Towards a Modelling & Simulation Capability for Training Autonomous Vehicles in Complex Maritime Operations

Arnau Carrera Viñas, Thomas Mansfield, Pilar Caamaño Sobrino and Alberto Tremori
NATO STO Centre for Maritime Research and Experimentation (CMRE),
ITALY

Arnau.Carrera@cmre.nato.int, Thomas.Mansfield@cmre.nato.int, Pilar.Caamano@cmre.nato.int,
Alberto.Tremori@cmre.nato.int

ABSTRACT

Developers of autonomous systems require testing to train and verify their algorithms. End users may also use this data when deciding how to efficiently utilise the systems. Simulation represents an alternative to experimentation in real environments, which is safer, more cost-effective and, allows repeatable and controllable experiments to be executed. Traditionally, experts in robotics use simulators that focus on details related to their system while simplifying aspects related to the environment, communications and inter-asset relationships. As an alternative, CMRE proposes a Maritime Simulation Framework (MSF), interoperable with robotics middleware (i.e. MOOS and ROS), adopting a hardware and software-in-the loop-simulation approach, which permits the simulation of important external factors which are usually simplified. The inclusion of these expanded elements enables the discovery of interactions that may not be known to the autonomous system's developer, improving the robustness of the system under development. The aim of this work is to build a configurable and extendable simulation framework to train and test autonomous behaviours for maritime systems, to assist system developers and to support end users' operational decisions.

The framework is composed of dedicated simulators within the high level architecture (HLA), with federates simulating environments, platform dynamics, sensing emulation, communications and intuitive visualizations. The proposed framework provides a simulation that encompasses the challenges of complex maritime operations, with a focus on the underwater domain, providing a more comprehensive and realistic capability than more traditional approaches. To date the MSF has been used to support the development of autonomous system algorithms in Mine Counter Measure (MCM) and Anti-Submarine Warfare (ASW) missions, with single or multiple vehicle configurations.

1.0 INTRODUCTION

With every day that passes, autonomous systems (AS) are deployed in an increasing number of diverse scenarios where they are required to complete challenging tasks autonomously. This implies that the intelligent algorithms governing the behaviour and decisions of the systems need to be more complex, comprehensive and adaptive to tackle the diversity of the situations faced. In this context, autonomous systems use machine learning methods to partially or fully solve complex tasks. This process usually requires a training or learning process to generate a computational model of the task. This model defines the actions that the system can take and the result of the execution of those actions in the environment where the system is performing the task. With this model, the system is able to predict which of the possible actions could take it one step forward in the consecution of its goal. The accuracy of this prediction depends, mainly, on the quality of the training. Modelling and Simulation (M&S) has proven to be a key element in the creation of high quality and realistic simulation environments to train and test machine learning algorithms for autonomous systems [1] [2].

In simple terms, the goal of all machine learning algorithms is to tune a set of parameters according to a set of inputs to obtain a desired output. Within the branch of machine learning, AS developers can find diverse methodologies and approaches to address this issue. In some methodologies, like supervised and unsupervised machine learning, the algorithm can learn the model from either a dataset obtained from the real world or synthetic data generated by simulation. Meanwhile, machine learning methods based on reinforcement learning require the system to be able to receive feedback from the learning environment in order to adapt its behaviour to the actions that have a positive impact to obtain the desired results. In both cases, M&S can be a supporting tool in the learning process. In the former case, generating extensive synthetic data sets when real data is difficult or costly to obtain limiting the learning capabilities. In the latter case, providing a safe-to-fail and interactive environment where the systems can receive feedback minimizing risks.

