된 실제 선체의 이동 속도이며, Estimate는 항법 연산의 속도 결과이다. 가속도계를 통한 속도는 한번 적분하여 속도로 표현되었으며, 속도 정보는 100 Hz로 산출되었다. 항법 연산의 속도 결과는 엔코더 정보를 통해 1 Hz로 보정하여 산출되기 때문에 그래프에서 보이듯이 갱신 주기에 따라 속도가 변화되는 것을 볼 수 있다. Fig. 3.30을 통해 항법 연산의 속도 결과가 잘 추정되는 것을 확인할 수 있다.

Fig. 3.31은 항법 연산의 위치 결과와 실제 위치를 비교한 것이다. 각 축별로 Trajectory는 실제 위치, Estimate는 항법 연산의 위치 정보를 나타낸다. Fig. 3.32는 항법 연산의 위치 결과와 실제 위치의 차이를 각 축별로 나타낸 것이다. 각 축별로 가장 큰 에러 정도는 X축이 약 1.5 m, Y축이 약 -7 m, Z축이 약 1.2 m로 확인되며, 위치 RMS 오차는 [0.74 2.58 0.64] m로 산출된다. Case 1, 2의 위치 RMS 오차와 비교하면 Y가 다소 변경되었으며, X와 Z는 큰 차이가 없다. 3가지 경우를 바탕으로 항법 연산의 RMS 오차는 [2 3 1] m 이내로 확인된다.

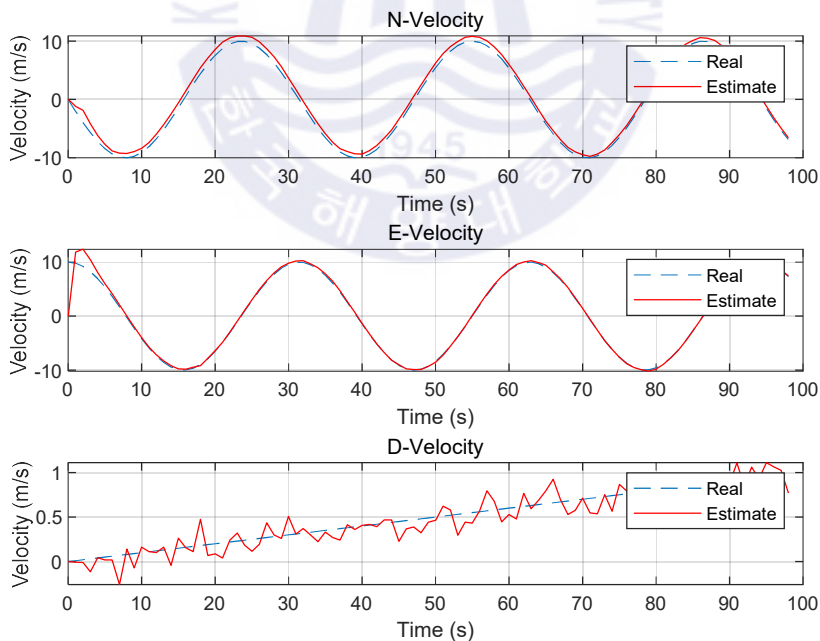


Fig. 3.30 Each velocity results of navigation (Case 3)

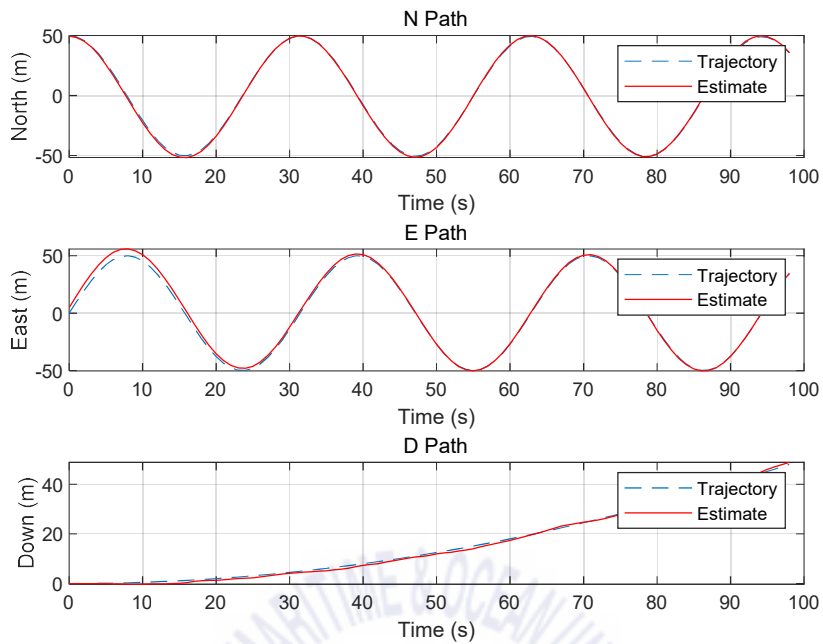


Fig. 3.31 Position result of navigation vs. real position (Case 3)

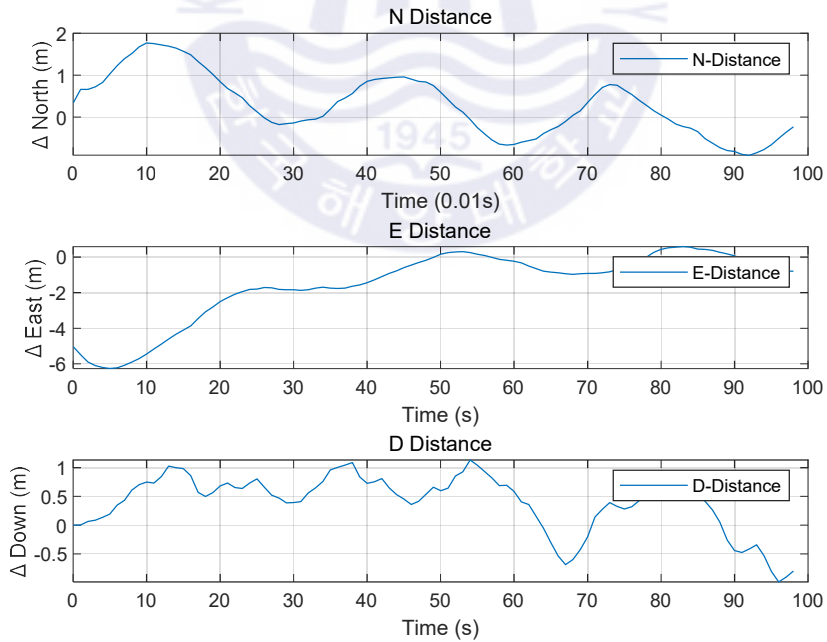


Fig. 3.32 Difference between position result of navigation and real position (Case 3)

제 4 장 수중항법 시스템 및 플랫폼 구성

4.1 수중항법 시스템

본 절에서는 3장에서 소개된 융합 항법 알고리즘의 성능 확인을 위하여 개발된 수중항법 시스템의 하드웨어 구성에 대하여 소개한다. 개발된 수중항법 시스템의 각 센서의 제원, 형상, 기타 탑재 장비, 제어 시스템, 통신 방법을 서술한다.

4.1.1 시스템 구성

본 연구에서 개발된 수중항법 시스템의 외관은 아래 Fig. 4.1과 같다. 그림에서 확인할 수 있듯이 외관은 방수가 가능한 박스 형태의 알루미늄 케이스를 이용하며, 상단에는 GPS 안테나를 측면에 측심기(Altimeter), 측면 상단에는 USBL responder를 탑재 가능하도록 설계되었다. 내부에는 항법 연산을 위한 센서들과 누수 측정을 위한 누수 센서, 통신을 위한 Ethernet Hub, 센서에 전원 공급 및 데이터 통신을 위한 인터페이스 보드가 탑재되었다. 아래 Table 4.1은 수중항법 시스템의 제원과 탑재 장비를 나타낸다.

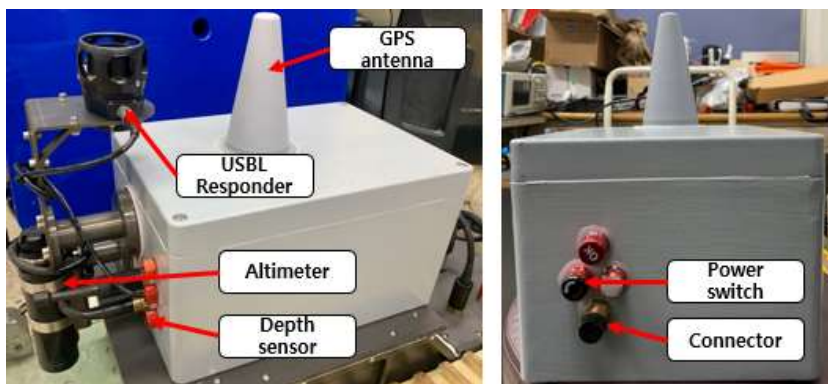


Fig. 4.1 Developed underwater navigation system

Table 4.1 Specifications of underwater navigation system

Index	Value	Units
Length	420	mm
Width	300	mm
Height	300	mm
Weight (dry)	8	kg
Weight (water)	2	kg
Standard components	RLG, Depth, USBL, GPS, TCM, Altimeter, Leak sensor, Ethernet Hub	

수중항법 시스템은 개별 또는 수중트랙로봇에 장착되어 실험이 가능하도록 설계하였다. Fig. 4.2는 수중항법 시스템의 개략도이다. 시스템은 3개의 부분으로 나뉘는데, 센서부, 전원부, 통신부로 구성된다. 센서부는 항법 연산과 시스템 운용을 위한 센서들이 장착되는데 총 8가지이다. GPS, 자력계, RLG, 수심 센서, 측심기, 누수 센서 마지막으로 육상 실험 시에 항법 연산 결과의 대조군이 될 INS가 있다. 전원부는 외부에서 공급된 전원을 각 센서에 맞는 전원으로 변환하여 원활하게 동작할 수 있도록 한다. 또한 통신부는 각 센서들로부터 받은 데이터를 한 노드로 모아서 외부로 전달하는 역할을 수행한다.

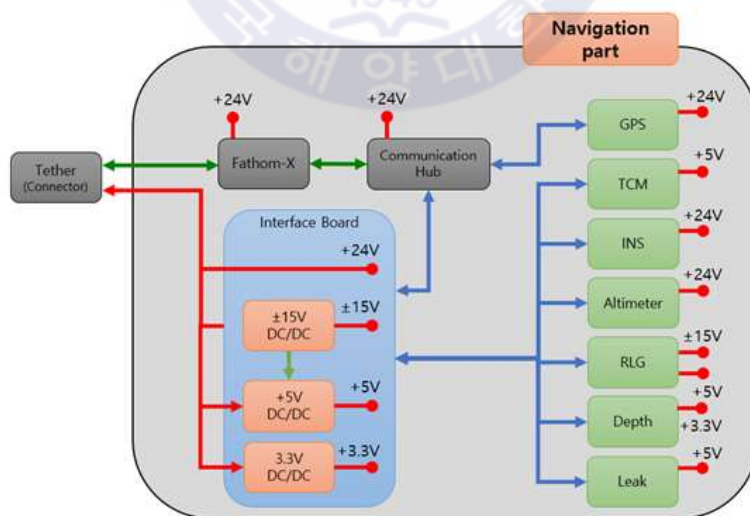


Fig. 4.2 Block diagram of underwater navigation system

이러한 전원부와 통신부를 관장하는 인터페이스 회로를 설계하였으며, 인터페이스 보드는 Fig. 4.3과 같이 제작되었다. 인터페이스 보드에는 본래의 기능에 전원 스위칭 회로를 추가하여, 외부에 장착된 스위치로 내부에 공급되는 전원을 제어(on/off)할 수 있도록 설계하였다. 또한 RLG를 제대로 동작시키기 위해서는 $\pm 15V$ 와 $+5V$ 의 전압 스위칭이 중요한데, 이 조건을 맞추기 위해 서로 다른 두 전압을 스위칭하기 위한 회로도를 추가로 설계하였다. 설계된 인터페이스 회로에 대한 자세한 설명은 부록에 추가로 작성하였다. Fig. 4.4는 앞에 설명한 구성품들이 모두 장착된 수중항법 시스템의 내부 모습이다.

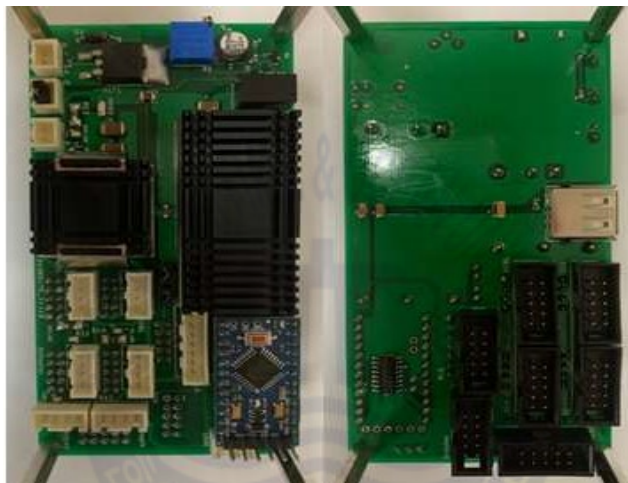


Fig. 4.3 Printed circuit board of the interface board

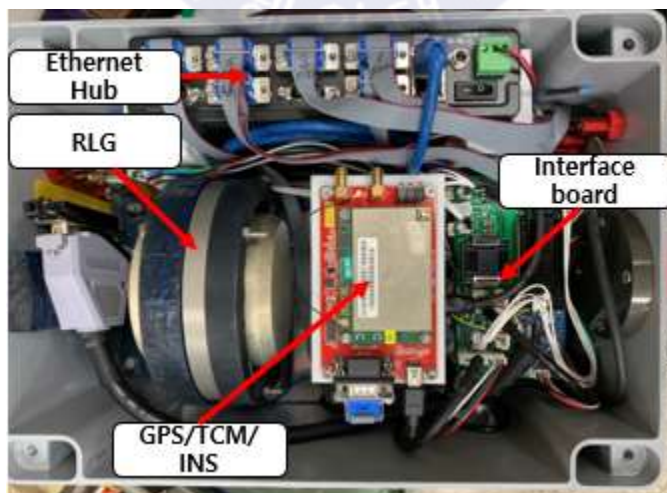


Fig. 4.4 Inside of underwater navigation system

수중항법 시스템에 사용된 항법 센서는 RLG, GPS, 자력계, USBL, 수심 센서, 측심기 총 6개가 사용되었으며, 개별 센서의 제원은 Table 4.2와 같다. INS의 연산 성능을 높이기 위해 MEMS타입의 IMU 대신에 RLG를 선정하였다. USBL은 소형 플랫폼에서 사용할 수 있도록 작은 크기를 갖는 센서를 선정하였다.

Table 4.2 Specifications of sensors included in navigation system

Index		Value		Units
RLG	Operating Range	Gyro	Accelerometer	$^{\circ}/\text{sec}, g$
		± 1074	70	
	Scale Factor Linearity	150	300	PPM(1σ)
	Bias Repeatability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
	Bias Stability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
	Operating Voltage	5, ± 15		V
GPS	Position Accuracy	1.3(H), 1.9(V)		m
	Velocity Accuracy	0.8(H), 1.3(V)		cm/s
	Update Rate	25		Hz
	Operating Voltage	4.5~24		V
Magnetic compass	Measurement Range	± 80		μT
	Resolution	± 0.05		μT
	Repeatability	± 0.1		μT
	Operating Voltage	4.5~10		V
USBL	Measurement Range	1(H), 1(V)		km
	Resolution	± 50		mm
	Operating Voltage	9~28		V
Depth	Measurement Range	300		m
	Relative Accuracy	± 200		mbar
	Operating Voltage	2.5~5.5		V
Altimeter	Measurement Range	0.7 to 100		m
	Operating Frequency	200		kHz
	Operating Voltage	2.5~5.5		V

4.1.2 운용 시스템

수중항법 시스템의 운용을 위하여 육상 운용콘솔을 설계 및 제작하였으며, 운용 소프트웨어는 C#을 이용하여 설계되었다. 제작된 운용콘솔과 운용 소프트웨어는 Fig. 4.5와 Fig. 4.6과 같다. 다중 센서 데이터를 처리하기 위해 별도의 소프트웨어가 필요하다. 본 소프트웨어는 각각의 센서 데이터를 취득하고 항법 알고리즘에 적용하기 위해 데이터를 가공하며 최종적으로 수중트랙로봇의 위치 및 자세를 연산한다. 또한 실험 동안 취득된 센서 데이터와 연산된 항법 데이터를 지속적으로 저장한다. 실험에 사용될 소형 무인 선박의 제어 박스에 설치된 소형 PC에 제작된 소프트웨어를 설치할 수 있는 환경을 구축하였다. 운용콘솔을 RF 통신을 통해서 소형 무인 선박의 PC와 연결되어 운용 소프트웨어를 운용할 수 있도록 구성되었다.



Fig. 4.5 Operation console of total system



Fig. 4.6 Operation software of underwater navigation system

4.2 수중트랙로봇

본 절에서는 4.1절에서 개발된 수중항법 시스템을 장착하여 성능 실험을 하기 위한 소형 수중트랙로봇(Underwater Track Robot, UTR)의 하드웨어 구성에 대하여 소개한다. 개발된 수중트랙로봇은 두 개의 트랙과 한 개의 몸체로 구분되며, 각 구성품의 제원, 형상, 제어 시스템, 통신 방법을 서술한다.

4.2.1 플랫폼 구성

제작된 수중트랙로봇은 Fig. 4.7과 같다. 그림에서 확인할 수 있듯이 플랫폼은 육상 및 해저면에서 이동이 가능하도록 2개의 무한궤도가 장착된 구조이다. 상부에는 수중항법 시스템을 탑재할 수 있도록 설계되었다. 또한 상판에는 전원 on/off를 위한 스위치, 전원 확인 LED, 방수 체크밸브, 수중항법 시스템과 연결을 위한 커넥터, 수상의 운용 컴퓨터와 연결하기 위한 커넥터, 배터리 충전 커넥터, 플랫폼 견인을 위한 결속부 등이 배치되어있다.

아래의 Table 4.3은 수중트랙로봇의 제원과 탑재 장비를 나타낸다. 플랫폼 외형은 길이 600 mm, 양측 무한궤도의 끝점 거리는 500 mm, 바닥부터 최상단 높이까지의 거리는 250 mm로 설계되었다. 총 2기의 BLDC 모터가 각각의 무한궤도를 독자로 제어하며, 제어 방식에 따라 전/후진 및 좌/우 선회가 가능하도록 설계되었다. 감속기와 추가 기어를 통해 전기 모터의 최종 기어비는 129(43×3) : 1 이다.

