

이종 운영체제 장비를 통합한 DDS 기반 선박 IPMS 시뮬레이터 성능 분석

Performance Analysis of Ship IPMS Simulator Based on DDS integrated with Different Operating System Equipment

오성원*

국립목포해양대학교 해군사관학부

Seongwon Oh*

Division of Naval Officer Science, Mokpo National Maritime University, Jeollanam-do, 58628, Korea

[요 약]

선박의 자동화 수준이 높아지고 자율운항선박의 개발에 따라 선박의 IPMS (integrated platform management system)는 다양한 운영체제로 통제되는 개별 장비들을 통합하고 다량의 데이터를 실시간으로 처리해야 한다. 이러한 상황에서 데이터 처리를 위해 DDS (data distribution service) 미들웨어 기반 선박 IPMS 시뮬레이터를 개발하고 성능을 평가하였다. 이종 운영체제로 동작하고 있는 신호입출력장치와 IPMS 서버에서 초당 1,000개의 토픽을 비동기로 발간하고 구독한 시뮬레이션을 수행하였으며, 초기 4초 동안 일부 토픽의 누락을 제외하고는 오류 없이 데이터를 수신하였다. 개발된 시뮬레이터는 실제 선박에서 DDS를 이용하여 다른 제어체계를 통합할 수 있음을 보여준다.

[Abstract]

With increasing automation in ships and the development of autonomous ships, an IPMS (Integrated Platform Management System) in ship needs to integrate and process large amounts of real-time data from various equipment operating on different operating systems. A ship IPMS simulator based on data distribution service (DDS) was developed and its performance was evaluated to handle data processing similar in real ship environment. Errors were monitored while 1,000 topics/sec were asynchronously published and subscribed from data acquisitive units (DAUs) and an IPMS server operating on different operating systems. Except for the loss of some topics during the initial 4 seconds, topics were received without an error thereafter. The developed simulator demonstrates the feasibility of using DDS to integrate various control systems using different operating systems in actual ships.

Key word : Data distribution service, Integrated platform management system, Middle-ware, Operating system, Simulator.

<https://doi.org/10.12673/jant.2024.28.2.210>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 22 February 2024; Revised 27 April 2024

Accepted (Publication) 29 April 2024 (30 April 2024)

*Corresponding Author: Seongwon Oh

Tel: +82-61-240-7213

E-mail: osw123@mmu.ac.kr

1. 서론

선박은 추진기관, 감속기, 펌프와 같은 많은 장비와 설비를 효율적으로 감시하고 제어하기 위해서 통합플랫폼감시제어체계 (IPMS; integrated platform management system)에 통합하여 제어하고 있으며, 이는 ICMS (integrated control and monitoring system), IMCS (integrated monitoring and control system), IACS (integrated automation and control system), ECS (engineering control system) 등으로도 언급되고 있다[1],[2].

최근 선박의 자동화가 증가하고 원격제어선박이나 자율운항선박 등의 개발에 따라 많은 센서와 장비들이 IPMS에 연동되거나 통합되고 있으며, 항해계통장비와 센서들도 통합선교시스템 (IBS; integrated bridge system) 또는 통합항해시스템 (INS; integrated navigation system) 등과 연동되거나, 이를 IPMS에 통합하고 있다 [2].

그림 1은 선박의 일반적인 IPMS 구성도이며, 메인 서버, 이더넷 네트워크, 신호입출력장치 (DAU; data acquisition unit) 및 각각의 필드 장비로 구분할 수 있다[2]. 여기서 IPMS의 네트워크는 자동화와 자율운항기술의 발전에 따라 IBS의 네트워크 또는 다른 제어체계의 네트워크와 연동될 것이다. 기 개발된 장비들의 제어장치들은 각각 다양한 운영체제에 기반하여 개발되었기 때문에 하나의 운영체제로 통합하는 것은 비용이나 개발기간 측면을 고려할 때 거의 불가능하다. 따라서, 이를 변경하지 않고 최소한의 노력으로 통합하는 것은 선박 전체를 감시 제어하기 위한 주요 연구과제가 된다.