Traditionally, machine learning-based frameworks to train and validate development of new algorithms and systems are developed using common benchmarking problems or in a very ad-hoc manner [3] [4]. The scenarios used in benchmarking frameworks are not often designed to train systems to learn applied problems in real world conditions, since the goal is to compare new algorithms with stablished methods. For training algorithms that will be used in real world applications, the developer usually extends an existing simulation tool or creates a new one specifically to solve the problem. These approaches narrow the learning capabilities of the system to a set of foreseen conditions and limits the discovery of possible interactions with elements unknown by the autonomous system's developer. Moreover, it hampers the overcoming of the reality gap, which is one of the most important considerations in the transition of the systems from a testing and training environment to real operations.

In order to deal efficiently with these shortcomings, the authors of this work started in 2015 the development of a multidisciplinary standard-based distributed simulation environment for underwater autonomous systems [5] [6]. This paper presents an update of that initial framework now called Maritime Simulation Framework (MSF).

The content of this paper is structured as follows. Section 2 describes the background of this work. Section 3 explains in detail the MSF capabilities and functionalities. Section 4 presents possible use cases of the MSF for training autonomous underwater vehicles, and Section 5 closes this paper summarizing the conclusions of the work including the benefits of using the MSF.

2.0 BACKGROUND

In 2015, when the development of MSF started, most of the simulation tools or frameworks available for the underwater domain were mainly focused on the simulation of Remotely Operated Vehicles (ROVs) and on providing functionalities for training operators [7] [8]. Those frameworks include accurate physics affected by the sea state and the motion of the ROV's tether, sensor simulations (e.g. sonar and optical cameras), and different set of tools such as underwater manipulators, laser measuring devices, and water samplers [9] [10]. But none of them meet the requirements on scalability and flexibility needed for the CMRE framework.

In the latest years, as the development of underwater unmanned system advanced, the development of simulation frameworks for underwater scenarios has also advanced and new tools are available [11] [12]. Simulation tools such as UWSim [13], UUV [14], StoneFish [15], URSim [16], USVSim [17], Rock-Gazebo [18] or *freefloating-gazebo* [19] provide accurate simulation of dynamics and hydrodynamics, and realistic visual simulation. However, these tools lack of some functionalities like compliance with High Level Architecture (HLA) [20], the NATO standard for distributed M&S (STANAG 4603 [20]; ability to simulate underwater communications and the effect of environmental conditions on these communications; or interoperability with Command and Control (C2) systems. These features are required in order to simulate NATO Maritime Operational Missions of high importance for CMRE researchers and the military operational community that CMRE is supporting.

Developed following the IEEE recommended practices for distributed simulation and interoperability [21], MSF allows for the seamless integration of autonomous systems in the simulated world maximising interoperability, the usage of distributed resources, and reusability to facilitate the development of complex simulated environments. While, the majority of the works presented in the robotics' community are based on the injection of simulated events directly into the operating system of the robot [22] or the integration of the machine learning algorithms in a learning framework or simulation. The presented proposal is derived from M&S community ideas, like virtual reality and immersive environments, where systems are "immersed" in simulated environments.

3.0 MARITIME SIMULATION FRAMEWOK (MSF)

The MSF capability is achieved bringing together the different communities using M&S standards as a baseline for the integration and connectivity with the standards of other communities [5]. The main motivation behind the development of a standards-based immersive environment is to eliminate the effort required to include existing ASs into complex and interoperable simulated scenarios. In this sense, this proposal preserves the existent control architecture in the system, by adding a layer or middleware in the simulation environment to achieve the interoperability of the desired system at software or even at hardware level.

MSF is interoperable with the most common robotics middleware's such as the Robotic Operating System (ROS) [23] and the Mission-Orientated Operating Suite (MOOS) [24]. MSF is a distributed, modular and scalable simulation framework. By adopting an approach of Hardware and Software-in-the-loop, MSF allows to immerse robotic systems into realistic scenarios and the exchange of information with specialized models or modules developed by Subject Matter Experts (SMEs). With this variety of simulation tools, MSF permits the simulation of complex maritime scenarios that incorporate the underwater, water surface, and air domains. MSF can be configured to different levels of detail and includes capabilities required for extended maritime simulations, including the execution of cooperative and collaborative scenarios.