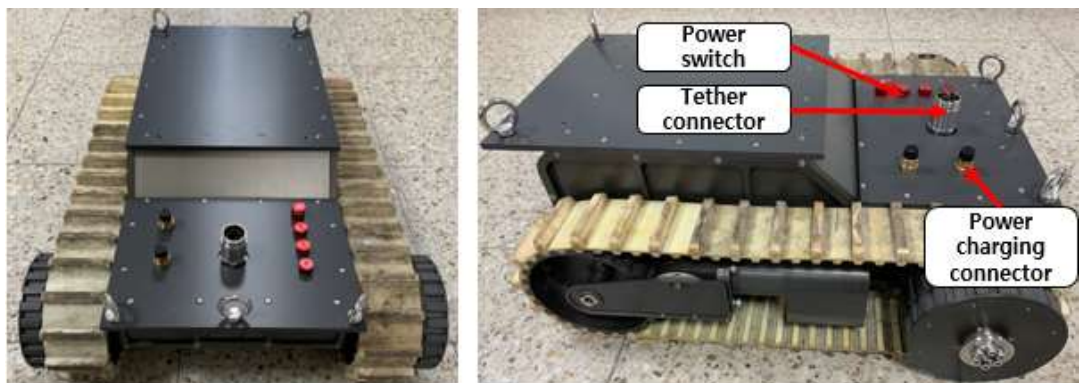


Fig. 4.7 Developed UTR

Table 4.3 Specifications of UTR

Index	Value	Units
Length	600	mm
Width	500	mm
Height	250	mm
Weight (dry)	35	kg
Weight (water)	10	kg
Standard components	Ethernet Hub, 200W BLDC motor, Motor drive, Battery	

수중트랙로봇은 상판에 수중항법 시스템이 장착 가능하도록 설계하였다. Fig. 4.8은 수중트랙로봇의 개략도이다. 시스템은 3개의 부분으로 나뉘는데, 구동부, 전원부, 통신부로 구성된다. 구동부는 BLDC 모터 2개와 모터드라이브 1개이다. 모터는 엔코더로 트랙의 속도를 측정할 수 있도록 엔코더가 결합된 모터를 사용하였다. 한 개의 모터드라이브가 2개의 BLDC 모터를 독립적으로 제어할 수 있는 모터드라이브를 선정하였다. 전원부는 내부에 탑재된 24V 30A LiPo 배터리로부터 각 부분과 수중항법 시스템에 전원을 공급한다. 또한 통신부는 모터드라이브에 명령 하달 및 데이터를 취득하는 역할을 수행한다. Fig. 4.9와 Fig. 4.10은 제작된 UTR의 제어 시스템이다.

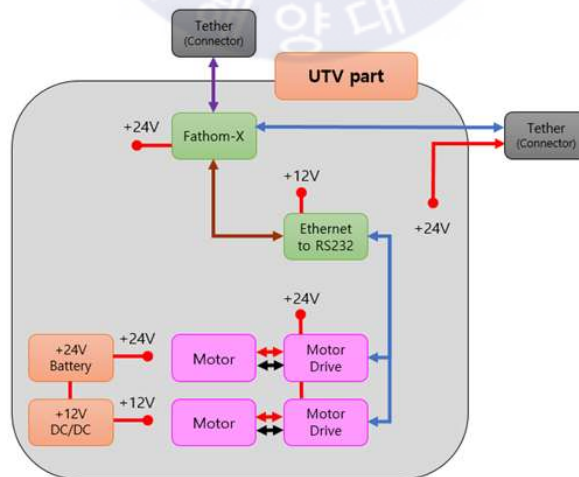


Fig. 4.8 Block diagram of UTR

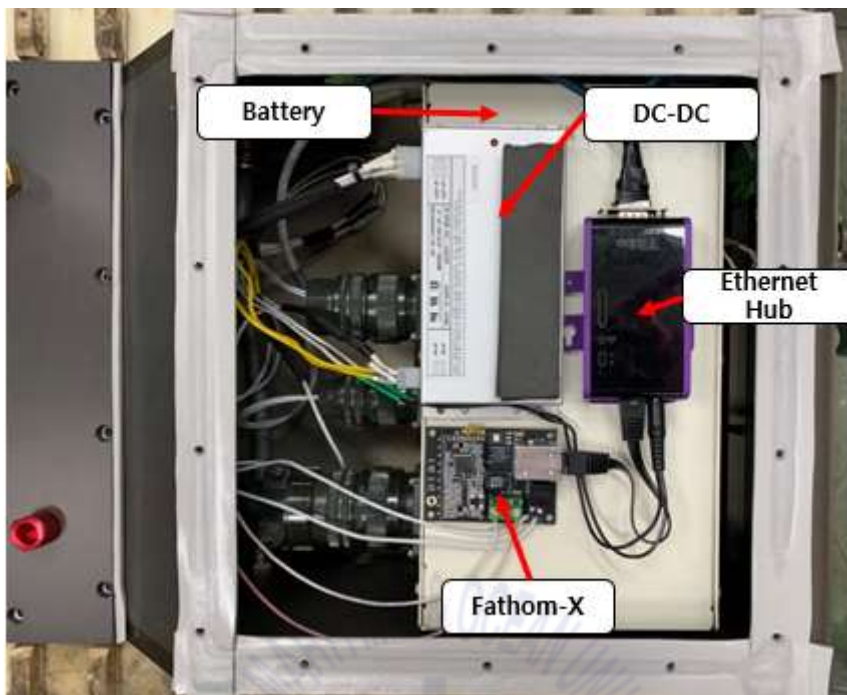


Fig. 4.9 Battery and communication parts of UTR

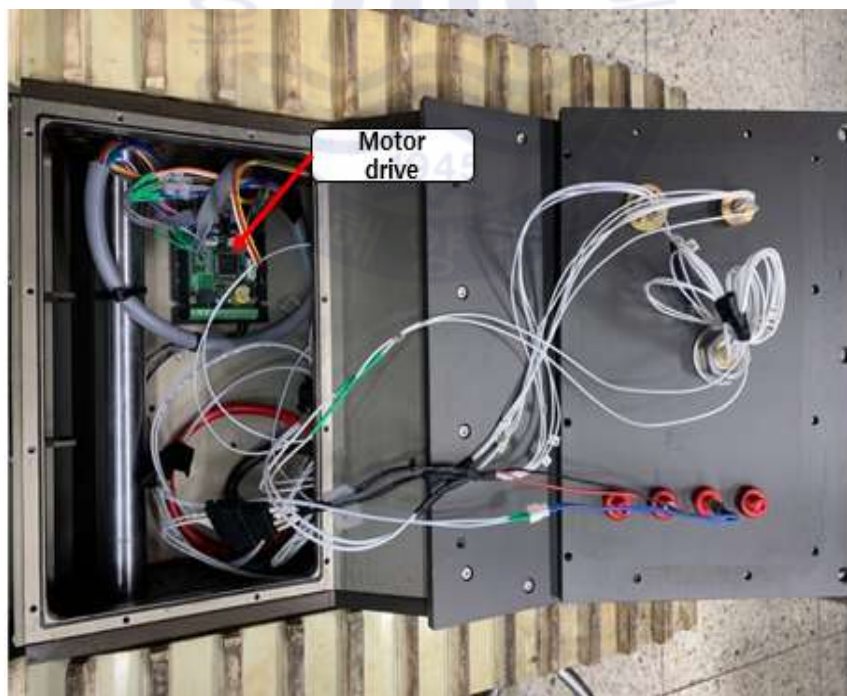


Fig. 4.10 Motor drive and connector parts of UTR

설계 단계에서 선정된 배터리 용량에 따라 수중항법 시스템과 수중트랙로봇의 운용 가능 시간을 산출하였다. Table 4.4는 수중항법 시스템의 각 구성품별 사용 개수와 전압, 전류, 소모 전력을 나타낸 것이다. 총 4종류의 전압이 사용되며, 가장 적은 전력을 소모하는 것은 수심 센서이며, 가장 많은 전력을 소모하는 것은 통신 장비인 MOXA NPort이다. Table 4.5는 DC-DC 효율(85%)에 따른 수중항법 시스템의 전체 소모 전력을 산출한 것이다. 수중과 육상에서 운용될 시에 사용되는 센서가 다르기 때문에 케이스 별로 산출하였다. 수중에서 운용 시에 최소 17 W에서 최대 32 W가 소모되는 것을 확인하였다.

Table 4.4 Power specifications for each component of underwater navigation system

Part Number	Q'ty (EA)	Voltage (V)	Current (mA)		Power (W)	
			Max.	Avg.	Max.	Avg.
Honeywell HG1700AG37	1	5	800	350	4	1.75
		15	700	250	10.5	3.75
		-15	90	70	1.35	1.05
Seatrak X150	1	24	250	25	6	0.6
PNI TCM3	1	5	20	20	0.1	0.1
Bar30 Depth Sensor	1	5	1.25	1.25	0.0063	0.0063
Tritech PA200	1	24	90	90	2.16	2.16
AsteRx-m	1	5	200	200	1	1
Mti-G-710	1	5	150	150	0.75	0.75
SOS Leak Sensor	1	5	20	15	0.48	0.36
Arduino pro mini	1	5	22	22	0.528	0.528
MOXA NPort5650-8-DT	1	24	300	300	7.2	7.2
Fathom-X Tether Interface Board	1	24	90	90	2.16	2.16

Table 4.5 Power specifications of underwater navigation system

Index	Power (W)	
	Max.	Avg.
Underwater	32.08425	17.26425
Ground	23.91	14.61

Table 4.6은 수중트랙로봇의 각 구성품별 사용 개수와 전압, 전류, 소모 전력을 나타낸 것이다. 총 2종류의 전압이 사용되며, 가장 적은 전력을 소모하는 것은 통신 장비인 Fathom-X이며, 가장 많은 전력을 소모하는 것은 BLDC 모터이다. Table 4.7은 DC-DC 효율(85%)에 따른 수중트랙로봇의 전체 소모 전력을 산출한 것이다. 운용 시에 최소 39.5 W에서 최대 372 W가 소모되는 것을 확인하였다.

Table 4.6 Power specifications for each component of UTR

Part Number	Q'ty (EA)	Voltage (V)	Current (mA)		Power (W)	
			Max.	Avg.	Max.	Avg.
MAXON EPOS4 Compat 50/15 CAN & 200W MAXON BLDC Motor	2	24	7580	728	181.92	17.472
Tibbo DS1102	1	12	500	200	6	2.4
Fathom-X Tether Interface Board	1	24	90	90	2.16	2.16

Table 4.7 Power specifications of UTR

Part	Power (W)	
	Max.	Avg.
Basic system	372	39.504

식(4.1)은 소모 전력에 대한 배터리 에너지량을 식으로 표현한 것이다. 여기서 Wh 는 배터리 에너지량(Wh), W_c 는 시스템의 에너지 사용량(W), L_f 는 평균 부하율(%), Op 는 가동률(%), H 는 운용 가능 시간(h), B_a 는 배터리 사용 가능 용량(%), S_f 는 사용 시간에 따른 배터리 성능 감쇄 및 안전 계수이다.

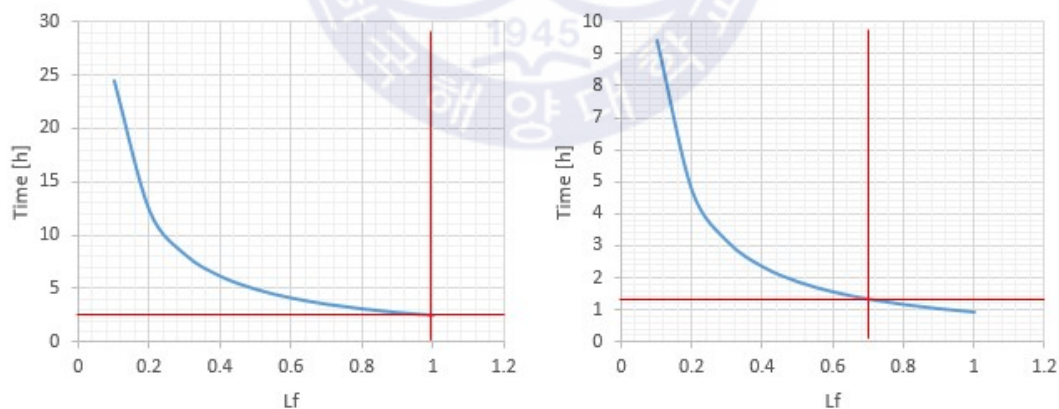
$$H = \frac{Wh}{(W_c \times L_f) \times Op / B_a \times S_f} \quad (4.1)$$

식(4.1)을 이용하여 수중항법 시스템과 수중트랙로봇의 운용 가능 시간을 산출하기 위해 Table 4.8과 같이 시스템별 사용 가능 배터리 용량을 나누고 연산에 사용될 파라미터를 선정하였다.

Table 4.8 Total power and parameters of the system

Wh (Total)	Wh (Navigation)	Lf	Op	Ba	Sf
480	80	1	0.9	0.85	1.2
	Wh (UTV)	Lf	Op	Ba	Sf
	400	0.7	0.8	0.85	1.2

전체 배터리 용량은 480 Wh이며, 이때 수중항법 시스템은 80 Wh, 수중트랙로봇은 400 Wh를 사용할 수 있다고 가정하였다. 또한 각 시스템별 연산에 사용될 파라미터를 위와 같이 산출하고 평균 부하율을 조절하며 운용 가능 시간을 산출하였다. Fig. 4.9의 (a)는 평균 부하율에 따른 수중항법 시스템의 운용 가능 시간을 나타낸 것이며, Fig. 4.9의 (b)는 평균 부하율에 따른 수중트랙로봇의 운용 가능 시간을 나타낸 것이다. 산출 결과 수중항법 시스템은 약 2.3 h, 수중트랙로봇은 약 1.3 h 운용 가능한 것으로 추정된다. 따라서 480 Wh의 배터리로 수중항법 시스템과 수중트랙로봇을 같이 운용하였을 때 적어도 1.3 h 운용 가능할 것으로 추정된다.



(a) Navigation system

(b) UTV

Fig. 4.11 Operating time of each system according to Lf

수중항법 시스템과 수중트랙로봇 간의 통신, 수중트랙로봇과 수상 운용 PC 간의 통신은 PoE(Power over Ethernet) 통신이 사용된다. 이와 같은 통신 방법을 선택한 이유는 2가닥의 선만을 사용하여 고속(80Mbps)의 통신이 가능하기 때문이다. 전체 시스템의 통신 구성도는 Fig. 4.10과 같다. 수중항법 시스템에서 7가지의 정보, 수중트랙로봇에서 1가지의 정보를 병렬로 PoE 통신을 이용하여 수상의 운용 PC에 전달하게 된다. 전달된 정보들은 4.1.2에서 설명한 운용콘솔과 운용 소프트웨어를 통해 항법 연산 및 수중트랙로봇을 구동하는 명령을 하달하게 된다.

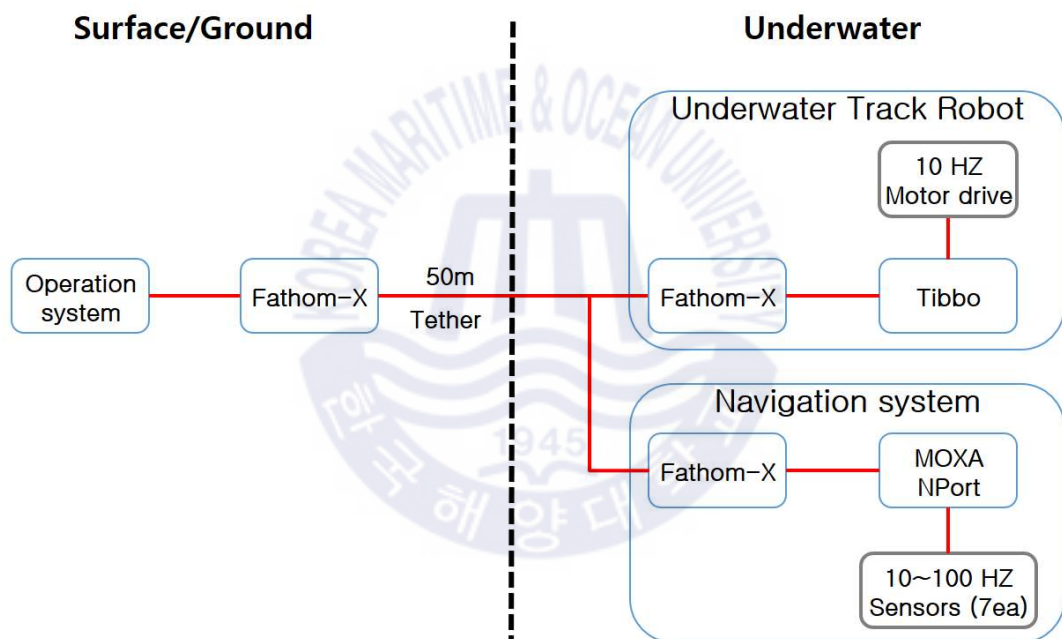


Fig. 4.12 Communication diagram of total system

4.2.2 운용 시스템

수중트랙로봇의 운용을 위한 육상 운용콘솔은 4.1.2에서 제작된 운용콘솔을 공통으로 사용된다. 마찬가지로 운용 소프트웨어는 C#을 이용하여 설계되었다. 제작된 운용 소프트웨어는 Fig. 4.11과 같다. 플랫폼에 장착된 두 개의 모터를 구동하고 엔코더 정보를 취득하기 위해서는 별도의 소프트웨어가 필요하다. 본 소프트웨어는 양측 모터에 구동 명령을 하달하며, 엔코더 정보를 취득하여 플랫폼의 운행 속도를 추정하기 위해 사용된다. 또한 실험 동안 취득된 모터 구동 상태와 추정된 플랫폼의 속도를 지속적으로 저장한다. 수중항법 시스템과 같이 소형 무인 선박의 제어 박스에 설치된 소형 PC에 제작된 소프트웨어를 설치할 수 있는 환경을 구축하였다. 수중항법 시스템과 수중트랙로봇의 운용 소프트웨어는 병렬로 실행하여 운용된다. 운용콘솔을 RF 통신을 통해서 소형 무인 선박의 PC와 연결되어 운용 소프트웨어를 운용할 수 있도록 구성되었다.