IPMS는 엔진과 같은 필드 장비와 DAU 간에는 일반적으로 산업용 프로토콜(예, Modbus, CAN)로 장비 데이터를 전송하고, 필수적인 신호에 한해 아날로그 신호를 전송하고 있으나,

선박의 자동화와 디지털화가 진행됨에 따라 아날로그 신호에 의한 장비 데이터 전송은 점점 줄어들고 있다. 대부분의 데이터는 실질적으로 DAU와 IPMS 서버 간에 고정 IP를 사용하는 이더넷을 기반으로 전송된다. 그러나 자율운항선박 기술의 개발, 승조원 감소에 따른 자동화의 증대, 태블릿과 같은 포터블 디바이스에 의한 선박의 감시 제어, 사물 인터넷 등 다양한 기술과 개념이 적용됨에 따라 선박의 IPMS에 연결되는 장비가 대폭 증가하며, 시간에 따라 연결되는 장비 수도 변할 것이다. 따라서 기존의 고정 IP에 기반한 IPMS는 연결되는 장비가 증가할 때마다 IPMS를 대폭 수정해야 하는 단점이 있어 이에 대한 보완이 필요하다.

DDS (data distribution service) 미들웨어는 OMG (Object Management Group) [3]에서 표준으로 정의한 통신 미들웨어로써 발간 (publish) / 구독 (subscribe) 방식을 기반으로 미들웨어 간에 데이터를 교환한다. DDS는 불특정 다수의 발간자 (publisher)와 구독자 (subscriber)가 네트워크에 참여하며 발간자가 발간한 정보를 구독자가 구독함으로써 데이터의 교환이 이루어진다. 교환되는 메시지는 별도의 서버를 거치지 않고 발간자와 구독자 사이에서 직접적으로 전달된다. 따라서 이러한 발간/구독 모델은 많은 수의 단말에서 동시 다발적으로 데이터 통신이 발생하고 실시간성을 요구하는 데이터 교환이 많은 시스템에 적합하다[4]. 요약하면 DDS는 분산 환경에서 실시간 통신을 위한 발간/구독 방식을 지원하며, 성능, 확장성 그리고 가용성 측면에서 강점을 가진다. 이러한 장점 때문에 국방, 무인체계, 의료, 철도 관제, 스마트 그리드와 같은 다양한 산업에서 이용되고 있다 [5]-[8].

DDS는 다수의 송신자와 수신자가 참가하고 실시간 데이터 처리가 필요한 환경에 적합하므로, 선박의 IPMS에 점차 많은

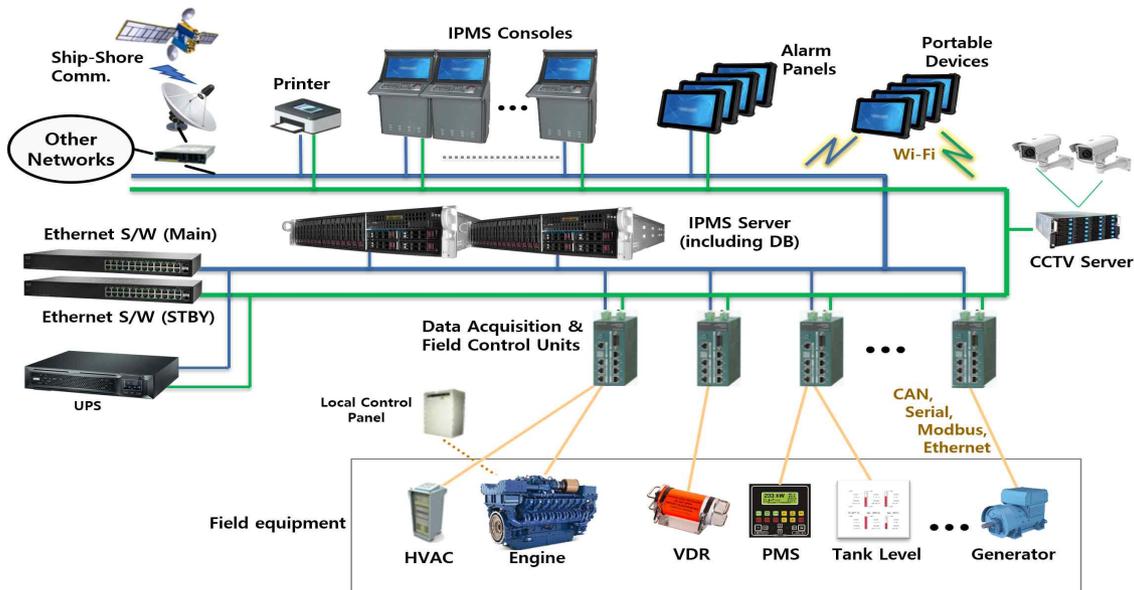


그림 1. 선박 IPMS 구성도

Fig. 1. Ship IPMS (integrated platform management system) configuration drawing.