MSF offers mission capabilities and functionalities that are not frequently found in simulation frameworks commonly used for training autonomous systems, like environmental, communications and multi-asset simulations. With the availability of these tools, the autonomous systems can be trained in more complex situations allowing testing and validation in more challenging conditions, closer to those that the systems will face in reality.

These developments, and the ones on going, considering a possible integration in a future with machine learning frameworks like TensorFlow [25], PyTorch [26], Sci-Kit Learn [27], or H2O [28], for supervised and unsupervised learining; or OpenAI Gym [29], RLLib [30], Tensorforce [31], or Horizon [32] , for reinforcement learning and its combination with deep learning.

3.1 MSF High Level Architecture Design

Figure 1 shows a high-level view of the different simulators and tools available in the MSF. Each of these components communicates using the Run Time Infrastructure (RTI) middleware required to communicate using HLA standard. The simulation framework has been designed following a loosely coupled architecture so the simulation can run with only the desired components and, thanks to the HLA principles, it can be executed in one computer or distributed among multiple computers across one or more networks. Moreover, the simulation has a time-management system, which allows the user to run the simulation in fast-time, a feature which is of paramount importance in time-intensive machine learning training applications.

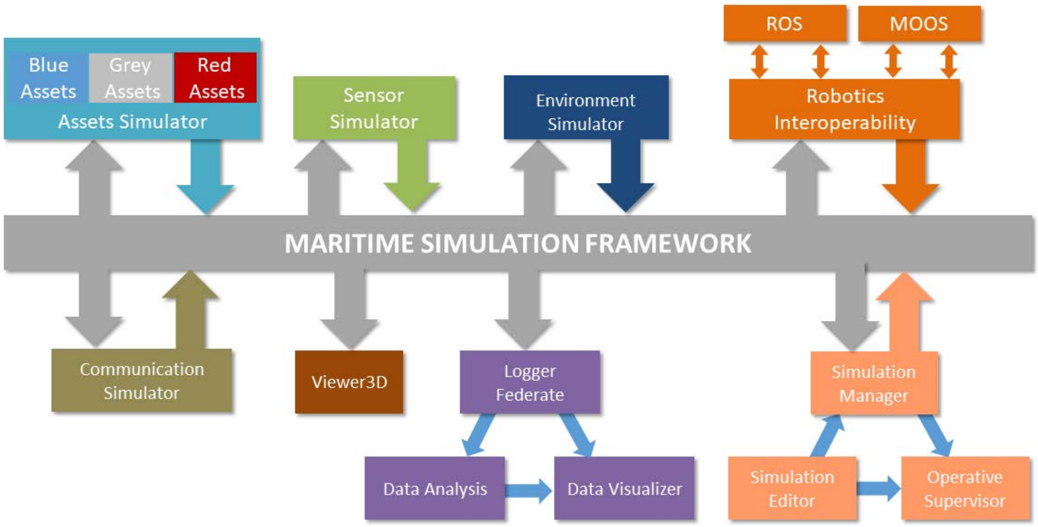

Figure 1 Design of the Maritime Simulation Framework (MSF)

The next subsections briefly describe the modules or federates that compose the MSF and their capabilities.

3.1.1 Asset Simulator

The asset simulator is responsible for simulating all the entities that move in the different domains (underwater, sea surface, and air) inside the simulation environment. It takes into consideration the dynamics and features of the assets, their guidance type, environmental effects and the uncertainty in estimating the position.

The assets simulator defines three conceptual categories (hovering, torpedo, and glider) that cover almost all the asset types found in a maritime scenario. These classification categories have been identified according to the asset's controllable Degrees of Freedom (DoF). The features of the assets are parametrized and can be added or modified editing the configuration files of the simulator.