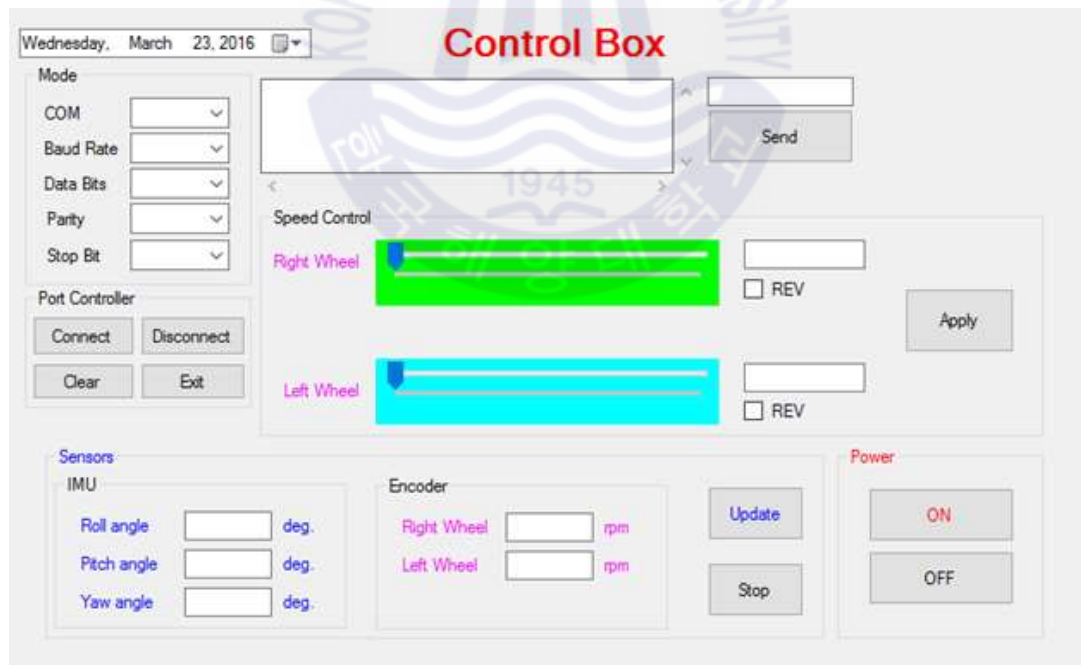


Fig. 4.13 Operation software of UTR

제 5 장 융합 항법 성능 실험

5.1 트랜칭 수중트랙로봇에 적용한 융합 항법 성능 실험

4장에서 제작된 수중트랙로봇에 수중항법 시스템을 적용하여 실험을 수행하기 이전에 상용 트랜칭 수중트랙로봇에 개발한 수중항법 시스템을 적용하여 성능 실험을 수행하였다(Ji et al., 2019). 이번 실험에 사용된 트랜칭 수중트랙로봇은 Fig. 5.1과 같다. 이 플랫폼은 해저 지반에서 해저케이블 매설 및 해저 수중 건설 작업을 수행하는 트랙 기반의 수중 작업 로봇이다. 주행을 위한 2개의 트랙 및 작업을 위한 트랜처가 결합된 형태이다. 플랫폼의 상세 제원은 Table 5.1과 같다. 외형은 길이 5800 mm, 양측 무한궤도의 끝점 거리는 3495 mm, 바닥부터 최상단 높이까지의 거리는 2595 mm로 설계되었다. 총 2기의 유압 모터가 각각의 무한궤도를 독자로 제어하며, 제어 방식에 따라 전/후진 및 좌/우 선회가 가능하다.



Fig. 5.1 Commercial trenching UTR

Table 5.1 Specification of commercial trenching UTR

Index	Value	Units
Length	5800	mm
Width	3495	mm
Height	2595	mm
Weight (dry)	35	ton
Contact length of track	400	mm
Track width	900	mm
Standard Components	IMU, Depth, USBL, GPS, TCM, Altimeter, Camera (analog /HD), Image sonar, Oil hydraulic motor, Trenching tool	

이 실험에서 사용된 플랫폼에 장착된 수중항법 시스템과 항법 알고리즘은 Fig. 5.2와 같다. 본 연구에서 개발한 수중항법 시스템의 간략화 모듈이다. 그 이유는 수중트랙로봇의 속도를 추정하기 위한 엔코더 정보를 받지 못하기 때문에 엔코더 정보를 통한 속도 보정을 제외한 항법 알고리즘을 사용하였다. 엔코더를 제외한 나머지 센서 구성은 기존의 시스템과 같다. RLG / 자력계는 동일 모델을 사용하였으며, USBL / 수심 센서 / 측심기는 기존의 시스템과 다른 모델이 사용되었다. 이 실험에 사용된 센서들의 상세 제원은 Table 5.2와 같다.

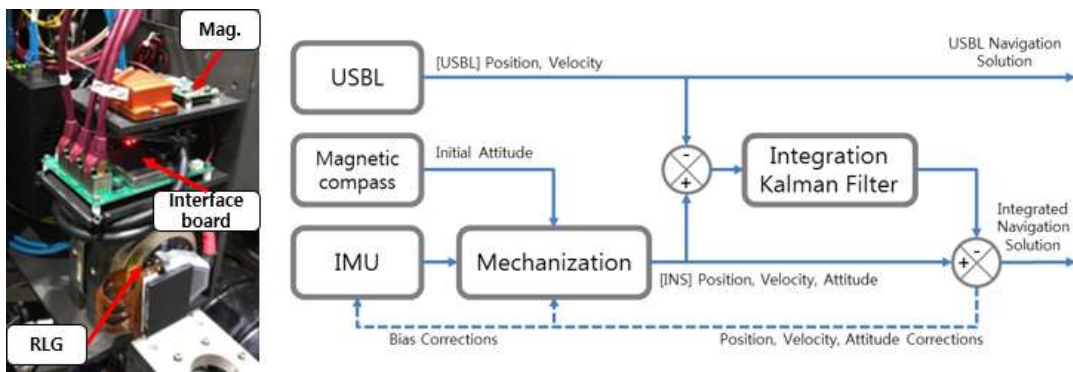


Fig. 5.2 Underwater navigation system without encoder

Table 5.2 Specifications of sensors included in trenching UTR

Index		Value		Units
RLG	Operating Range	Gyro	Accelerometer	$^{\circ}/\text{sec}, g$
		± 1074	70	
	Scale Factor Linearity	150	300	PPM(1σ)
	Bias Repeatability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
	Bias Stability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
Operating Voltage		5, ± 15		V
Magnetic compass	Measurement Range	± 80		μT
	Resolution	± 0.05		μT
	Repeatability	± 0.1		μT
	Operating Voltage	4.5~10		V
USBL	Measurement Range	7(H), 7(V)		km
	Resolution	± 0.1 Range		%
	Operating Voltage	24~48		V
Depth	Measurement Range	3000		m
	Relative Accuracy	± 0.01 Range		%
	Operating Voltage	9~30		V
Altimeter	Measurement Range	0.1 to 100		m
	Operating Frequency	500		kHz
	Operating Voltage	9~28		V

5.1.1 Heading 추정 성능 실험

수중항법 알고리즘의 진북 방향(Heading) 연산을 확인하기 위하여 플랫폼을 이용하여 육상에서 실험을 진행하였다. 플랫폼이 가로 / 세로가 30 m인 트랙을 이동했을 때 추정된 진북 방향 결과는 Fig. 5.3과 같다. 결과를 보면 총 3번의 회전을 하였으며, 초기 0° 에서 목표인 90° , 180° , 270° 를 정확히 추정하는 것을 확인할 수 있다. 목표 진북 방향으로 회전 후 에러 값이 큰 것은 플랫폼의 제어 성능의 영향이며, 목표 값으로 정확히 수렴하지 않는 것은 트랙과 바닥 면이 맞닿는 면이 균등하지 않아 완벽히 수렴하지 못한 것으로 판단된다.

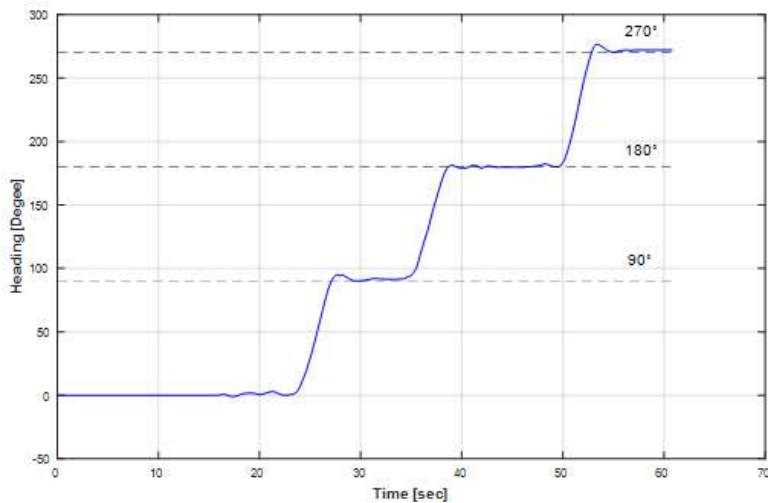


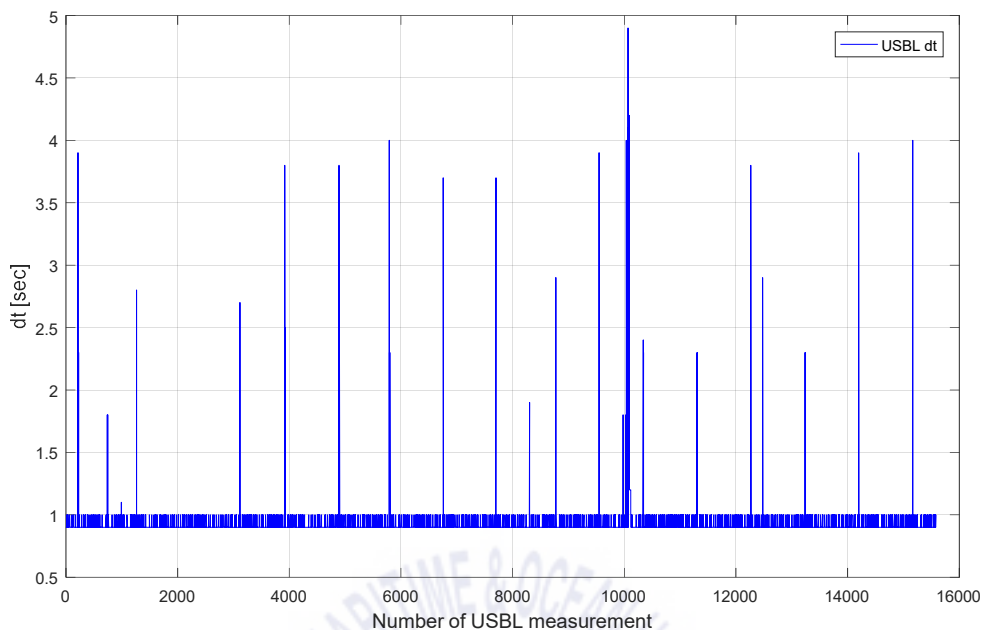
Fig. 5.3 Result of rectangle trajectory experiment

5.1.2 실험역 융합 항법 성능 실험

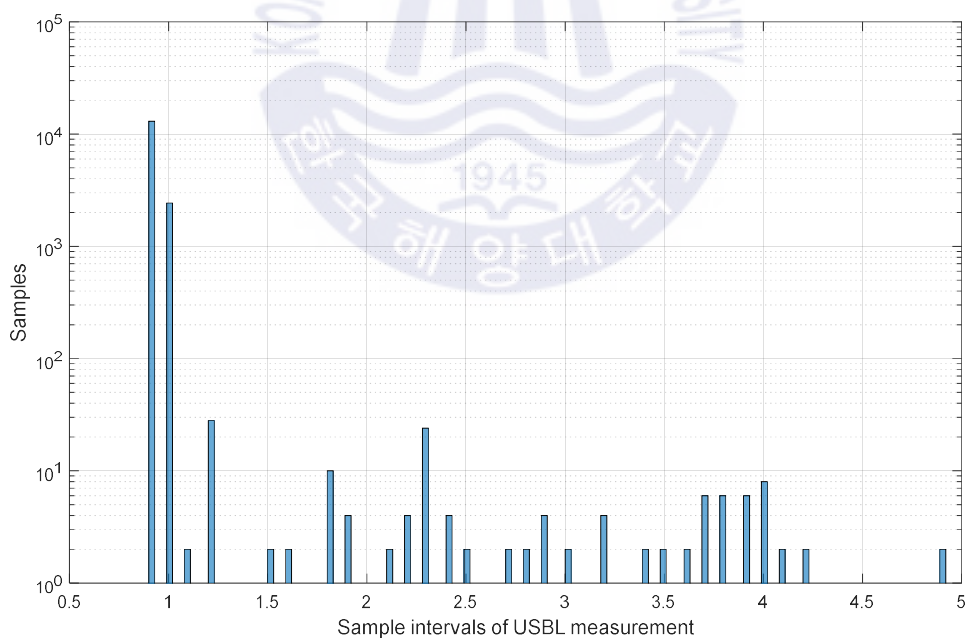
개발된 수중항법 시스템의 성능을 검증하기 위하여 동해에서 상용 트랜칭 수중트랙로봇의 트랜칭 작업 및 항법 실험을 수행하였다. 실험 지역의 위경도 좌표는 $36^{\circ} 05' 53.8''$ N, $129^{\circ} 26' 59.8''$ E이고, 실험 수심은 23m였다. Fig. 5.4는 실험을 위해 플랫폼을 실험역에 런칭하고 있는 모습이다.



Fig. 5.4 Launching underwater track robot



(a) Sample intervals of USBL measurements



(b) Histogram of sample intervals of USBL measurements

Fig. 5.5 Intervals of USBL measurements of navigation experiment

먼저 Fig. 5.5 (a)는 실험이 진행되는 동안 USBL의 신호가 갱신될 때마다 횡축은 USBL의 취득 신호 순번을, 종축은 취득 신호의 시간 간격을 나타낸 것이다. Fig. 5.5 (b)는 샘플시간 간격의 빈도수를 종축을 log scale로 나타낸 히스토그램이다. 실험 전 USBL의 데이터 갱신 주기를 1 초로 설정하였는데, 실험 결과를 보면 절반 이상의 갱신 주기는 1 초 정도로 보인다. 데이터가 갱신되는 시간이 길게는 5 초 정도까지 늘어났지만, 유효하지 않은 신호를 측정할 경우는 없었다. 데이터 갱신 시간이 1 초를 초과하는 경우는 플랫폼이 트랜칭 작업을 진행하거나 또는 지면이 고르지 않은 곳을 통과하면서 생긴 진동에 의해 USBL의 트랜스폰더가 영향을 받는 경우로 판단된다. 실험 수심이 23 m로 깊지 않았기 때문에 이와 같이 데이터 갱신이 잘 이루어진 것으로 판단된다.

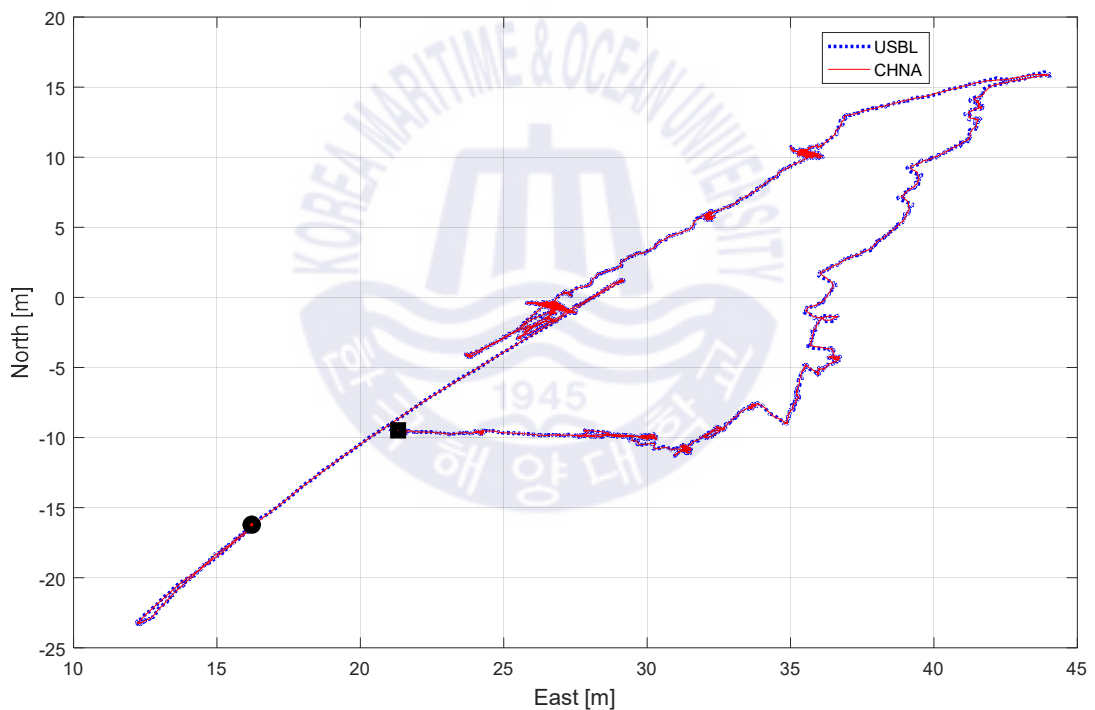


Fig. 5.6 Position result of navigation #1

다음은 3장에서 제안된 융합 항법 알고리즘을 이용하여 Fig. 5.6에 결과를 나타내었다. Fig. 5.6은 상용 트랜칭 수중트랙로봇이 이동한 수평면 궤적을 나타내며, USBL 측정 위치(점선), 항법 추정 위치(실선)로 표시되었다. 수중항법 알

고리즘은 플랫폼이 진수됨과 동시에 수행하여 회수될 때까지 약 4시간 동안 진행되었다. 플랫폼은 동그라미 모양의 위치에서 진수되었으며, 시계 방향으로 이동하며 실험을 진행한 뒤에 네모 모양의 위치에서 회수되었다. 실험 결과 중에 데이터가 다소 밀집되어 있는 부분들이 6지점이 있는데, 이는 트랜칭 작업 시 플랫폼에 진동이 크게 발생되어 밀집된 형태로 나타난 것으로 판단된다. Fig. 5.7은 추정된 속도 성분을 나타낸다. 결과를 보면 각 축 별 최대 속도는 0.3228 m/s, 0.3944 m/s, 0.1376 m/s로 플랫폼의 이동 속도가 다소 느린 것을 확인할 수 있다.

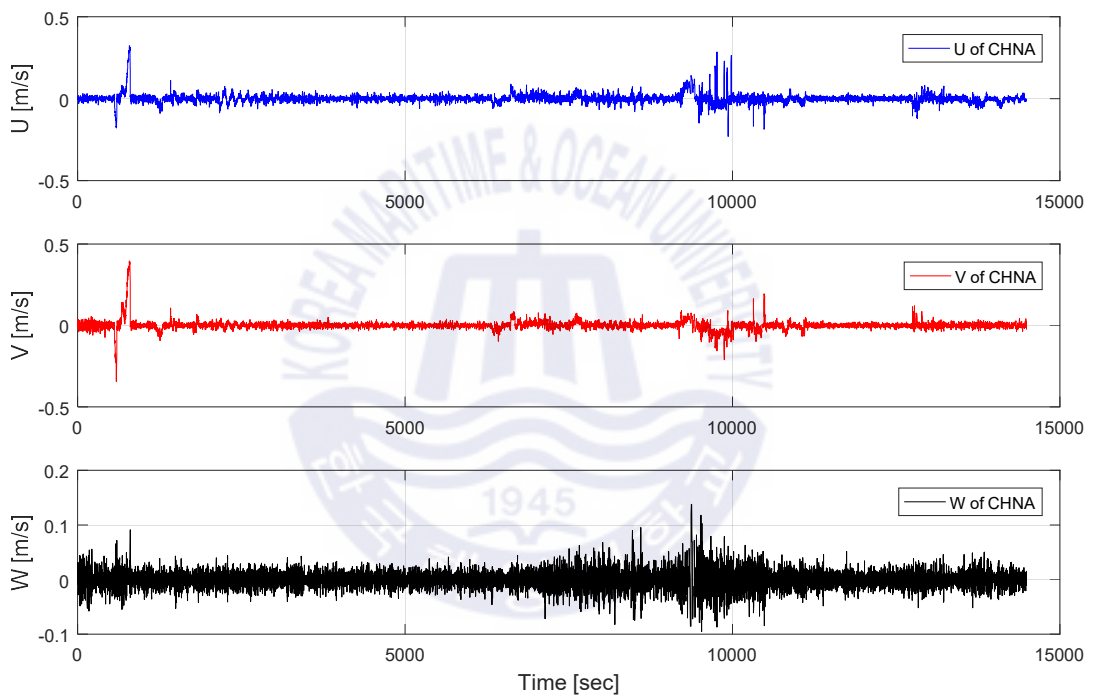


Fig. 5.7 Velocities estimated by underwater navigation system

Fig. 5.8은 USBL 측정 위치에 대한 항법 결과의 위치 추정 차이를 나타낸다. Fig. 5.7과 같이 플랫폼의 이동 속도가 매우 느리며(0.5 m/s 이하), Fig. 5.5와 같이 USBL의 블랙아웃 상황이 발생하지 않아 두 축의 차이 값이 1m 이내로 측정되었다. 차이 값이 커지는 부분은 플랫폼이 트랜칭 작업을 하는 시점으로 판단된다. 트랜칭 시에는 플랫폼의 진동이 매우 크게 발생하였다.