필드 장비와 센서류가 통합되고 있는 상황에서 확장성에 대비한 효과적인 애플리케이션이 될 수 있으나, DDS와 같은 미들웨어의 적용보다는 기존 프로토콜을 이용한 개별 장비의 제어 로직 [9]이나 검증을 위한 시뮬레이터 개발에 국한되었다 [10].

본 연구에서는 선박의 자동화가 증가하고 자율운항선박 개발에 따라 선박 내의 많은 기기들이 IPMS에 통합되거나 연동될 것이므로, 서로 다른 운영체제로 동작하고 있는 기기들을 효율적으로 통합하기 위해 DDS 미들웨어를 적용하여 선박의 IPMS 시뮬레이터를 개발하고 다량의 데이터가 통신되는 상황에서 그 성능을 확인하기 위해 수행하였다.

II. DDS 미들웨어

DDS는 분산 환경에서 장비나 소프트웨어 컴포넌트를 포함하는 노드 간에 데이터를 실시간으로 교환하기 위한 발간 및 구독 방식의 통신 미들웨어의 일종으로서 복잡한 프로그램을 단순화하는 네트워크 미들웨어이며, 그 개괄적인 구성은 그림 2와 같다.

DDS는 데이터 공유 영역인 도메인(domain)으로 구분되며, 도메인 내에는 하나 이상의 참여자 (participant)를 포함하고 있다. 참여자는 하나 이상의 발간자 (publisher)와 구독자 (subscriber)를 포함할 수 있으며, 발간자는 전송할 데이터를 생성하고 배포하며, 구독자는 토픽을 수신한다. 이때 발간자와 구독자는 동일한 dds 네트워크 도메인 (domain) 범위 내에서 통신을 하게 되며, 송수신되는 데이터는 토픽 (topic)으로 정의된다 [3]. 선박의 IPMS에서 토픽은 대부분 온도, 회전수, 전류, 전압 등과 같은 센서 데이터와 On/Off, 셋팅치와 같은 명령 데이터가 대부분을 차지할 것이다.

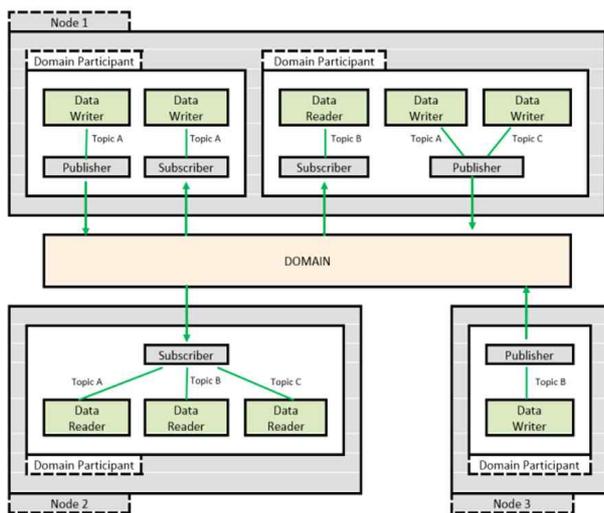


그림 2. DDS 구성요소 및 통신 흐름도 [11]
 Fig. 2. DDS entities and communication flow [11].

DDS 표준은 응용프로그램에 제공하는 데이터 발간/구독 기능 인터페이스인 DCPS (data centric publish / subscribe)와 네트워크 계층 통신 프로토콜인 RTPS (real-time publish / subscribe) 두 가지를 지원한다 [5].