The guidance of the assets can be done with three types of commands; steering commands (target forward speed, target heading, target altitude/depth), waypoint missions, and track line missions. These commands can be implemented in two main ways; the mission can be defined as a part of a scenario, allowing the federate to take full control of the asset. Alternatively, the commands may be the provided by a connected robotic system.

3.1.2 Sensor Simulator

The Sensor Simulator aims at simulating the effect of a range of sensors in the simulation scenario. The modelling of a specific sensor's ability to detect and classify a particular target is based on the generation of a statistical model. The statistical model considers not only the sensor and target types, but also their relative orientation and range, shape, material, and local environmental considerations. In addition to the availability of models that represent the detection and classification, simplified models are also incorporated into the Sensor Simulator that generate clutter and false detections. Key benefits of this approach focus on the ability to abstract many of the complex and deeply analytical sensor interactions while maintaining the ability to provide realistic and representative inputs into the higher-level functions.

The sensors simulator classifies the sensor in four different conceptual model type based on how is defined their field of view. The defined types are Single directional field of view (i.e. Optical Camera), Twin swath field of view (i.e. side scan sonar), Omni-directional field of view (i.e. Passive sonar) and Multi static sensors (i.e. Multi static sonar).

Finally, the sensor simulator can be enrich with external tools (e.g. RAPS [33] or previously MSTPA [34]) to generate more detailed information for representing the environment, for example the inclusion of detailed acoustic underwater maps to represent the probability of detection using models like ARTEMIS [35] [36] [37] [38] or Bellhop [39].

3.1.3 Environmental Simulator

The goal of this federate is to generate a meteorological and oceanographic environment to influence a wide range of assets and simulation components. The generated environmental model is divided into four gridded zones, the Air Column, the Water Surface, the Water Column and the Seabed. Each zone is broken into a series of 'data cubes' that can be referenced by row, column and, in the case of the air and water zones, layer values.

The simulation environment can be loaded with meteorological and oceanographic data from various sources, with data accepted in a range of standardised formats that are well-known to the environmental modelling community. These data may be obtained from public sources such as Copernicus [40], Météo-France [41]and European Marine Observation and Data Network [42]. Several of these sources contain data that represent the evolution of the data over time, the environmental simulator is capable to update the environmental conditions according to the evolution of the data and the simulation.

This approach ensures that all of the published conditions are consistent with each other (i.e., it is unlikely to be snowing if the air temperature is 30℃). However, there are uses of M&S where this consistency is not required, and the simulation objectives require control over each parameter individually. To enable this, the user is able to select and tailor individual environmental parameters before the simulation using a Graphical User Interface (GUI).

3.1.4 Robotics Interoperability

The AS software architecture is usually organized following a modular and layered approach. The modules and layers are independent to facilitate the reusability and increase the robustness; however, this has the drawback that it requires a communication system between processes. Hence, one of the main roles of robotics middleware is to offer robust and reliable communication between modules. For this reason, the MSF disposes of wrappers or bridges to convert HLA data into MOOS and ROS middleware. In this way some layers or modules can be replaced by data generate in the MSF without modifying the software of the autonomous systems.

3.1.5 Communication Simulator

The Communication Simulator is in charge of simulating the transmission of messages between assets in the simulation. The federate needs to analyse the environment, the type of communication, and the geometry of the communication network to estimate the transmission probability and the delay to simulate the transmission and reception of messages. The estimation of the acoustic propagation is based on the BELLHOP [39] acoustic model. This functionality is of paramount importance for the simulation of missions with cooperative and collaborative behaviours.

The Communication Simulator contains software-in-the-loop, in order to simulate the underwater network strategies that needs to be consider to perform realistic transmission of messages. Currently it incorporates Cognitive Communications Architecture (CCA) [43] and is compatible with DCCL message encoding [44].