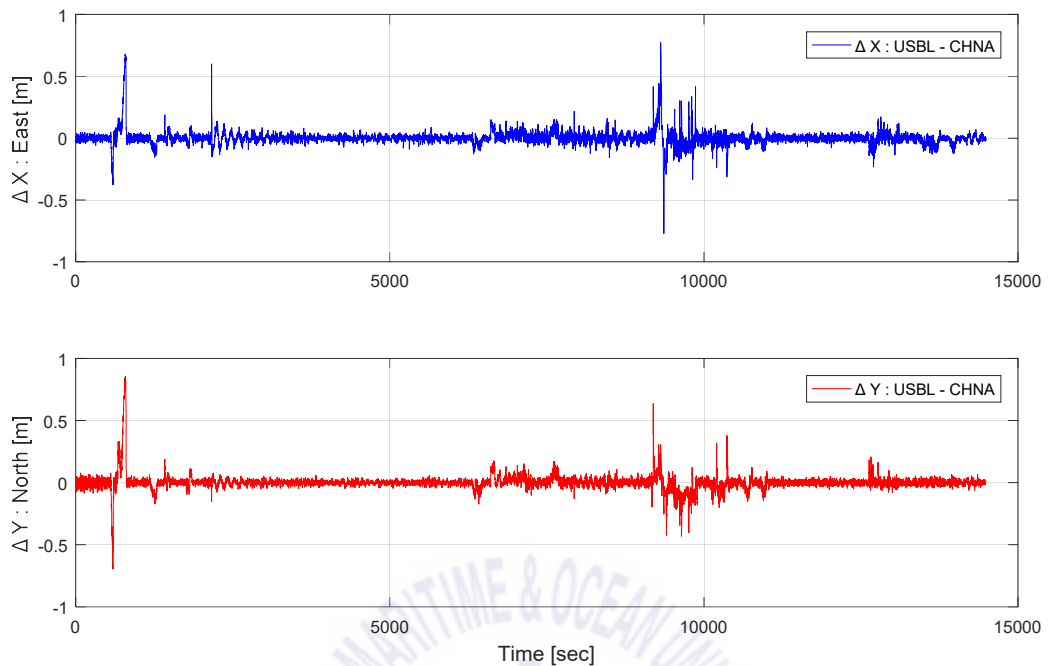
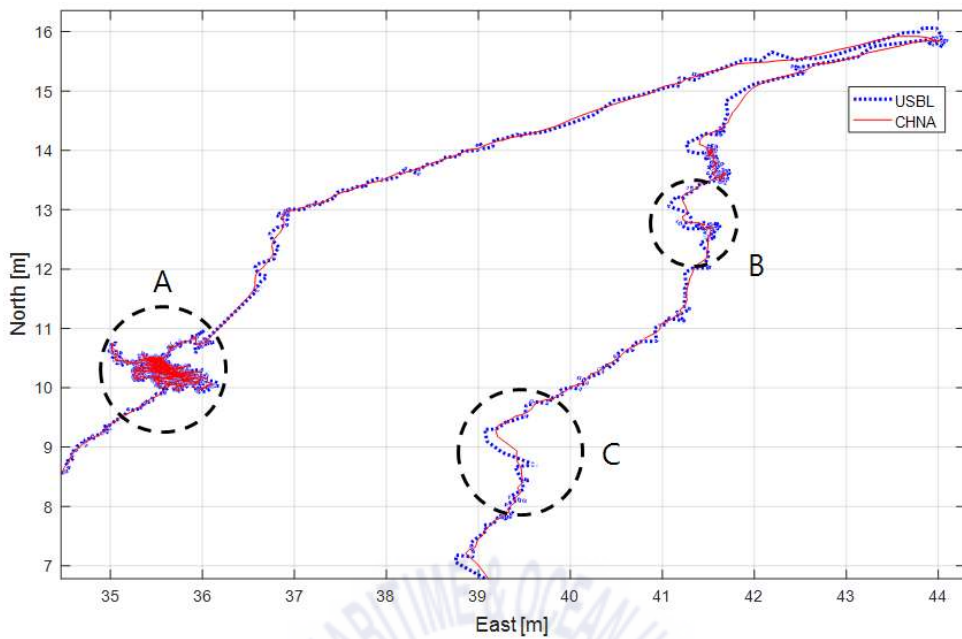
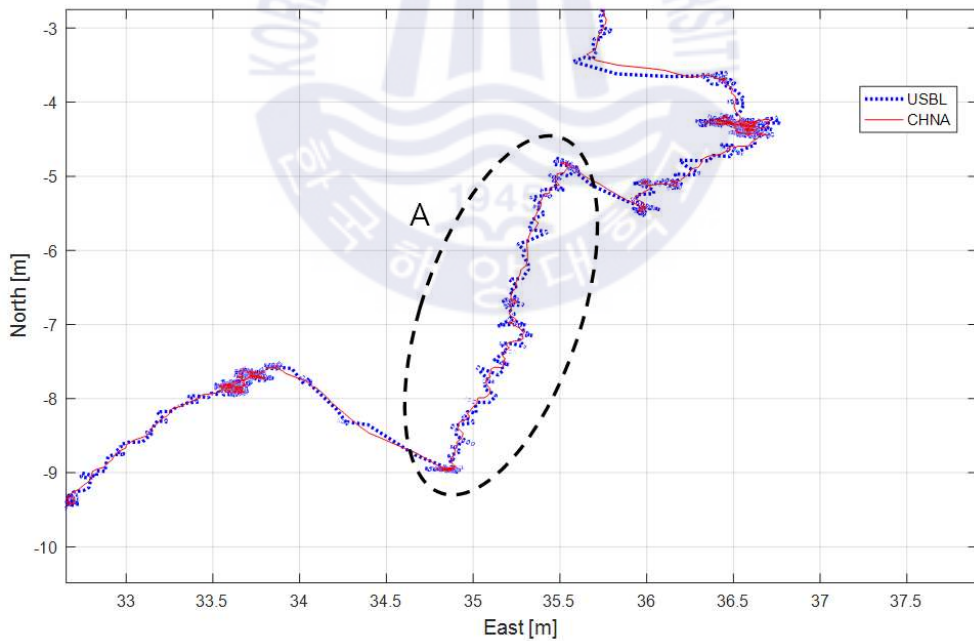


Fig. 5.8 Difference between USBL and position estimates

Fig. 5.9 (a)는 Fig. 5.6의 우측 상단부 [40, 12] m지점을 확대한 것이다. A는 플랫폼이 트랜칭 시 생기는 진동에 의해 위치 오차가 커져서 밀집된 형상이 생성된 것이다. 이 진동은 INS에도 심한 외란으로 작용한다. B와 C를 보면 USBL 데이터의 신호 상태가 좋지 않아 오차가 발생한 부분인데 제안된 수중항법 알고리즘의 결과는 이러한 부분을 상쇄하였다. B와 C 같은 지그재그 형태의 이동 경로는 덩치가 크고 무거운 상용 트랜칭 수중트랙로봇으로는 이루어질 수 없는 형태의 이동이다. 이와 같이 제안된 수중항법 알고리즘은 단독의 USBL 측정값에 오차 성분이 보정된 결과를 보여준다. 또한 Fig. 5.9 (b)는 Fig. 5.6의 우측 중앙부 [35, -6] m 지점을 확대한 것이다. A를 보면 USBL의 신호에 에러가 있어 위치 데이터의 상태가 좋지 않은 것을 볼 수 있다. 하지만 제안된 수중항법 알고리즘의 결과는 이러한 오차를 줄여주는 것을 확인하였다.



(a) Top right



(b) Center right

Fig. 5.9 Position result of navigation #2

5.2 수중트랙로봇에 적용한 융합 항법 성능 실험

본 절에서는 4장에서 서술한 수중항법 시스템과 수중트랙로봇을 이용하여 육상에서 융합 항법 성능을 실험하였다. 5.1절의 실험과는 달리 3장의 최종 알고리즘인 융합 항법 알고리즘에 엔코더의 정보를 이용하여 속도 보정을 하였으며, 그 결과를 서술하였다. 실험 지역의 위경도는 $35^{\circ} 04' 25.49''$ N, $129^{\circ} 05' 17.13''$ E이고, Fig. 5.10과 같이 원 위치에서 실험을 수행하였다. 수중에서 위치 측위를 위해 사용되는 USBL과 같은 기능을 육상에서 수행 가능한 GPS를 사용하였다. GPS의 수신 감도를 높이기 위해 주변에 높은 건물이 없는 운동장에서 실험을 수행하였다. 실험에 사용된 센서의 제원은 Table. 5.3과 같다. RLG를 이용해서 가속도와 각속도 정보를 취득하여 선체의 각도/속도/위치를 추정하며, GPS를 이용해서 추정된 위치 정보를 보정하며, Magnetic compass를 이용하여 추정된 각도 정보를 보정한다. 또한 본 논문에서 제안된 방식인 엔코더를 이용하여 속도를 보정하여 추정되는 속도와 위치의 정확도를 더욱 높여준다.

실험은 총 4번 수행되었으며, 첫 번째 실험은 항법 시스템의 기본 성능 확인, 두 번째 실험은 임의의 미션 궤적으로 주행한 결과로 제안된 항법과 DR, 엔코더 보정을 하지 않았을 시의 결과를 비교하였다. 또한 세 번째와 네 번째 실험은 같은 궤적을 다른 속도로 주행한 결과로 제안된 항법의 결과를 비교하였다.



Fig. 5.10 Location of navigation experiment

Fig. 5.11은 성능 실험 시의 모습을 나타내며, (a)는 잔디 위에서의 모습, (b)는 우레탄 트랙 위에서의 모습으로 궤적 주행 시에 서로 다른 지면을 이동하였다. 수중트랙로봇이 다른 지면으로 이동 시에 가속도계에 큰 변화가 측정되어 항법 연산 결과에 반영될 것으로 추정된다.

Table 5.3 Specifications of sensors used in ground experiments

Index		Value		Units
RLG	Operating Range	Gyro ± 1074	Accelerometer 70	$^{\circ}/\text{sec}, g$
	Scale Factor Linearity	150	300	PPM(1σ)
	Bias Repeatability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
	Bias Stability	1	1	$^{\circ}/\text{hr}, m-g(1\sigma)$
	Update Rate	100		Hz
GPS	Position Accuracy	1.3(H), 1.9(V)		m
	Update Rate	1		Hz
Magnetic compass	Measurement Range	± 80		μT
	Resolution	± 0.05		μT
	Repeatability	± 0.1		μT
	Update Rate	30		Hz
Encoder	Counts per turn	500		.
	Max. speed	24000		rpm
	Update Rate	10		Hz



(a) On the grass



(b) On the urethane track

Fig. 5.11 Environment of navigation experiment

5.2.1 기본 항법 성능 실험

첫 실험에서는 항법 시스템의 기본 성능을 확인하였다. 실험은 수중트랙로봇을 East 축 방향으로 총 20 m를 직진시켜 플랫폼의 위치를 실측하였으며, 실측한 위치와 항법 시스템의 결과를 비교하였다. 위치를 실측한 방법은 East 축에 2 m 간격으로 위치를 표기하고 플랫폼이 표기한 위치를 지날 때마다 카메라를 찍어 North 축의 위치를 측정하였다. Fig. 5.12에서 GPS는 GPS 측정 위치(긴 점선), Navi.는 제안된 항법 추정 위치(실선), No encoder는 엔코더를 통한 속도 보정이 없을 시의 항법 추정 위치(짧은 점선), DR은 DR으로 추정한 위치(점-실선), Trajectory는 실측 위치(별 실선)으로 표시되었다. 수중트랙로봇에 장착된 양측 모터의 구동 속도를 2000 rpm으로 설정하여 직선으로 이동시켰지만, 실제로는 지면의 상태와 여러 가지 환경 요인으로 인해 완전한 직선 형태가 아닌 다소 휘어진 형태로 이동하였다. 항법은 수중트랙로봇이 이동을 시작하고 멈출 때까지 약 61 s 동안 수행되었다.

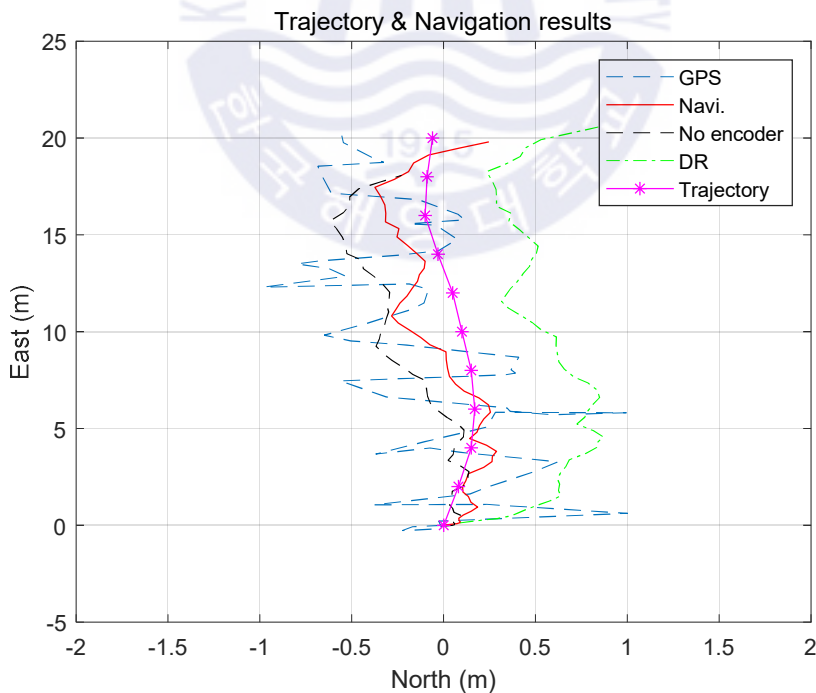


Fig. 5.12 Trajectory and navigation position results (Test 1)

Fig. 5.12의 실험 결과를 확인하면 제안된 항법의 추정 위치는 실측 위치와 유사하게 나타났지만, DR의 추정 위치는 다소 차이가 나는 것을 확인할 수 있다. DR은 초기 추정 위치부터 플랫폼의 진행 방향과 다른 방향으로 위치를 추정하며, 시간이 지남에도 참조 위치값에 수렴하지 못하는 것으로 나타난다. DR은 실험 수행시간이 길어지면 더욱 큰 에러를 갖을 것으로 판단된다. 엔코더 정보를 제외한 항법은 제안된 항법과 동일하게 GPS를 통한 위치 보정 단계를 수행하기 때문에 큰 오차를 갖지는 않지만, 추정 위치가 다소 차이가 발생한다. 실험 시에 취득된 GPS의 측정 위치를 보면 센서 자체의 수평 정밀도 때문에 다소 큰 오차를 보임을 확인하였다. 실험 결과로 제안된 항법 위치 결과가 다른 항법 결과에 비해 실측 위치와 가장 유사한 것을 확인하였다.

Fig. 5.13은 실측 위치와 제안된 항법 위치의 오차를 축별로 나타낸 것이다. North 축은 + 모양으로, East 축은 O 모양으로 표시되었다. 총 11번의 비교 결과를 산출하였으며, 산출된 결과는 Table 5.4와 같다. 축별 최대 오차는 [0.267 0.5503] m, RMS 오차는 [0.13 0.22] m로 산출되었다. 실험의 결과를 통해 제안된 항법이 RMS 오차 0.3m 이내로 위치를 추정함을 확인하였다.

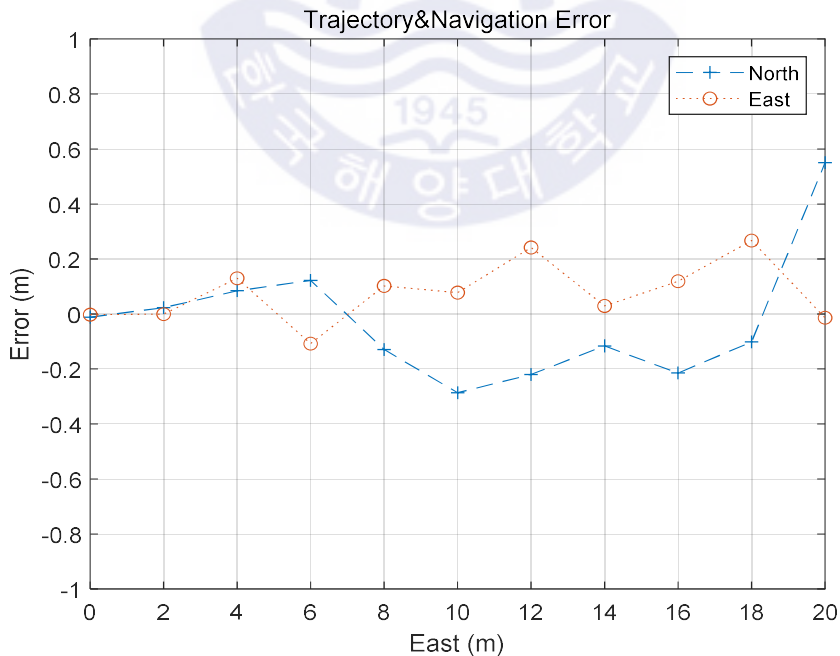


Fig. 5.13 The error between trajectory and navigation position results (Test 1)

Table 5.4 Position error result of navigation experiments

Index	Position Error		
	X	Y	Units
1	-0.0025	-0.0116	m
2	-0.0004	0.0231	m
3	0.1299	0.0843	m
4	-0.1078	0.1220	m
5	0.1021	-0.1289	m
6	0.0777	-0.2864	m
7	0.2418	-0.2198	m
8	0.0292	-0.1160	m
9	0.1189	-0.2148	m
10	0.2670	-0.1013	m
11	-0.0138	0.5503	m
RMS Error	0.13	0.22	m

5.2.2 미션 궤적 항법 성능 실험

이번 실험은 제안된 항법의 결과와 엔코더를 통한 속도 보정이 없을 시 항법의 결과, DR의 결과를 비교하였다. 수중트랙로봇에 장착된 양측 모터의 구동 속도를 3000 rpm으로 설정하고, 잔디 위에서 Fig. 5.14와 같이 [0 0] m 위치에서 출발하고 ㄱ자 모양 반복 궤적으로 이동하였으며, [27 -20] m 위치에서 정지하였다. 이와 같은 실험을 수행한 이유는 선체의 진행 방향이 좌/우로 지속적으로 변경되는 상황에서 항법의 위치 오차가 어느 정도로 추정되는지 확인하기 위함이다. 이번 실험에서는 플랫폼의 위치를 실측하기 어려워 GPS에서 측정된 위치를 참조 값으로 선정하여 항법 결과들과 비교하였다. Fig. 5.14에서 GPS는 GPS 측정 위치(긴 점선), Navi.는 제안된 항법 추정 위치(실선), No encoder는 엔코더를 통한 속도 보정이 없을 시의 항법 추정 위치(짧은 점선), DR은 DR으로 추정한 위치(점-실선)로 표시되었다. 항법은 수중트랙로봇이 이동을 시작하고 멈출 때까지 약 450 s 동안 수행되었다.