III. 시뮬레이터 구성 및 실험

향후 선박의 IPMS에는 이종의 운영체제에 기반한 개별 제어 장비나 체계들이 IPMS 네트워크로 통합되거나 연동될 것이다. 이를 고려하여 각기 다른 운영체제를 기반으로 개발된 감시 제어체계가 서로 연동될 경우를 가정하여 시뮬레이터를 구성하고 DDS 네트워크상에서 임의로 발간한 다량의 토픽들의 수신 오류를 측정하는 데 중점을 두었다.

3-1 시뮬레이터 구성

시뮬레이터는 윈도우를 기반하는 IPMS 서버와 윈도우 또는 리눅스를 기반으로 하는 제어시스템이 DDS 미들웨어를 기반으로 각각 토픽을 발간하고 구독할 경우를 고려하여 구성하였다.

그림 3은 IPMS 시뮬레이터의 데이터 흐름을 나타낸다. 엔진이나 펌프와 같은 필드 장비를 모사하는 마이컴에서 시리얼 데이터 (RS-485)를 생성하여 신호입출력장치로 전송하고, 신호입출력장치에서 토픽을 발간하여 IPMS 서버에서 구독하는 방식이다.

그림 4는 전체 IPMS 시뮬레이터 구성도이며 이에 대한 사양은 표 1에 나타나 있다.

하드웨어는 IPMS 서버 역할을 하는 랩탑 1대, 신호입출력장치 역할을 하는 랩탑 1대와 마이컴 1대 및 신호입출력장치 역할을 하는 라즈베리파이 3대 및 이더넷 스위치 1대로 구성되어 있다.

소프트웨어는 서버와 신호입출력장치의 운영체제인 윈도우 10과 신호입출력장치 운영체제인 리눅스, 그리고 미들웨어로 OpenDDS를 선정하였다.

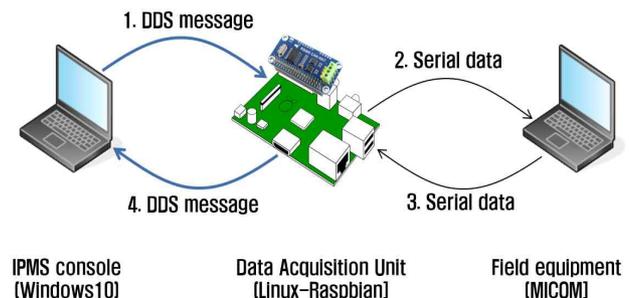


그림 3. IPMS 시뮬레이터 데이터 흐름
 Fig. 3. Data flow in IPMS simulator.

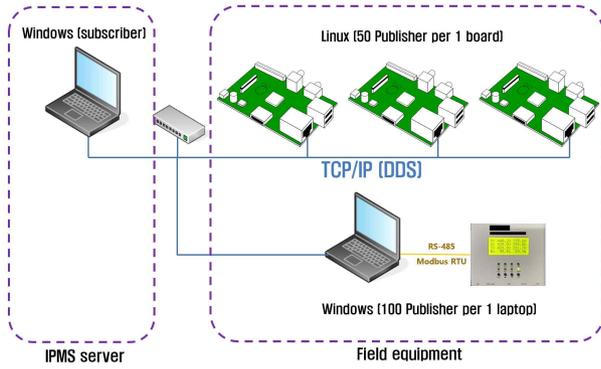


그림 4. DDS에 기반한 시뮬레이터 구성
 Fig. 4. Configuration of simulator based on DDS.

표 1. 시뮬레이터 구성 및 제원

Table 1. Simulator configuration and specification.

Item(OS)	Specification	Quantity
IPMS server (Windows10 64bit)	Laptop (Intel Core i5-1135G7)	1
DAU (Raspbian 64bit)	Raspberry Pi 4 (Cortex-A72)	3
DAU (Windows10 64bit)	Laptop (Intel Core i5-1135G7)	1
Ethernet switch	1 Gbps 8 port	1
Middle-ware	OpenDDS 3.20	-

DDS는 일반적으로 RTI사의 Connex DDS, PrismTech사의 Vortex OpenSplice DDS, Twinoaks Computing사의 CoreDX DDS 등이 많이 이용되고 있으며, 함정이나 항공기의 전투관리 체계나 무인체계의 체계통합과 같이 고성능을 요구하는 분야에 주로 적용되고 있다. 본 연구에서 DDS는 선박 IPMS가 상대적으로 고성능을 요구하지 않고 재배포가 가능하며 비용 등을 고려하여 오픈소스인 OpenDDS를 적용하였다. OpenDDS도 마찬가지로 OMG의 표준을 따르고 있다.