3.1.6 Viewer 3D Federate

The Viewer Federate displays an intuitive 3D representation in a virtual environment of the entire simulated scenario. The realistic representation of the scenario has several benefits; in particular, it supports face validation of the simulation and facilitates the explanation of the autonomous behaviours to users. The visualisation can be presented using conventional screens or using Virtual Reality (VR) devices. Figure 2 shows two screen captures of the Viewer 3D.

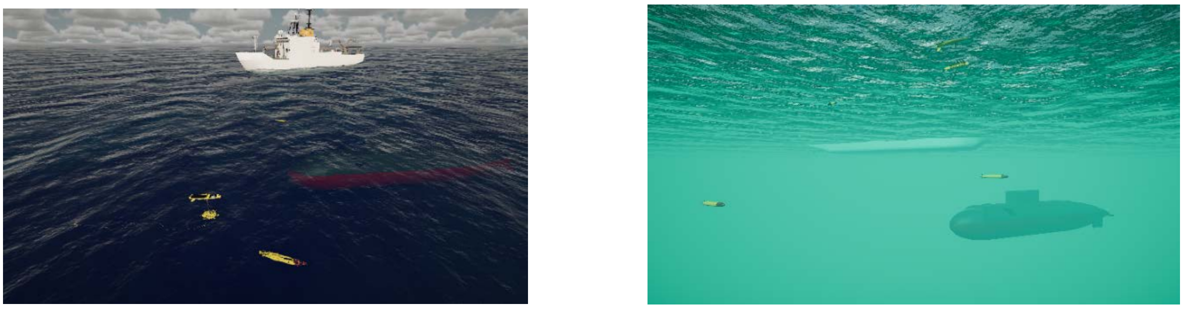

Figure 2 Two images of the viewer 3D showing the typical assets used in an ASW scenario with the assets of the NATO STO CMRE. At the left an image from the surface and on the right an image from the underwater perspective.

3.1.7 Logger Federate, Data Analysis and Data Visualization

The logger federate aims to collect all the relevant information that is generated during the simulation. The information collected ranges from telemetry of the assets to the detections generated from the simulated sensors, it includes the information estimated during the simulations and the ground truth to perform the analysis of the results. The goal is to record similar information to that scientists or engineers receive from typical experimentation enriched with ground truth information available only in a simulated environment.

The Data Analysis and Visualizer are complementary tools developed by the authors to support the interpretation and analysis of the simulation results. The Data Analysis processes the raw data generated by the logger during the simulation or simulations and computes Key Performance Indicators (KPIs) that allow the user to measure an algorithm's performance. The Data Visualizer reads part of the raw data and part of the KPIs to display the results in a friendly and visual way.

These tools can support the training of the machine learning algorithm in multiple forms. For example spotting areas that are not improving over the training of the behaviour caused by an incomplete definition of the learning problem. Furthermore, comparing the results of different learning strategies or traditional methods to identify which is the best option.

3.1.8 Simulation Manager, Simulation Editor and Operational Supervisor

The simulation manager, simulation editor and operational supervisor are software tools that improve the user experience in the definition of the simulated scenario, the execution of one or multiple simulations, and the supervision of the execution. The three tools offer a GUI to coordinate the

The definition of scenarios includes selecting the geographical location, vehicles with their payloads and missions, and the meteorological conditions. This helps to prepare the scenario for the simulation. Which can be also used to prepare the training scenario of the AS, and it is performed in the simulation editor (see Figure 3).