Fig. 5.14의 실험 결과를 확인하면 제안된 항법과 엔코더 정보를 제외한 항법의 추정 위치는 GPS 측정 위치와 유사하게 나타났지만, DR의 추정 위치는 다소 차이가 나는 것을 확인할 수 있다. DR은 초기 추정 위치부터 플랫폼의 진행 방향과 다른 방향으로 위치를 추정하며, 시간이 지남에도 참조 위치값에 수렴하지 못하는 것으로 나타난다. DR은 실험 수행시간이 길어지면 더욱 큰 에러를 갖을 것으로 판단된다. 엔코더 정보를 제외한 항법은 제안된 항법과 동일하게 GPS를 통한 위치 보정 단계를 수행하기 때문에 큰 오차를 갖지는 않지만, 추정 위치가 다소 차이가 발생한다.

Fig. 5.15는 Fig. 5.14의 결과를 축별로 비교한 것이다. X축과 Y축의 추정 위치는 모두 ㄱ자 형태를 나타내며, X축은 지속적으로 증가하고, Y축은 지속적으로 감소한다. 선체의 진행 방향이 바뀔 때 오차가 증가하고, 시간이 지남에 따라 수렴하는 형상이 반복됨을 확인하였다. 위치 에러는 제안된 항법의 추정 위치와 참조 위치값의 차이로 나타내었으며, 각 축별로 가장 큰 에러 정도는 X축이 약 2 m, Y축이 약 2.2 m로 확인되며, 위치 RMS 오차는 [0.98 1.29] m로 산출된다.

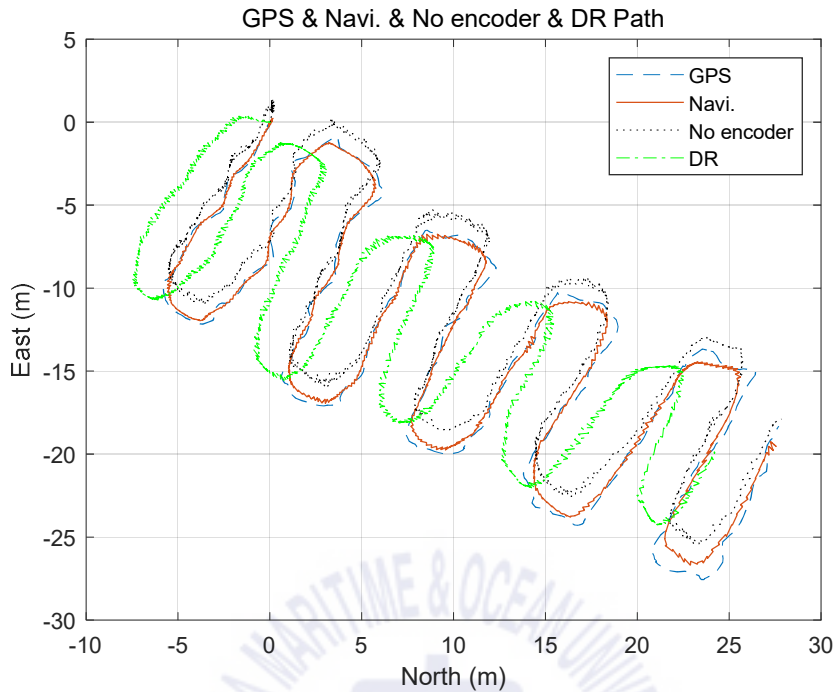


Fig. 5.14 Position results of each navigation (Test 2)

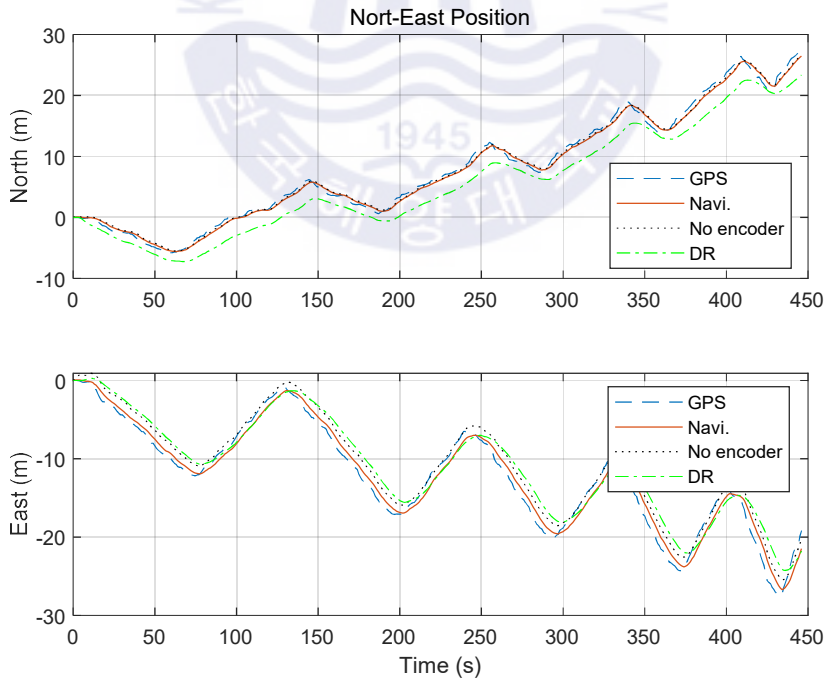


Fig. 5.15 N-E position results of each navigation (Test 2)

Fig. 5.16은 제안된 항법과 엔코더 정보를 제외한 항법, DR에서 추정된 속도 성분을 나타낸다. 제안된 항법의 속도 결과를 확인하면 각 축별 최대 속도는 $[0.46 \ 0.7] \text{ m/s}$, 평균 속도는 $[-0.07 \ 0.01] \text{ m/s}$ 인 것을 확인할 수 있다. 엔코더 정보를 제외한 항법의 속도 결과는 제안된 항법의 속도 결과보다 Y축의 속도가 다소 높게 추정이 되었으며, 더 큰 잡음 성분을 가진 것으로 나타난다. 또한 DR의 속도 결과는 초기부터 X축의 속도가 반대 방향으로 추정되었으며, 제안된 항법 X축 속도 결과에 비해 거의 모든 실험 수행시간 동안에 다소 낮게 추정이 되었다. 또한 Y축 속도는 제안된 항법의 속도에 비해 다소 높게 추정된 것으로 나타난다. 이렇게 추정 속도에서 발생한 에러들이 엔코더 정보를 제외한 항법과 DR의 최종 결과인 추정 위치의 에러값을 높이는 요인으로 판단된다.

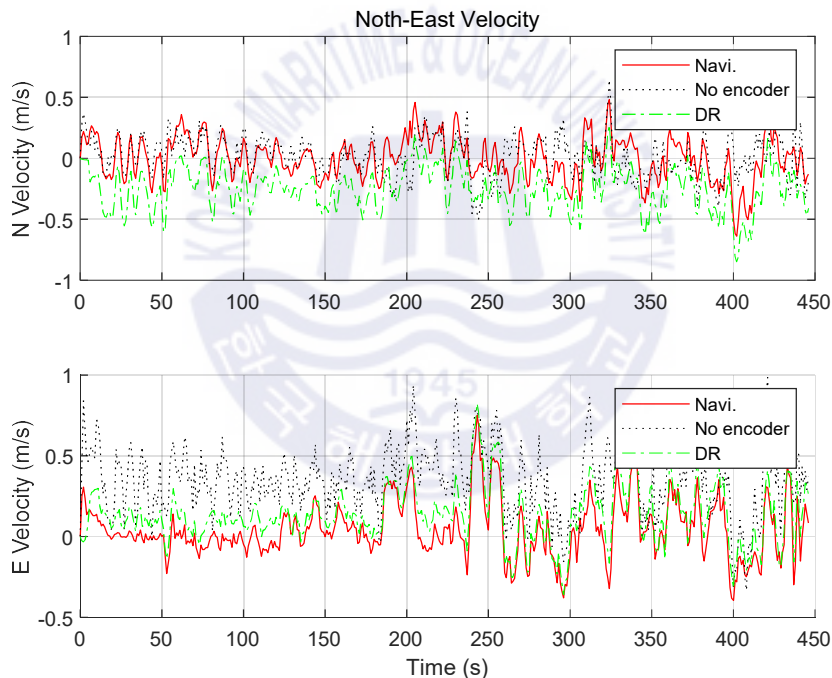


Fig. 5.16 N-E velocity results of each navigation (Test 2)

Fig. 5.17은 제안된 항법의 각도 결과이다. 각도 결과를 확인하면 실험 수행 동안에 롤과 피치는 0° 에 거의 수렴하며, 플랫폼의 진행 방향이 회전할 시에 다소 각의 변화가 나타난다. 헤딩의 경우 초기 -125° 에서 55° 로 약 180°

간격으로 선체의 진행 방향이 총 9번 변경되는 것으로 나타난다. 세부적으로는 실험 수행 중 약 80 s와 120 s, 190 s, 240 s, 290 s, 340 s, 370 s, 410 s, 430 s 지점 총 9번을 플랫폼의 진행 방향이 U턴한 것을 확인할 수 있다.

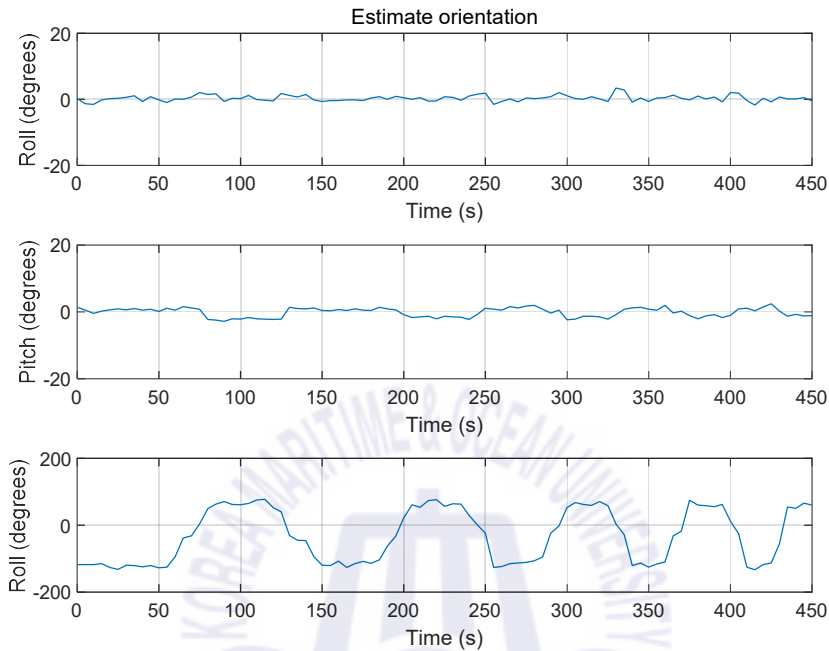


Fig. 5.17 Orientation results of navigation (Test 2)

앞선 실험들의 결과를 통해 GPS로부터 실시간으로 계측되는 위치 데이터는 큰 잡음이 포함되어있는 반면, 제안된 항법 알고리즘을 적용한 결과에서는 위치 데이터의 잡음이 훨씬 줄어든 특성을 확인할 수 있다. 또한 엔코더 정보를 제외한 항법과 DR의 결과를 제안된 항법과 비교하였을 때 제안된 항법이 위치를 더욱 효과적으로 추정함을 확인하였으며, 제안된 항법을 적용할 때의 장점을 확인하였다.

5.2.3 사각형 궤적 항법 성능 실험 #1

5.2.3과 5.2.4는 같은 궤적을 수중트랙로봇에 장착된 모터의 구동 속도를 달리 하여 실험을 수행하였다. Fig. 5.18은 제안된 항법의 추정 위치를 수평면 궤적으로 나타낸 것으로, GPS는 GPS 측정 위치(점선), Navi.는 제안된 항법의 추정 위치(실선)로 표시되었다. 이번 실험은 수중트랙로봇에 장착된 양측 모터의 구동 속도를 2000 rpm으로 설정하여 수행하였으며, 수중트랙로봇은 [0 0] m 위치에서 출발하고 반시계 방향인 사각형 궤적으로 이동하여 최종적으로는 [0 -6] m 위치에서 정지하였다. 융합 항법은 수중트랙로봇이 이동을 시작하고 멈출 때까지 약 300 s 동안 수행되었다. 실험 결과를 확인하면 항법 추정 위치가 GPS 측정 위치와 유사하게 나온 것을 확인할 수 있으며, 진동하는 형상의 데이터가 다소 밀집된 부분이 한 지점([-16 -7.5] m)이 있는데, 이는 수중트랙로봇이 우레탄 트랙에서 잔디로 이동할 때 발생한 진동의 영향으로 판단된다. Fig. 5.19는 Fig. 5.18의 결과를 축별로 비교한 것이다. Fig. 5.19는 축별 GPS 측정 위치와 항법 추정 위치의 차이를 나타내며, 각 축별로 가장 큰 에러 정도는 X축이 약 2 m, Y축이 약 2 m로 확인되며, 위치 RMS 오차는 [1.18 0.94] m로 산출된다.

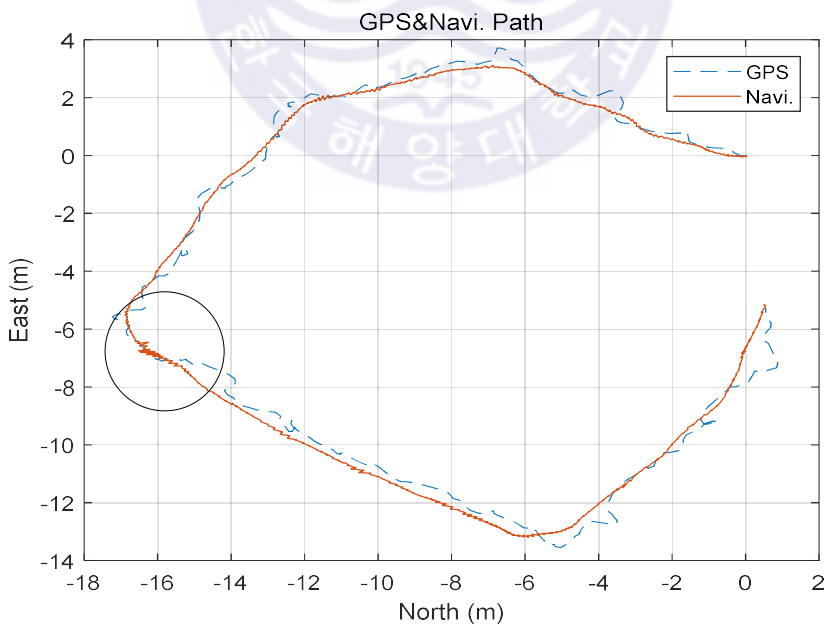


Fig. 5.18 Position result of navigation (Test 3)

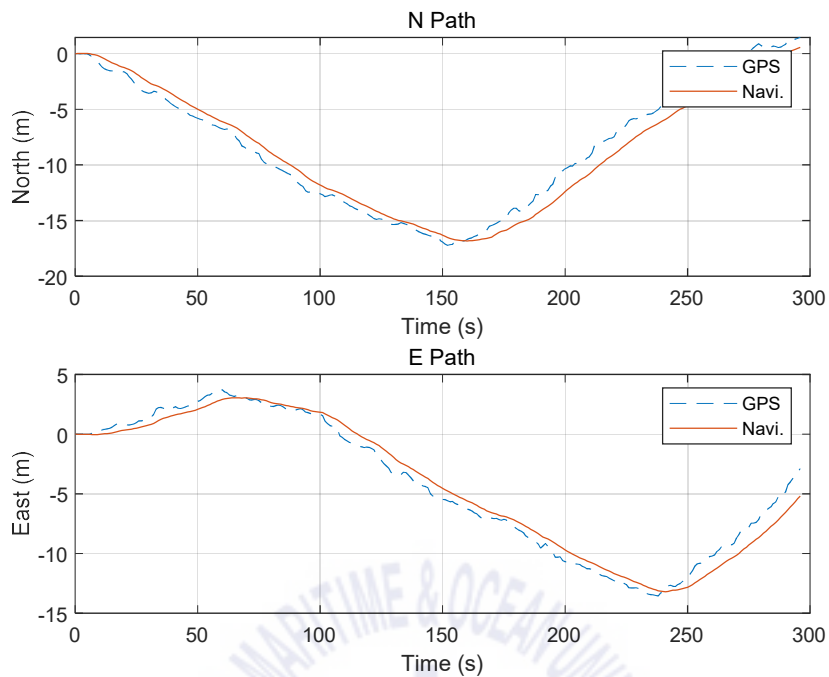


Fig. 5.19 Position result of navigation vs. GPS position (Test 3)

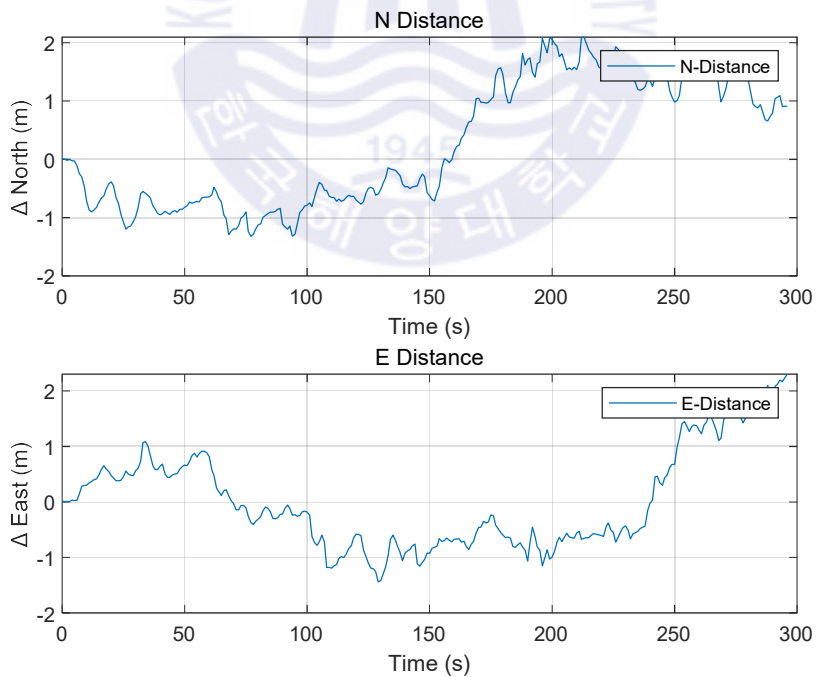


Fig. 5.20 Difference between position result of navigation and GPS position (Test 3)