OpenDDS는 중앙화된 디스커버리를 지원하는 DCPS (data-centric publish-subscribe) 계층과 탈중앙화된 디스커버리 메커니즘을 운용하는 RTPS (real-time publish subscribe) 계층을 모두 지원하며, 이번 연구에서는 RTPS 계층을 이용하여 IPMS 시뮬레이터를 구현하였다.

3-2 실험 및 결과

본 연구는 DDS를 기반으로 개발된 선박의 IPMS가 이중의 운영체제로 제어되는 필드 장비들이 다량의 센서 데이터를 동시에 무작위로 송신하는 것을 시뮬레이션하고, 이를 IPMS 서버에서 데이터를 수신하는데 오류가 있는지를 측정하는 것을 목표로 하였다. 여기서 오류는 프로세스별로 송신한 토픽을 누락하거나 순서가 바뀌어 처리한 것으로 정의하였다.

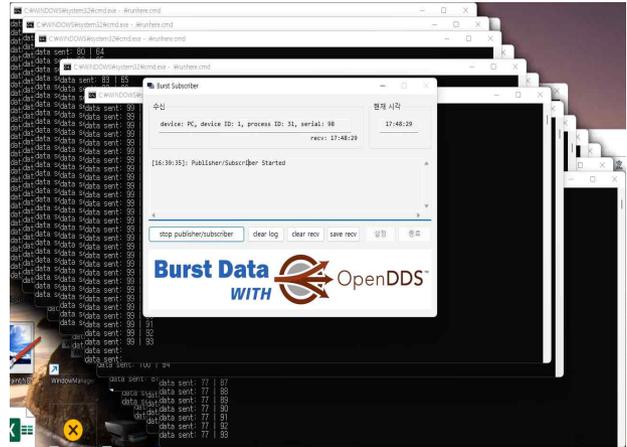


그림 5. 랩탑(윈도우10)의 토픽 발간 프로세스 화면
 Fig. 5. Topic publishing process screen on a laptop (Windows 10).

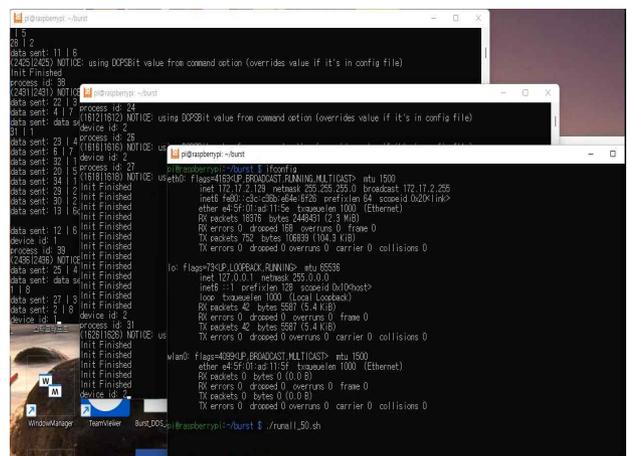


그림 6. 라즈베리파이(리눅스)의 토픽 발간 프로세스 화면
 Fig. 6. Topic publishing process screen on a Raspberry Pi (Linux).

센서 데이터는 신호입출력장치 랩탑 1대에서 100개의 토픽으로 발간하고, 라즈베리파이 3대는 각각 50개의 토픽을 발간하는 것으로 구현하였다. 총 250개의 토픽이 각각 별도의 운영체제의 독립된 프로세스에서 송신되며, 송신간격은 250 ms로 설정하여 1초에 1,000개의 토픽을 송신하였다. 토픽 구성은 장비 ID, 프로세스 ID, 일련번호, 생성 시간 (온도와 같은 장비 상태 데이터 모사)을 포함하며, 토픽 1개의 크기는 50 byte로 설정하였다.