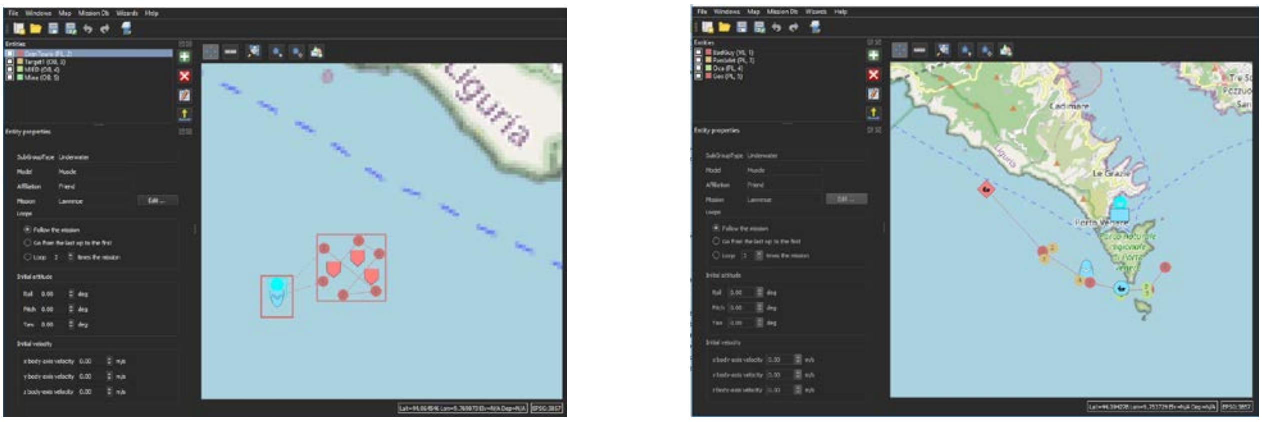

Figure 3 Two simulation scenario configured in the Simulation Editor tool. On the left a Trackline mission to be followed by an AUV with 3 objects located on the seabed, on the right a boat that cross an area where an Unmanned Aerial Vehicle (UAV) and a Unmanned Surface Vehicle (USV) are patrolling.

The simulation manager facilitates the execution of the simulation simplifying the execution of a distributed simulation, the proper configuration of the simulation, and concentrating all the usual spares information in one single interface (see Figure 4).

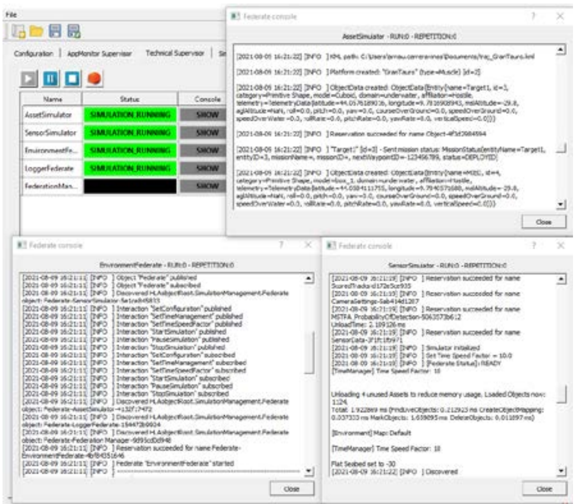

Figure 4 Screen Capture of the Simulation Manager while running a simulation. In the image can be seen four active federates (Assets Simulator, Sensor simulator, Environmental Federate and Logger Federate), and the command line outputs of three of them.

The operational supervisor, is two dimensional (2D) representation of the scenario where is displayed the current locations, trajectory performed and planned missions during the execution of the simulation (see Figure 5). This offers a face validation capability less computational heavy than the viewer 3D. Moreover, 2D representation is a more comprehensive representation when the assets are sparse in large areas.