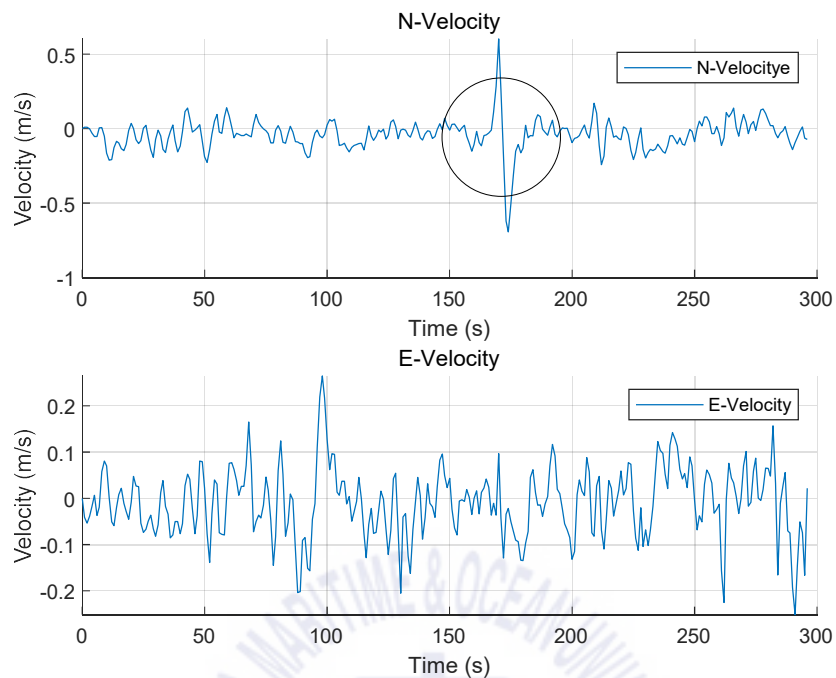


Fig. 5.21 Velocity result of navigation (Test 3)

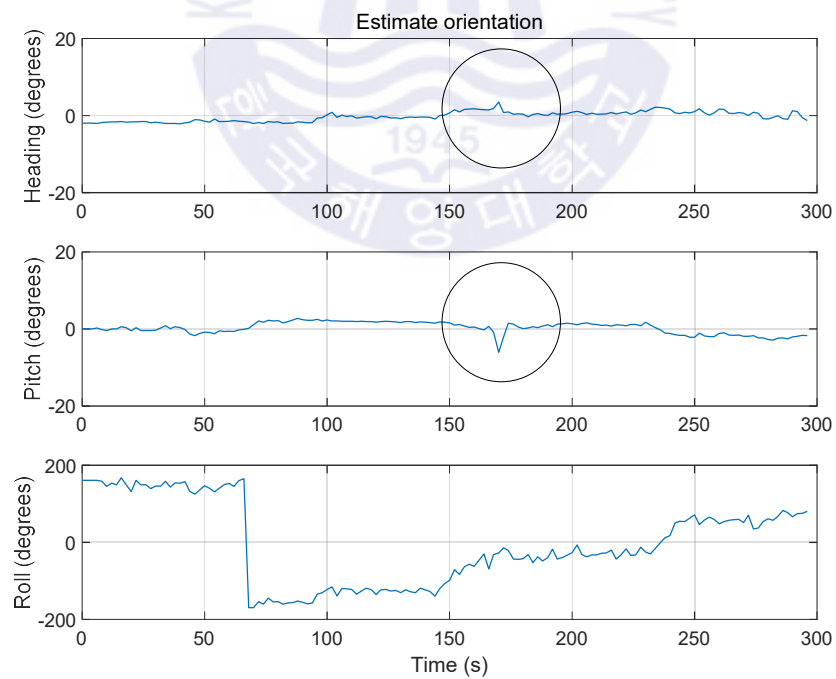


Fig. 5.22 Orientation result of navigation (Test 3)

Fig. 5.21은 제안된 항법에서 추정된 속도 성분을 나타낸다. 결과를 확인하면 각 축별 최대 속도는 $[0.6 \ 0.3] \text{ m/s}$, 평균 속도는 $[-0.05 \ -0.01] \text{ m/s}$ 인 것을 확인할 수 있다. 약 180 s경에 X축의 속도가 $\pm 0.6 \text{ m/s}$ 로 크게 변동하는 것이 확인되는데, 이는 우레탄 트랙에서 잔디로 수중트랙로봇이 이동할 때 생긴 진동으로 추정된다. 결과로 확인했을 때 우레탄 트랙과 잔디 사이에 높이의 차이가 있으며, 우레탄 트랙이 잔디보다 높은 높이에 있는 것으로 판단된다. 수중트랙로봇의 특성상 지면의 상태에 따라 선체가 진동하게 되는데 이것은 추정된 속도 결과를 보면 확인할 수 있다.

Fig. 5.22는 제안된 항법에서 추정된 각도 성분을 나타낸다. 앞선 실험 결과들과 함께 확인하면, 약 70 s와 150 s, 240 s 지점에서 수중트랙로봇의 진행 방향이 크게 변한 것을 확인할 수 있다. 또한 180 s 지점에서는 앞서 설명한 바와 같이 우레탄 트랙에서 잔디로 이동한 지점으로 헤딩과 피치가 순간적으로 변하는 것을 확인할 수 있다.

5.2.4 사각형 궤적 항법 성능 실험 #2

이번 실험은 5.2.3의 실험과 달리 수중트랙로봇에 장착된 양측 모터의 구동 속도를 3000 rpm으로 설정하여 수행하였으며, Fig. 5.23과 같이 $[0 \ 0] \text{ m}$ 위치에서 출발하고 반시계 방향인 사각형 이동하여 최종적으로는 $[0 \ -6] \text{ m}$ 위치에서 정지하였다. GPS는 GPS 측정 위치(점선), Navi.는 제안된 항법의 추정 위치(실선)로 표시되었다. 융합 항법은 수중트랙로봇이 이동을 시작하고 멈출 때까지 약 260 s 동안 수행되었다. 5.2.2의 실험과 마찬가지로 실험 결과를 확인하면 항법 추정 위치가 GPS 측정 위치와 유사하게 나온 것을 확인할 수 있다. 하지만 진동하는 형상의 데이터가 다소 밀집된 부분이 세 지점($[-17 \ -5] \text{ m}$, $[-10 \ 4] \text{ m}$, $[-12 \ -6] \text{ m}$)이 있는데, 이는 수중트랙로봇이 우레탄 트랙에서 잔디로 이동할 때 발생한 진동과 플랫폼 회전 시의 진동의 영향으로 판단된다. Fig. 5.24는 Fig. 5.23의 결과를 축별로 비교한 것이다. Fig. 5.25는 축별 GPS 측정 위치와 항법 추정 위치의 차이를 나타내며, 각 축별로 가장 큰 에러 정도는 X축이 약 2.6 m, Y축이 약 2.1 m로 확인되며, 위치 RMS 오차는 $[1.4 \ 1.04] \text{ m}$ 로 산출된다.

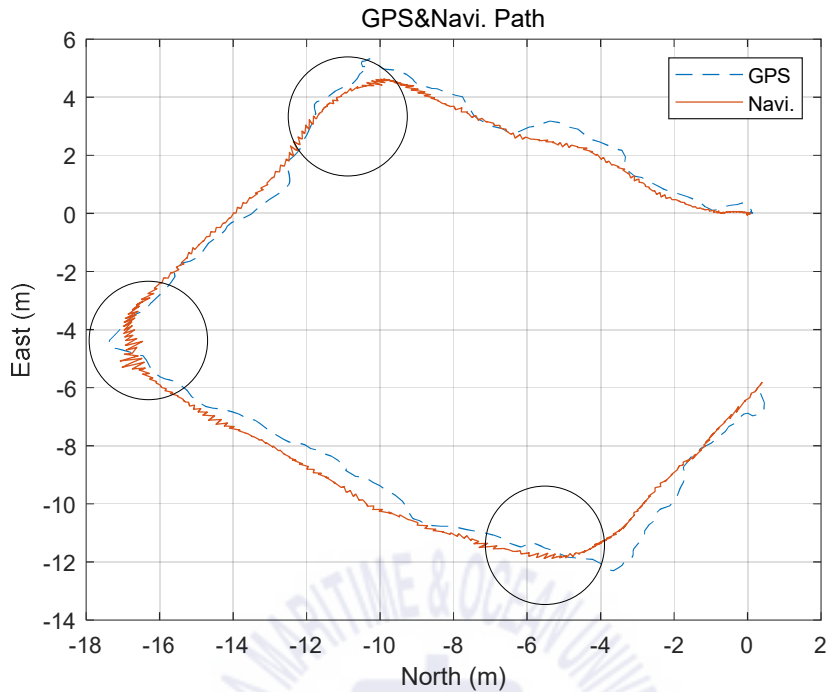


Fig. 5.23 Position result of navigation (Test 4)

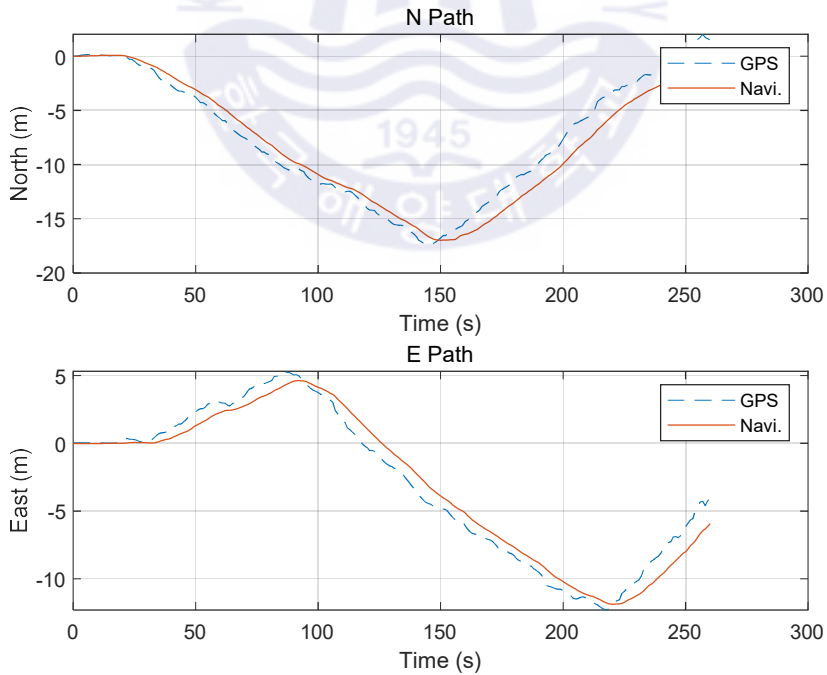


Fig. 5.24 Position result of navigation vs. GPS position (Test 4)

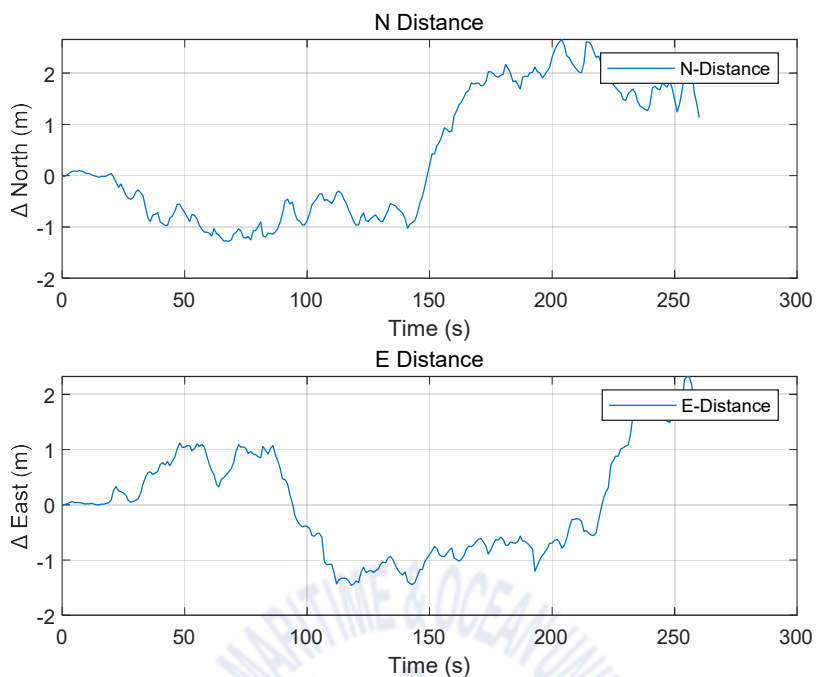


Fig. 5.25 Difference between position result of navigation and GPS position (Test 4)

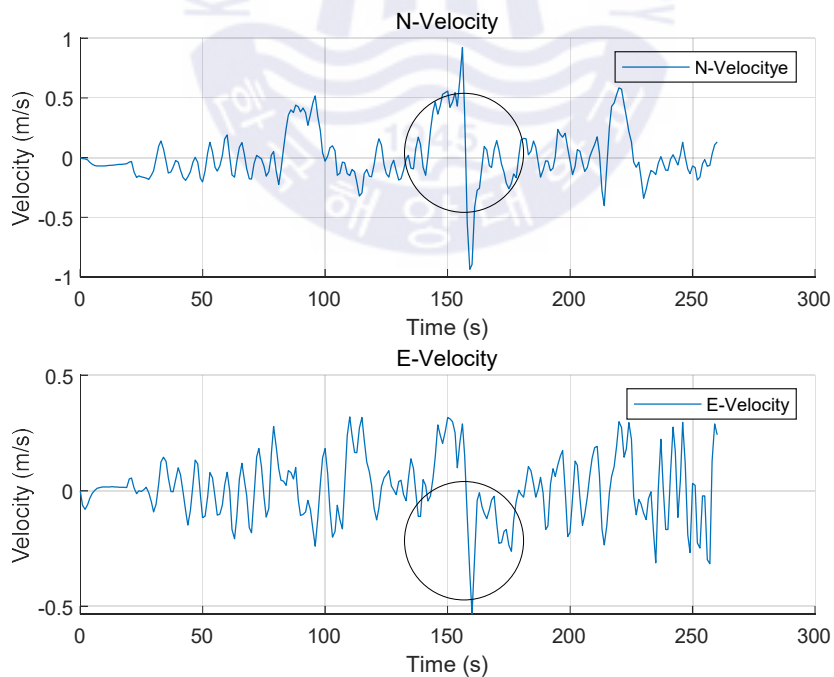


Fig. 5.26 Velocity result of navigation (Test 4)

Fig. 5.26은 제안된 항법에서 추정된 속도 성분을 나타낸다. 결과를 확인하면 각 축별 최대 속도는 $[0.9 \ -0.5] \text{ m/s}$, 평균 속도는 $[0.02 \ 0.01] \text{ m/s}$ 인 것을 확인할 수 있다. 이전 실험과 유사하게 약 160 s경에 X축과 Y축의 속도가 크게 변동하는 것이 확인되며, 속도 변화의 크기는 $[\pm 0.9 \ -0.5] \text{ m/s}$ 로 이전 실험에 비해 다소 크다. 이는 우레탄 트랙에서 잔디로 수중트랙로봇이 이동할 때 생긴 진동으로 추정된다.

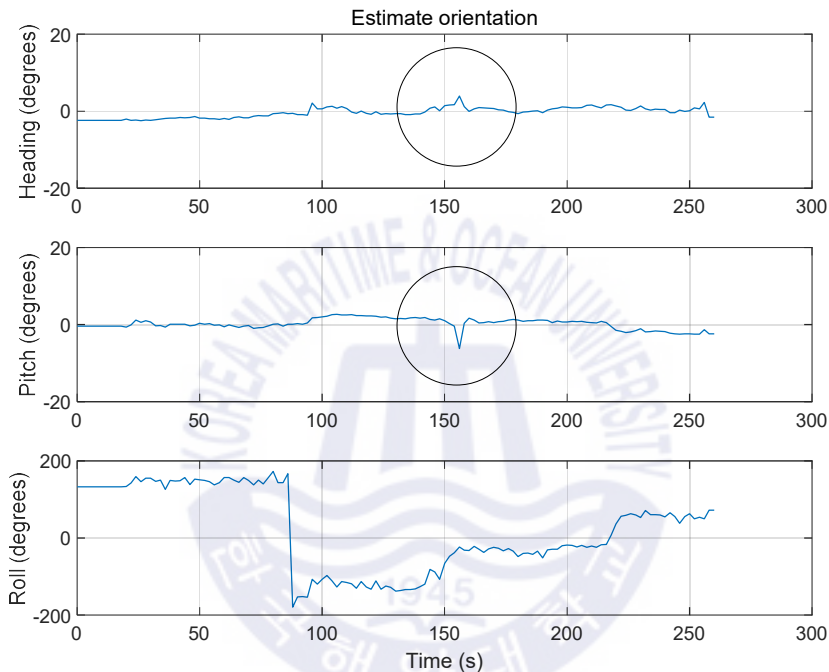


Fig. 5.27 Orientation result of navigation (Test 4)

Fig. 5.27은 제안된 항법에서 추정된 각도 성분을 나타낸다. 앞선 실험 결과들과 함께 확인하면, 약 90 s와 150 s, 220 s 지점에서 수중트랙로봇의 진행 방향이 크게 변한 것을 확인할 수 있다. 이번 실험에서 진행 방향의 변화가 이전 실험과 거의 비슷한 시간으로 나타난 것은 실험 초기에 수중트랙로봇을 약 20 s 정도 정지 후에 실험을 진행하였기 때문이다. 150 s 지점에서는 앞서 설명한 바와 같이 우레탄 트랙에서 잔디로 이동한 지점으로 헤딩과 피치가 순간적으로 변하는 것을 확인할 수 있다.