그림 5는 윈도우10 기반의 랩탑에서 토픽을 생성하기 위해 100개의 발간자 (publisher) 프로세스를 구동한 화면이며, 그림 6은 리눅스 기반의 라즈베리파이에서 토픽을 생성하기 위해 프로세스를 구동한 화면이다. 윈도우와 리눅스용 소스 코드는 동일하며, 토픽 구성과 크기도 동일하다.

IPMS 서버 역할을 하는 랩탑에서는 필드장비 시뮬레이터에서 발간한 토픽을 구독하며, 각각의 프로세스에서 발간한 토픽의 누락이나 순서를 확인하였다.

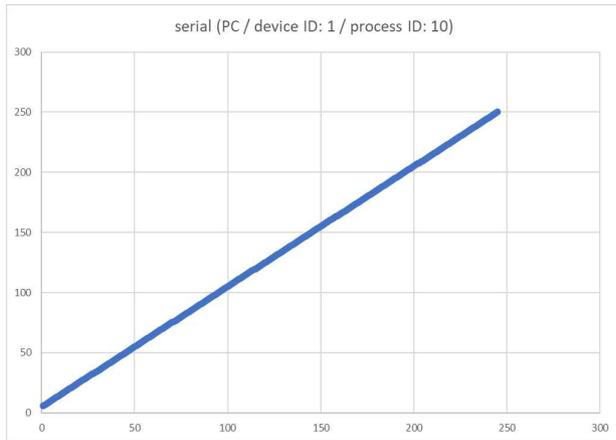


그림 7. 토픽 수신 오류 확인 결과
 Fig. 7. Error verification for topic subscription error.

250개의 토픽 발간 프로세스에서 60,000개의 토픽을 송신하였으며, 분석 결과 윈도우 기반의 랩탑에서만 4개 토픽이 누락되었으나, 순서가 바뀐 토픽은 없었다. 누락된 토픽은 0.01%의 낮은 비율이며 subscriber 생성 초기 4초 이내의 시간 동안에 발생하였고, 이후에는 오류가 없었다. 이는 특정 운영체제상의 RTPS 구조에서 토픽 수신이 안정화되는 과정에서 발생한 현상으로 판단된다. 이러한 현상은 시뮬레이터 구동 초기에 불규칙적으로 발생할 가능성이 있으며, 특정 운영체제에서 많은 토픽을 발간하는 경우에 발생 빈도가 증가할 가능성이 있다. 따라서 실제 DDS를 기반으로 선박의 IPMS를 개발할 때는 추가적인 테스트가 필요하다.

그림 7은 특정 프로세스(필드 장비 시뮬레이터 랩탑의 10번 토픽 생성 프로세스)에서 발간한 토픽을 IPMS 서버인 랩탑에서 수신한 결과를 예를 들어 보여주고 있다. 그림에서 x축은 토픽 발간 시에 부여한 일련번호이며, y축은 토픽의 수신 순서이다. 그림에서 보는 것처럼 누락된 토픽이나 순서가 바뀐 토픽은 해당 프로세스에서는 발견되지 않았다.

이중 운영체제로 동작하는 필드 장비를 DDS로 통합한 선박 IPMS 시뮬레이터를 개발하였고, 실제 선박과 같이 다수의 토픽들이 비동기적으로 발간 및 구독되는 상황에서 토픽의 손실이나 순서의 에러를 확인하였다. 결과적으로 구동 초기에 0.01% 수준의 토픽 손실은 발생하였으나 토픽 순서의 에러는 발생하지 않았다.

IV. 결론 및 향후 연구 방향

선박의 IPMS는 선박에 설치된 대부분의 필드 장비를 종합하여 감시 제어하는 중요한 통제장치이다. 선박에 승조하는 인원이 감소하여 자동화가 증가함에 따라 핵심 통제장치인 IPMS에는 많은 센서와 장비들이 연결되어 각각 비동기적으로 다른 특성의 데이터를 송수신하게 될 것이다. 또한 자율운항선박이나 무인선박에 대한 기술개발이 진행됨에 따라. 기존의 추진기

관에 관한 정보뿐만 아니라 항해센서, 통신체계와 관련된 모든 정보를 통합하는 상황에 놓여있다.

이러한 상황에서 각기 다른 구동 환경에서 개발된 필드 장비들을 통합할 필요가 있으며, 본 연구에서는 DDS를 기반으로 통합하는 IPMS 시뮬레이터를 구현하여 데이터 송수신 오류를 시험하였다.