5.2.3(Test 3)와 5.2.4(Test 4)의 실험 결과를 비교하여 정리하면 Table 5.5와 같다. 같은 지면에서 같은 궤적으로 이동하지만 주행 속도를 달리한 제안된 항법 실험의 결과로 평균 위치 RMS 오차(GPS와 비교)는 약 [1.3 1] m로 산출된다. 하지만 트랙의 주행 속도를 빠르게 설정했을 시에 X축의 위치 RMS 오차(GPS와 비교)가 0.22 m 증가한 것을 확인하였다. 이는 같은 주기로 엔코더 정보를 이용한 속도 성분을 보정하지만 상대적으로 모터 구동 속도가 빨라지면 느린 속도에 비해 빠른 속도 시에 속도 추정의 정확도가 떨어지는 것으로 판단된다. 상대적으로 빠른 속도에서도 정확도가 높은 속도 추정하기 위해서는 높은 응답의 엔코더와 보정 처리 속도가 필요한 것으로 판단된다.

Table 5.5 Position RMS error result of navigation experiments

Index	Position RMS Error		
	X	Y	Units
Test 3	1.18	0.94	m
Test 4	1.4	1.04	m

제 6 장 결 론

수중트랙로봇은 해저면의 연약지반을 주행하면서 심해자원의 채집 작업이나 해양 건설작업 등을 수행하는 로봇으로, 효율적인 작업을 위해서는 정확한 위치 추정이 가능해야 한다. 이와 같이 위치를 추정하는 방식인 수중항법은 수중 로봇의 경우 일반적으로 DVL을 이용하여 선체의 속도를 추정하는데 사용된다. 하지만 수중트랙로봇과 같은 해저면에서 운용되는 플랫폼들은 센서 운용의 제약으로 DVL을 제대로 사용할 수 없다. 그렇기 때문에 고전적 추측 항법을 사용할 수밖에 없어 플랫폼을 장기간 운영 시에 자세 및 위치 값에 심각한 오류를 가지게 된다. 따라서 본 논문에서는 INS와 USBL을 결합한 수중항법 시스템에 트랙의 속도를 측정 가능한 엔코더의 정보를 이용하여 속도를 보정하는 방법의 연구를 수행하였다.

먼저 IMU에 포함되는 가속도계, 각속도계, 자력계의 수학적 모델과 각 센서의 매개변수를 이용하여 센서 모델링을 수행하였다. 각 센서의 모델은 방향, 가속도 입력, 각 센서의 매개변수를 사용하여 센서의 판독 값을 모델링 하였다.

다음으로 융합 항법 알고리즘을 설계하였다. IMU를 이용하여 INS를 구성하였으며, INS에서 도출되는 속도는 트랙의 엔코더 정보로 보정, 위치는 GPS 또는 USBL(수평)과 수심 센서(수직)로 보정, 방향각은 자력계로 보정하였다. 마지막으로 확장 칼만필터를 거쳐 최종 결과 값(속도, 위치, 방향각)을 도출한다. 그리고 제안된 융합 항법을 육상 및 실험실에서 실험 이전에 확인하기 위하여 컴퓨터 시뮬레이션을 통하여 확인하였다. 시뮬레이션에 사용된 센서는 IMU의 가속도계와 각속도계, 자력계, GPS, 엔코더 총 5개다. 시뮬레이션에 사용된 모든 센서의 파라미터는 본 논문에서 개발한 수중항법 시스템에 사용된 센서의 제원을 바탕으로 선정되었다. 시뮬레이션은 서로 다른 입력으로 3번 실시하여 융합 항법의 성능을 확인하였으며, 효과적으로 위치를 추정하였다.

다음으로 제안된 융합 항법 알고리즘의 성능 확인을 위한 실험을 수행하기 위하여 수중항법 시스템 및 수중트랙로봇의 하드웨어를 개발하였다. 개발된 수중항법 시스템은 방수가 가능한 박스 형태로 항법 연산을 위한 관성 센서들과 누수 측정을 위한 누수 센서, 통신을 위한 Ethernet Hub, 센서에 전원 공급 및 데이터 통신을 위한 인터페이스 보드가 탑재되었다. 또한 이를 운용 및 항법 연산을 수행하기 위한 운용 소프트웨어를 개발하였다. 개발된 수중트랙로봇은 육상 및 해저면에서 이동이 가능하도록 2개의 무한궤도가 장착된 구조이며, 상부에는 수중항법 시스템을 탑재할 수 있도록 설계되었다. 또한 설계 단계에서 선정된 배터리 용량에 따라 수중항법 시스템과 수중트랙로봇의 운용 가능 시간을 산출하였다. 산출 시에 DC-DC 효율(85%)을 고려하였으며, 산출 결과로 수중항법 시스템은 약 2.3 h, 수중트랙로봇은 약 1.3 h 운용 가능한 것으로 추정되었다. 따라서 480 Wh의 배터리로 수중항법 시스템과 수중트랙로봇을 같이 운용하였을 때 적어도 1.3 h 운용 가능할 것으로 추정되었다. 내부 통신은 PoE 통신으로 구성되며, 플랫폼 운용을 위한 운용 소프트웨어를 개발하였다.

성능 실험은 2가지로 수행되었는데, 첫 번째는 엔코더 속도 보정을 제외한 수중에서의 항법 실험과 두 번째는 엔코더 속도 보정을 포함한 육상에서의 항법 실험을 수행하였다. 첫 번째 실험에서는 개발한 수중항법 시스템의 간략화 모듈과 상용 수중트랙로봇을 이용하였다. 수중항법 시스템의 간략화 모듈을 사용한 이유는 상용 수중트랙로봇의 속도를 추정하기 위한 엔코더 정보를 받지 못하기 때문에 엔코더 정보를 통한 속도 보정을 제외한 항법 알고리즘을 사용하였다. 개발된 수중항법 시스템이 장착된 플랫폼을 수심 23 m인 실험해역에서 항법 실험을 실시하였다. 실험 결과로 덩치가 크고 무거운 플랫폼이라 최대 이동속도는 각 축 별로 0.5 m/s를 초과하지 못하였다. 플랫폼이 트랜칭 중이거나 고르지 못한 지면을 지날 시 즉, 진동이 심해질 시에 위치 추정 지점이 밀집되는 현상을 확인하였다. 하지만 진동이 줄어들면 다시 원래 위치를 추정하였으며, 수중항법 시스템이 효과적으로 위치를 추정함을 확인하였다.

두 번째 실험에서는 개발된 수중항법 시스템과 수중트랙로봇을 이용하여 육상에서 융합 항법 성능을 실험하였다. 제안된 융합 항법 알고리즘을 사용하였으며, GPS의 수신 감도가 높은 개활지에서 실험을 수행하였다. 실험 환경은 지

면이 우레탄 트랙과 잔디였다. 실험은 총 4번 수행되었으며, 첫 번째 실험을 통해 항법 시스템의 기본 성능이 위치 RMS 오차(실측 위치와 비교)가 0.3m 이내임을 확인하였으며, 두 번째 실험인 임의의 미션 궤적 항법 성능 실험은 ㄱ자 형태의 반복 궤적으로 주행하여 제안된 항법과 엔코더 통한 속도 보정을 제외한 항법, 그리고 DR의 결과를 비교하는 실험을 수행하였다. ㄱ자 형태의 궤적으로 실험을 수행한 이유는 선체의 진행 방향이 좌/우로 지속적으로 변경되는 상황에서 항법의 위치 오차가 어느 정도로 추정되는지 확인하기 위함이다. 제안된 항법과 이외의 항법을 비교하면, 먼저 엔코더 정보를 제외한 항법의 경우 속도 성분에 다소 에러를 가지며, 잡음 성분을 가진 것으로 확인하였다. 다음으로 DR의 경우에는 초기의 속도 성분의 방향이 다르며 지속적으로 속도 오차를 가진 것으로 확인하였다. 이렇게 추정된 속도에서 발생한 에러들이 엔코더 정보를 제외한 항법과 DR의 최종 결과인 추정 위치의 에러값을 높이는 요인으로 판단된다. 본 실험의 결과를 통해 엔코더 정보를 제외한 항법과 DR의 결과를 제안된 항법과 비교하였을 때 제안된 항법이 위치를 더욱 효과적으로 추정함을 확인하였으며, 제안된 항법을 적용할 때의 장점을 확인하였다. 세 번째와 네 번째 실험은 같은 궤적을 다른 속도로 주행하여 항법 연산의 결과를 비교하였으며, 평균 위치 RMS 오차는 약 [1.3 1] m로 산출되었다. 하지만 트랙의 모터 구동 속도를 빠르게 설정했을 시에 X축의 위치 RMS 오차(GPS와 비교)가 0.22 m 증가된 것을 확인하였다. 본 실험의 결과로 상대적으로 빠른 속도에서도 정확도가 높은 속도 추정하기 위해서는 높은 응답의 엔코더와 보정 처리 속도가 필요한 것으로 판단된다.

앞선 실험들의 결과를 통해 위치를 보정하는데 사용되는 USBL/GPS로부터 실시간으로 계측되는 위치 데이터는 큰 잡음이 포함되어 있는 반면, 제안된 융합 항법 알고리즘을 적용한 결과에서는 위치 데이터의 잡음이 훨씬 줄어든 특성을 확인할 수 있다. 또한 제안된 융합 항법의 결과를 이외의 항법(엔코더 정보를 제외한 항법, DR)과 비교하였을 때, 제안된 융합 항법이 더욱 효과적으로 속도를 추정하여 최종 결과인 추정 위치의 에러를 감소시켰다.

부 록

A-1 연철과 경철 효과

자력계는 일반적으로 전체 범위의 3 차원 회전을 통해 회전하기 때문에, 자계가 이상적일 경우 자력계의 이상적인 측정 값은 원점을 중심으로 완벽한 구를 형성해야 한다. 그러나 센서 회로 보드와 주변 환경에서 자기장이 왜곡되어 구형 자기 측정이 교란될 수 있다. 이와 같은 왜곡은 일반적으로 두 가지 효과로 분류된다.

첫 번째는 연철 효과로 Fig. A-1.1 (a)와 같이 구에서 타원체의 왜곡과 타원체의 기울기로 설명된다. 이 효과는 자기장에 영향을 주지만 자체 자기장을 생성하지 않는 교란으로 인해 발생한다. 예를 들어, 니켈 및 철과 같은 금속은 이러한 종류의 왜곡을 유발할 수 있다. 두 번째는 경철 효과로 원점에서 타원체 중심의 오프셋으로 설명된다. 이 효과는 지구 자기장에 일정한 부가 필드를 나타내는 재료에 의해 생성된다. 이 추가된 오프셋 상수는 Fig. A-1.1 (b)와 같이 연철 효과에 추가됩니다. 이 두 효과를 제거하기 위해 캘리브레이션 단계를 거쳐 자력계를 사용해야 한다(Ozyagcilar, 2012).

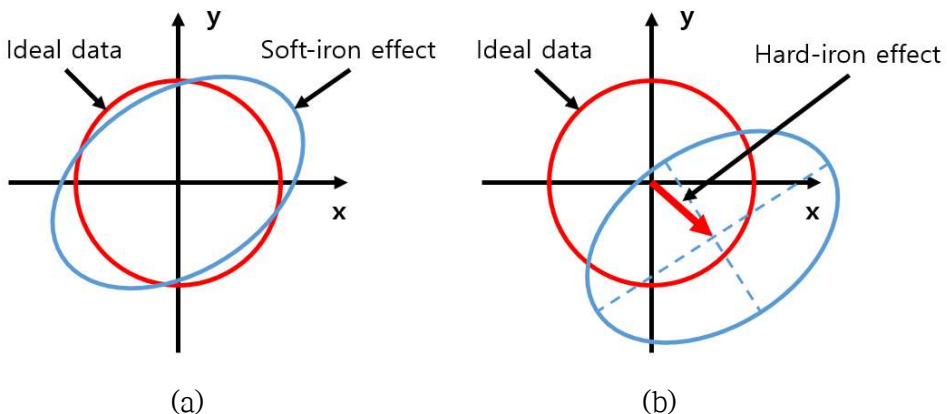


Fig. A-1.1 Soft and Hard iron effects in magnetic compass

A-1 오일러 각도의 방향 행렬

좌표계에서 오일러 각 회전 순서는 첫 번째가 요, 두 번째가 Pitch이며, 마지막이 롤이다. 롤 회전(ϕ)은 X축을 중심으로 하고, 피치 회전(θ)은 Y축을 중심으로 하고 요 회전(ψ)을 Z축을 중심으로 한다. 축별 회전 행렬은 다음과 같다.

$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \quad (\text{A.1})$$

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \quad (\text{A.2})$$

$$R_z(\psi) = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (\text{A.3})$$

지정된 회전 순서를 사용하는 회전 행렬은 다음과 같다.

$$\begin{aligned} R_{NED} &= R_x(\phi) R_y(\theta) R_z(\psi) \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \cos\phi\cos\psi + \sin\theta\sin\phi\sin\psi & \cos\theta\sin\phi \\ \cos\phi\cos\psi\sin\theta + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi & \cos\theta\cos\phi \end{pmatrix} \end{aligned} \quad (\text{A.4})$$

A-2 오일러 각도의 방향 사원수

좌표계에서 개별 오일러 각도 회전 사원수는 다음과 같다.

$$q_x(\phi) = \cos\left(\frac{\phi}{2}\right) + i \sin\left(\frac{\phi}{2}\right) \quad (\text{A.5})$$

$$q_y(\theta) = \cos\left(\frac{\theta}{2}\right) + j \sin\left(\frac{\theta}{2}\right) \quad (\text{A.6})$$

$$q_z(\psi) = \cos\left(\frac{\psi}{2}\right) + k \sin\left(\frac{\psi}{2}\right) \quad (\text{A.7})$$

사원수 $q_{zyx} = q_z(\psi)q_y(\theta)q_x(\phi)$ 는 다음과 같이 평가된다.

$$\begin{aligned} q_{zyx} &= q_z(\psi)q_y(\theta)q_x(\phi) \\ &= \left\{ \cos\left(\frac{\psi}{2}\right) + k \sin\left(\frac{\psi}{2}\right) \right\} \left\{ \cos\left(\frac{\theta}{2}\right) + j \sin\left(\frac{\theta}{2}\right) \right\} \left\{ \cos\left(\frac{\phi}{2}\right) + i \sin\left(\frac{\phi}{2}\right) \right\} \end{aligned} \quad (\text{A.8})$$

$$\begin{aligned} q_{zyx} &= \left\{ \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) \right\} \\ &+ \left\{ \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) - \sin\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) \right\} i \\ &+ \left\{ \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) \right\} j \\ &+ \left\{ \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) - \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) \right\} k \end{aligned} \quad (\text{A.9})$$

사원수 q_{zyx} 의 요소는 다음과 같다.

$$\begin{aligned}
q_0 &= \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) \\
q_1 &= \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) - \sin\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) \\
q_2 &= \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) + \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right) \\
q_3 &= \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\phi}{2}\right) - \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)
\end{aligned} \tag{A.10}$$

A-3 짐벌 락(Gimbal Lock)

짐벌 락은 두 번째 피치 회전이 첫 번째 요 및 세 번째 롤 회전으로 정렬할 때 항공 우주 좌표계에서 발생한다. 이것은 90° 피치 위쪽으로, 90° 피치 아래쪽으로 발생한다. 따라서 오리엔테이션의 오일러 각도 설명에서 자유도는 3에서 2로 감소하고 롤 및 요 각도는 그 합과 차이만으로 진동한다.

짐벌 락 $\theta = 90^\circ$ 및 $\theta = -90^\circ$ 에서 방정식 식(A.4)는 다음과 같이 단순화된다.

$$R_{NED} = \begin{pmatrix} 0 & 0 & -1 \\ -\sin(\psi - \phi) & \cos(\psi - \phi) & 0 \\ \cos(\psi - \phi) & \sin(\psi - \phi) & 0 \end{pmatrix} \text{ for } \theta = 90\text{deg} \tag{A.11}$$

$$R_{NED} = \begin{pmatrix} 0 & 0 & 1 \\ -\sin(\psi + \phi) & \cos(\psi + \phi) & 0 \\ -\cos(\psi + \phi) & -\sin(\psi + \phi) & 0 \end{pmatrix} \text{ for } \theta = -90\text{deg} \tag{A.12}$$

방향 행렬은 안정적이지만 두 각도의 짐벌 락 방향에서 차이 각도 $\psi - \phi$ 또는 합산 각도 $\psi + \phi$ 를 결정할 수 있다.

A-4 회전 행렬에서 오일러 각도 계산

식(A.4)에 정의된 회전 행렬로부터 오일러 각도를 계산하는 법을 설명한다. 3가지 오일러 각도에 대한 솔루션은 다음과 같다.

$$\phi = \tan^{-1}\left(\frac{R_{yz}}{R_{zz}}\right), \quad -180^\circ \leq \phi < 180^\circ \quad (\text{A.13})$$

$$\theta = \sin^{-1}(-R_{xz}), \quad -90^\circ \leq \theta \leq 90^\circ \quad (\text{A.14})$$

$$\psi = \tan^{-1}\left(\frac{R_{xy}}{R_{xx}}\right), \quad 0^\circ \leq \psi < 360^\circ \quad (\text{A.15})$$

자력계의 방향 각도 ρ 는 항상 요 각도 ψ 와 같다.

$$\rho = \psi \quad (\text{A.16})$$

짐벌 락에서 식(A.11) 및 (A.12)는 다음을 제공한다.

$$\tan(\psi - \phi) = \left(\frac{R_{zy}}{R_{yy}}\right) \text{ for } \theta = 90^\circ \quad (\text{A.17})$$

$$\tan(\psi + \phi) = \left(\frac{-R_{zy}}{R_{yy}}\right) \text{ for } \theta = -90^\circ \quad (\text{A.18})$$

수직 χ 로부터의 경사각은 회전된 중력 벡터의 스칼라 곱과 아래로 Z축으로부터 결정될 수 있다.

$$\cos\chi = \left\{ R_{NED} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} R_{xz} \\ R_{yz} \\ R_{zz} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = R_{zz} = \cos\theta \cos\phi \quad (\text{A.19})$$

A-5 인터페이스 회로 설계

Fig. A-5.1은 수중항법 시스템 안에 구성되는 센서 인터페이스의 전원 회로이다. 24 V 입력 전원을 전계효과 트랜지스터(Field effect transistor)를 이용하여 스위칭 회로를 구성하였으며, RLG에 입력 전원인 ± 15 V와 5 V를 두 개의 트랜지스터와 하나의 릴레이를 통해 전원 스위칭 회로를 구성한다(5 V를 ± 15 V보다 지연하여 구동). 24 V to ± 15 V와 24 V to 5 V, 총 두 종류의 DC-DC 변환기를 사용하며 입력단에는 과부하 시에 회로 손상을 방지하기 위해 퓨즈를 추가하였다. 또한 전원 입/출력단에는 전원 잡음을 상쇄시키기 위하여 커패시터들을 추가하였다. 이와 같이 DC-DC 변환기를 사용한 이유는 입력 전원과 시스템 안에서 사용될 전원을 분리(Isolation)하기 위함이다.