실제 선박에서는 다수의 필드장비에서 비동기적으로 많은 데이터를 송신하게 되므로 이와 유사한 실험환경을 구성하였다. 서로 다른 운영체제로 구동하는 필드장비 시뮬레이터 4대에서 총 250개의 토픽 프로세스를 각각 구동하여 초당 1,000개의 비동기적인 토픽을 발간하고, 이를 IPMS 콘솔 시뮬레이터에서 구독하여 오류를 확인하였다.

개발된 IPMS 시뮬레이터는 초기의 아주 작은 비율의 토픽 손실만을 보여주었으므로, 향후 실제 선박의 IPMS 개발이나 다른 제어체계의 통합이나 연동에도 적용 가능할 것으로 판단된다.

다만, 본 논문에서는 실제 선박 IPMS의 약 1/3 수준의 데이터와 노드(장비)를 모사하였으며 데이터의 수신 오류를 중심으로 한 제한사항이 있다. 향후에는 자율운항선박의 환경을 고려하여 항해계통과 기관계통의 DDS 도메인을 각각 구성하고 이들 간의 데이터 교환에 대한 실증연구를 진행할 예정이다.

References

- [1] J. W. Park, S. Y. Sung, Y. K. Lim, C. H. Yun, and S. G. Kim, "An implementation of integrated interface system for the digital ship," *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 16, No. 6, pp. 1158-1166, Jun. 2012.
- [2] H. Peplinski, *Ship and Mobile Offshore Unit Automation*, Sebastopol, CA: Gulf Professional Publishing, 2021.
- [3] OMG, *Data distribution service for real-time systems*, Version 1.4, 2015.
- [4] D. C. Schmidt, A. Corsaro, and H. Hag, "Addressing the challenges of tactical information management in net-centric systems with DDS," *The Journal of Defense Software Engineering*, pp. 24-29, Mar. 2008.
- [5] J. Y. Im, "Performance evaluation of discovery and message transmission of DDS (data distribution service) security," *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 25, No. 5, pp. 701-708. May 2021.
- [6] PMS 406, *Unmanned maritime autonomy architecture (UMAA) sensor and effector management (SEM) interface control document (ICD)*, Version 3.0.1, 2021.
- [7] S. A. Kim and S. W. Kim, "Quality analysis for the data distribution service of the real-time integrated railway

safety monitoring and control system,” *Journal of Korean Society for Urban Railway*, Vol. 6, No. 4, pp. 351-361, Dec. 2018.

[8] D. Lee and H. S. Yang, “Feasibility study of DDS based microgrid communication architecture,” *The Transactions of the Korean Institute of Electrical Engineers*, Vol. 71, No. 3, pp. 517-524, Mar. 2022.

[9] J. H. Jang, D. J. Kim, M. G. Kim, and J. S. Oh, “Development of naval ship propulsion system simulator for CODLOG based ECS verification,” *Journal of the Korea Institute of Information and Communication Engineering*, Vol. 21, No. 9, pp. 1976-1807, Sep. 2017.

[10] N. Y. Son, H. S. Lee, and J. S. Oh, “Development of propulsion equipment model and simulator for verification of propulsion system,” *Journal of the Korean Society of Marine Engineering*, Vol. 43, No. 1 pp. 48-55, Jan. 2019.

[11] M. J. Michaud, T. Dean, and S. P. Leblanc, “Attacking OMG data distribution service (DDS) based real-time mission critical distributed systems,” in *13th International Conference on Malicious and Unwanted Software*, Nantucket: MA, Oct. 2018.



오 성 원 (Seongwon Oh)

1996년 3월 : 해군사관학교 전기공학과 (공학사),
 2004년 2월 : 포항공과대학교 대학원 정보통신공학과(공학석사)
 2011년 8월 : 미국 Texas A&M University 전자공학과 (공학박사)
 1996년 3월 ~ 2019년 1월 : 해군(조함단, 전력발전단, 전력분석시험평가단, 국과연)
 2019년 3월 ~ 현재 : 국립목포해양대학교 해군사관학부 조교수
 ※ 관심분야 : 전자파 해석, 레이더, 배열안테나, 제어시스템