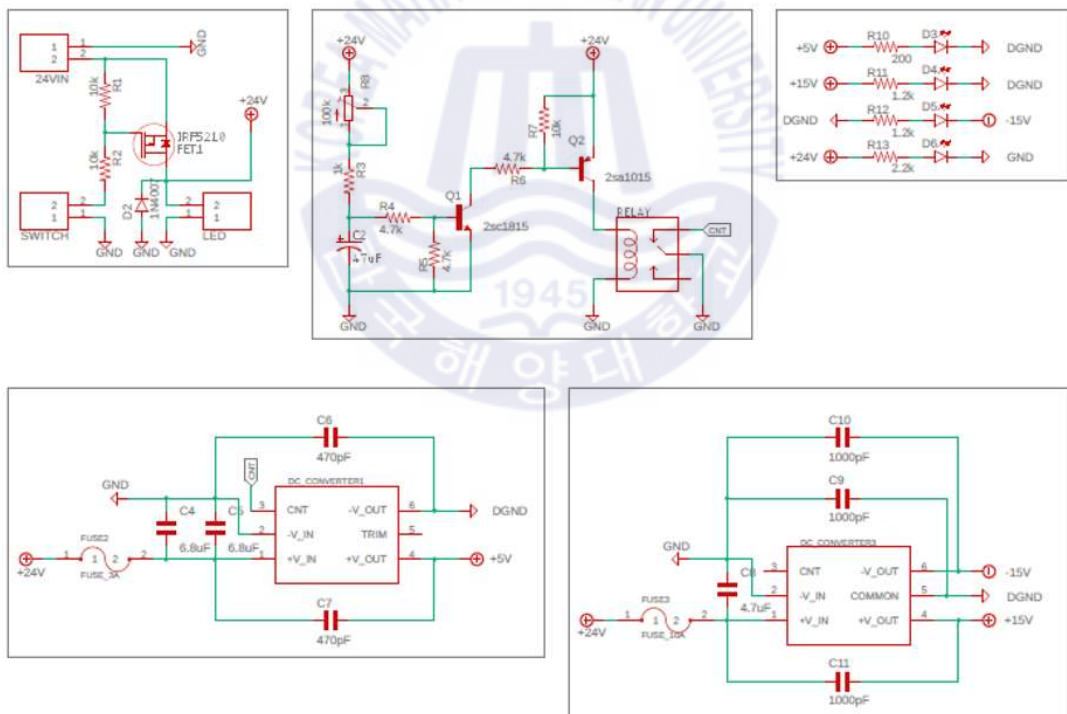


Fig. A-5.1 The power circuit of sensors interface

Fig. A-5.2는 센서 인터페이스의 통신 회로이다. 수중항법 시스템에 사용되는 각 센서들에 전원 공급과 통신을 연결하는 회로이다. 본 논문에서는 사용하지 않았지만 향후에 DVL을 장착할 수 있도록 전원 공급과 여분의 통신 포트를 확장하여 설계하였다. 누수 센서와 수심 센서는 직접적으로 통신이 연결되는 것이 아니고 회로에 ATMEL 프로세서를 추가하여 프로세서에서 센서 데이터를 받아 패킷화하여 최종 출력단으로 전송하게 설계하였다.

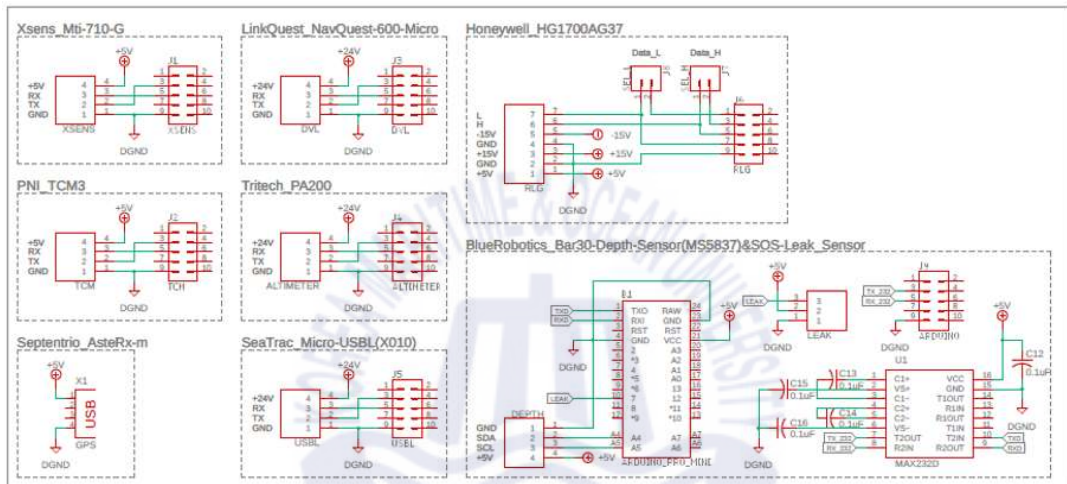


Fig. A-5.2 The communication circuit of sensors interface

Fig. A-5.3은 앞에서 설명한 센서 인터페이스 회로를 인쇄 회로 기판(Printed circuit board, PCB)로 설계한 결과물이다. 이는 Eagle 프로그램으로 설계되었으며, 수중항법 시스템의 한정된 공간에 배치하기 위하여 60×120 mm의 작은 크기로 설계되었다. 전원 잡음을 최대한 억제하기 위한 방식으로 설계를 하였으며, 회로의 윗단은 입출력 VCC를, 아랫단은 입출력 접지(GND)로 구성되었다. PCB에서 가운데를 기준으로 왼쪽은 전원 부분이며, 오른쪽은 각종 센서에 전원을 공급하고 통신을 연결하는 인터페이스 부분이다.

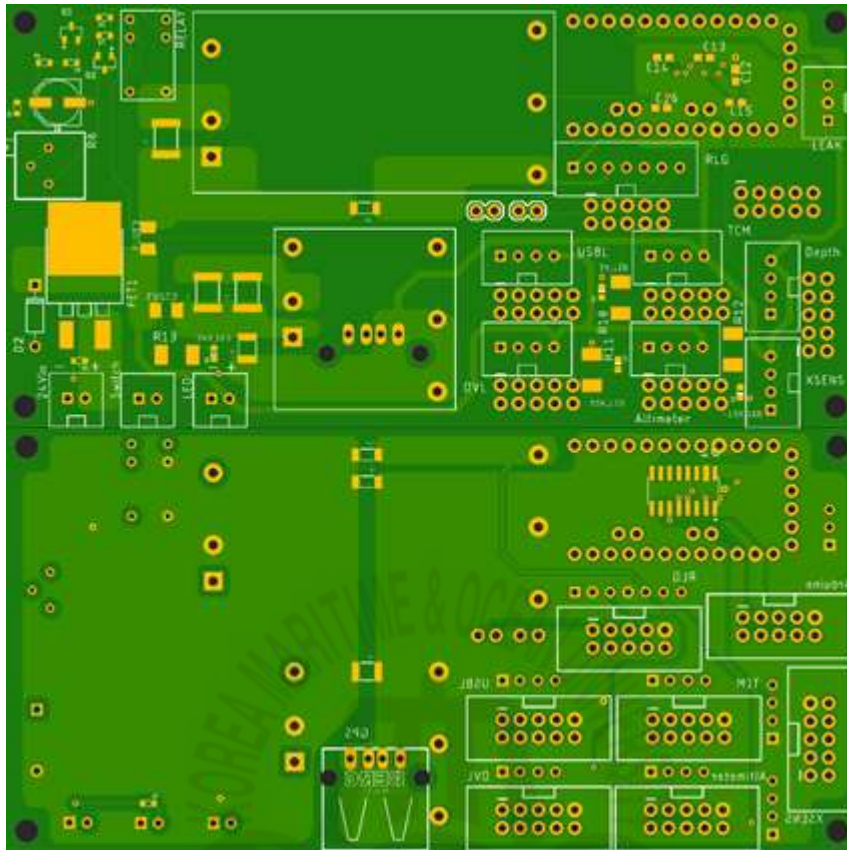


Fig. A-5.3 The PCB design of sensors interface

참고문헌

- 김용호, 2005. *MEMS 센서를 이용한 무인항공기용 자세측정장치의 특성에 대한 연구*. 석사학위논문. 인천:인하대학교.
- 김치효, 2013. *수중 건설로봇의 설계 및 제어에 관한 연구*. 박사학위논문. 경남: 창원대학교.
- 김형우, 홍섭, 2007. 심해저 연약지반 시험집광기 동적거동에 미치는 유체력 영향. *한국정밀공학회지*, 24(7), pp.19-25.
- 박성재, 2011. *심해저 망간단괴 시험집광기 제어·제측 시스템의 개발 및 실증 연구*. 부산:부경대학교.
- 박준기, 2014. *GNSS/INS/차속계/기압계를 이용한 차량용 항법 시스템 설계 및 구현*. 석사학위논문. 충북대학교.
- 윤석민, 2016. *심해저 연약지반용 무한궤도 차량의 경로추종 제어*. 박사학위논문. 부산:부경대학교.
- 이기원, 2002. 정밀 측위용 GPS와 INS의 통합 항법 시스템에 대한 연구. 석사학위논문. 대전:충남대학교.
- Alahyari, A., Rozbahani, S.G., Habibzadeh, A., Alahyari, R., & Dousti, M., 2011. INS/DVL positioning system using Kalman filter. *Aust. J. Basic Appl. Sci*, 5, pp. 1123-1129.
- Bijker, J., & Steyn, W., 2008. Kalman filter configurations for a low-cost loosely integrated inertial navigation system on an airship. *Control Engineering Practice*, 16(12), pp.1509-1518.
- Bruggemann, T.S., Greer, D.G., & Walker, R.A., 2011. GPS fault detection with IMU and aircraft dynamics. *IEEE Transactions on Aerospace and Electronic*

Systems, 47(1), pp.305–316.

Brunner, T., Lauffenburger, J.P., Changey, S., & Basset, M., 2015. Magnetometer-augmented IMU simulator: In-depth elaboration. *Sensors*, 15(3), pp.5293–5310.

Crassidis, J.L., 2006. Sigma-point Kalman filtering for integrated GPS and inertial navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 42(2), pp.750–756.

El-Sheimy, N., Chiang, K.W., & Noureldin, A., 2006. The utilization of artificial neural networks for multisensor system integration in navigation and positioning instruments. *IEEE Transactions on instrumentation and measurement*, 55(5), pp.1606–1615.

Georgy, J., Noureldin, A., Korenberg, M.J., & Bayoumi, M.M., 2009. Low-cost three-dimensional navigation solution for RISS/GPS integration using mixture particle filter. *IEEE Transactions on Vehicular Technology*, 59(2), pp.599–615.

Hostettler, R., & Sarkka, S., 2016. IMU and magnetometer modeling for smartphone-based PDR. In *2016 International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–8.

Ji, D.H., Choi, H.S., Vu, M.T., Nguyen, N.D., & Kim, S.k., 2019. Navigation and Control of Underwater Tracked Vehicle Using Ultrashort Baseline and Ring Laser Gyro Sensors. *Sensors and Materials*, 31(5), pp.1575–1587.

Karamat, T., Atia, M., & Noureldin, A., 2015. An enhanced error model for EKF-based tightly-coupled integration of GPS and land vehicle's motion sensors. *Sensors*, 15(9), pp.24269–24296.

Kim, H.W. et al., 2013. Study on Steering Ratio of Four-Row Rigid Tracked Vehicle on Extremely Cohesive Soft Soil Using Numerical Simulation. *Journal of Ocean Engineering and Technology*, 27(6), pp.81–89.

Kim, T.S. et al., 2018. Study on the Control and Topographical Recognition of

an Underwater Rubble Leveling Robot for Port Construction. *Journal of Navigation and Port Research*, 42(3), pp.237-244.

Kinsey, J.C., Eustice, R.M., & Whitcomb, L.L., 2006. A survey of underwater vehicle navigation: Recent advances and new challenges. *In IFAC Conference of Manoeuvring and Control of Marine Craft*, 88, pp.1-12.

Lee, P.M. et al., 2004. An integrated navigation system for autonomous underwater vehicles with two range sonars, inertial sensors and Doppler velocity log. *MTTS/IEEE TECHNO-OCEAN' 04*, 3, pp.1586-1593.

Lee, P.M., Jun, B.H., Kim, K., Lee, J., Aoki, T., & Hyakudome, T., 2007. Simulation of an inertial acoustic navigation system with range aiding for an autonomous underwater vehicle. *IEEE Journal of Oceanic Engineering*, 32(2), pp.327-345.

Lee, P., Kim, B., Shim, H., Baek, H., Baek, S., Park, J., & Jun, B., 2015. An Underwater Navigation System for an ROV Integrated with Inertial Sensors and USBL Position Measurements. *In Proceedings of the Fall Conference of the Korea Ocean Engineering Society*, pp. 450-453.

Liu, H., Nassar, S., & El-Sheimy, N., 2010. Two-filter smoothing for accurate INS/GPS land-vehicle navigation in urban centers. *IEEE Transactions on Vehicular Technology*, 59(9), pp.4256-4267.

Madden, M., 2006. Gravity Modeling for Variable Fidelity Environments. *In AIAA Modeling and Simulation Technologies Conference and Exhibit*, pp.6730.

MathWorks, 2018. *Sensor Fusion and Tracking Toolbox:Reference*. Massachusetts: MathWorks Inc.

Munguía, R., 2014. A GPS-aided inertial navigation system in direct configuration. *Journal of applied research and technology*, 12(4), pp.803-814.

Noureldin, A., Karamat, T.B., Eberts, M.D., & El-Shafie, A., 2008. Performance enhancement of MEMS-based INS/GPS integration for low-cost navigation

applications. *IEEE Transactions on vehicular technology*, 58(3), pp.1077-1096.

Ohlmeyer, E.J., 2006. Analysis of an ultra-tightly coupled GPS/INS system in jamming. In *2006 IEEE/ION Position, Location, And Navigation Symposium*, pp. 44-53.

Ozyagcilar, T., 2012. *Calibrating an ecompass in the presence of hard and soft-iron interference*. Freescale Semiconductor Ltd:Texas.

Penas, A.A., 2009. Positioning and navigation systems for robotic underwater vehicles. Ph.D. Tecnico of the university of Lisbon.

Qi, H., & Moore, J.B., 2002. Direct Kalman filtering approach for GPS/INS integration. *IEEE Transactions on Aerospace and Electronic Systems*, 38(2), pp.687-693.

Ribas, D., Ridao, P., Mallios, A., & Palomeras, N., 2012. Delayed state information filter for USBL-aided AUV navigation. In *2012 IEEE International Conference on Robotics and Automation*, pp. 4898-4903.

Ridao, P., Ribas, D., Hernandez, E., & Rusu, A., 2011. USBL/DVL navigation through delayed position fixes. In *2011 IEEE International Conference on Robotics and Automation*, pp.2344-2349.

Siouris, G.M., 2004. *Missile guidance and control systems*. Springer Science & Business Media:New York.

Somers, W., 2011. *Doppler-based localization for mobile autonomous underwater vehicles*. Ph.D, New Brunswick:Rutgers, The state university of New Jersey.

Sukkarieh, S., Nebot, E.M., & Durrant-Whyte, H.F., 1998. Achieving integrity in an INS/GPS navigation loop for autonomous land vehicle applications. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, 4, pp.3437-3442.

Sveinsson, A., 2012. *INS/GPS error analysis and integration*. Ph.D. School of

Science and Engineering Reykjavik University.

Syed, Z.F., Aggarwal, P., Niu, X., & El-Sheimy, N., 2008. Civilian Vehicle Navigation: Required Alignment of the Inertial Sensors for Acceptable Navigation Accuracies. *IEEE Transactions on Vehicular Technology*, 57(6), pp.3402–3412.

Titterton, D., & Weston, J., 2004. Strapdown inertial navigation technology. 2nd Ed. The Institution of Engineering and Technology: London.

Tsai, F., Chiou, Y. S., & Chang, H., 2013. A positioning scheme combining location tracking with vision assisting for wireless sensor networks. *Journal of applied research and technology*, 11(2), pp.292–300.

Wang, L., Zhang, Z., & Sun, P., 2015. Quaternion-based Kalman filter for AHRS using an adaptive-step gradient descent algorithm. *International Journal of Advanced Robotic Systems*, 12(9), pp.131.

Wang, J.H., & Gao, Y., 2010. Land vehicle dynamics-aided inertial navigation. *IEEE Transactions on Aerospace and Electronic Systems*, 46(4), pp.1638–1653.

Wang, W., Liu, Z.Y., & Xie, R.R., 2006. Quadratic extended Kalman filter approach for GPS/INS integration. *Aerospace Science and Technology*, 10(8), pp.709–713.

Wendel, J., & Trommer, G.F., 2004. Tightly coupled GPS/INS integration for missile applications. *Aerospace Science and Technology*, 8(7), pp.627–634.

Yunchun, Y. & Farrell, J.A., 2003. Magnetometer and differential carrier phase GPS-aided INS for advanced vehicle control. *IEEE Transactions on Robotics and Automation*, 19(2), pp.269–282.

Zhou, J., Knedlik, S., & Loffeld, O., 2010. INS/GPS Tightly-coupled Integration using Adaptive Unscented Particle Filter. *The Journal of Navigation*, 63, pp.491–511.

감사의 글

먼저 박사 과정 동안 연구에 매진할 수 있도록 끊임없는 격려와 지도를 해주신 최형식 교수님께 진심으로 감사드립니다. 교수님은 제가 할 수 있는 이상의 것을 펼칠 수 있게 해주셨고, 조언과 충고를 아끼지 않으셨습니다. 많은 가르침을 주시고 늘 응원해주시는 최형식 교수님이 계셨기에 박사 생활을 잘 마무리할 수 있었습니다. 그리고 바쁘신 가운데 저의 논문 심사를 맡아주시고, 소중한 조언과 충고의 말씀을 해주신 김준영 교수님, 유삼상 교수님, 주문갑 교수님, 조용성 교수님께 깊은 감사를 드립니다.

학부생 시절 처음 연구실에 들어와 박사를 졸업하는 지금까지 약 6년의 시간이 지났습니다. 하지만 이 기간이 제가 느끼기엔 정말 짧고도 빠르게 지나간 것처럼 느껴집니다. 처음에는 단지 수중로봇을 만들 수 있으면 참 멋지겠다는 생각으로 연구실에 들어와 연구실 생활을 시작하였습니다. 긴 시간 동안 많은 일들이 있었지만 지금과 같은 연구실 생활을 할 수 있었던 이유는 제 옆에서 도움을 주신 실험실 여러 선배님들과 진일형, 상기형, 지윤형, Vu, Duc, 현중, 현준, 한솔, 지형, 승현, 명준, 기범, Huang 그리고 실험실 식구와 마찬가지로 선박의장연구실의 승민, 현석, 승규 덕분인 것 같습니다. 지면을 통해서 언급하지 못한 분들을 포함하여 그동안 옆에서 많은 도움을 주셔서 감사드립니다.

마지막으로 가슴 벅찰 만큼 행복한 이 세상을 알게 하시고, 보게 하시고, 느낄 수 있게 해주신 저의 부모님인 아버지, 어머니께 정말 감사드립니다. 거리는 멀리 떨어져 있었지만 항상 마음속으로 응원하시고 기도해주셨습니다. 그리고 저를 항상 믿어주고 제 앞길을 같이 고민하고 인도해준 윤희 누나는 무엇을 주어도 아깝지 않습니다.

사랑하는 가족들 감사드립니다.

2020년 1월

지 대 형 올림