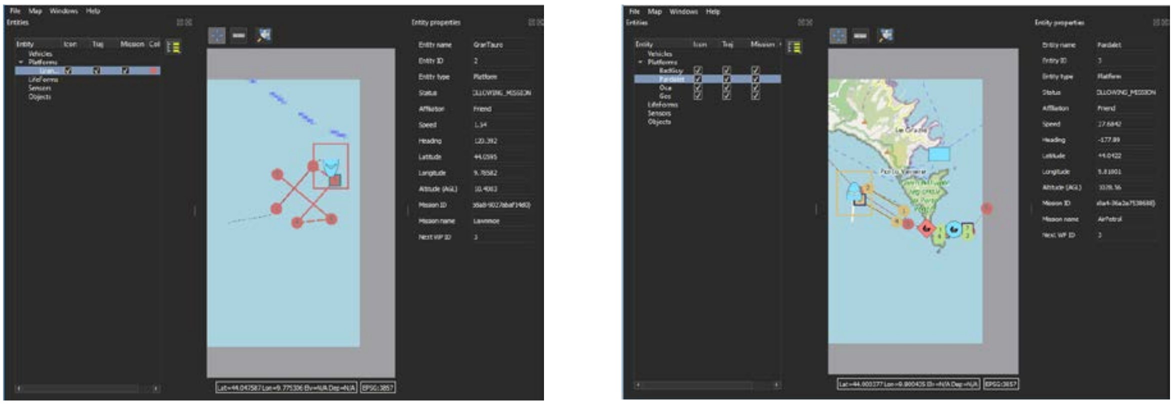

Figure 5 Two images of the Operational supervisor showing the entities status, active missions and performed trajectories, in the two scenarios designed in Figure 3.

4.0 USE CASES AND APPLICATION

The MSF has been designed and built in the context of the NATO STO CMRE, aiming to create a simulation framework to perform Verification and Validation (V&V) and Concept Development and Experimentation (CD&E) of innovative behaviours for maritime AS. However, with minor modification the same environment can be used by SME's to perform AS algorithm training, testing and mission-level concept development in maritime environments.

As an initial proof of concept, the MSF has been applied as the keystone of a distributed simulation capability supported by CMRE [45]. Within this example, the MSF environmental, communications, asset and sensor federates were run to generate a range of datasets that represented the evolution of a multi-domain, multi-asset maritime mission with varying meteorological and oceanographic environmental conditions. These datasets were used to train, then test, multiple data fusion machine learning algorithms developed by partner organisations.

This proof of concept event demonstrated the potential value of the MSF in two main ways; firstly, the simulation environment successfully provided machine learning developers with a persistent, on-line algorithm, training and testing environment in which they could gain access to representative datasets generated by the each of the connected federates. This allowed the algorithms SME's to train, test and improve their algorithms in advance of a series of real-life sea trials improvements to be identified and implemented. Secondly, the availability of the MSF visualisation and data analysis tools allowed the separate algorithm developer and end-user SME communities to jointly communicate and understand key algorithm requirements, capabilities and limitations. Again, conducted in advance of a series of live sea trials, this exchange of information took place in a controlled, repeatable and safe-to-fail environment allowing improvements to both technical capabilities and their mission-level application.

Building upon this initial proof of concept, the authors of this paper propose two further scenarios that could be simulated to better investigate the training and testing of AS behaviours within the MSF.

4.1 Learning how to protect a choke point.

The MSF is capable to offer a simulated scenario where an enemy entity wants to enter a choke point area while the blue assets protects the area and track the incoming entity. The MSF dispose of multiple sensors (Active and Passive sonars, Radar systems, Cameras) that simulation the detection of diverse types of trespassers.

From the point of view of the authors different aspects of this scenario could be learned, one for example could be the geographical disposition of the assets to maximize the full coverage of the area, second aspect that could be learn is the tracking behaviour once the suspicious asset has been detected in order to maximize the tracking time.

4.2 Learning how to Survey a Q-Route.

In this scenario the aim of the simulation is to present a Q-Route that needs to be surveyed in order to guarantee the safety passage of vessels in that area. The simulation is capable to offer a scenario with different type of seabed, where a set of objects can be placed by the user or randomly on the sea bottom. The simulation already offers a set of sensors that can be configure in order to detect and classify the objects and also to detect and classify the characteristics of the seabed. The user of the simulation could train a system to learn how to survey the Q-Route area while dynamically adapt the trajectory according to the obtained environmental data.

5.0 CONCLUSIONS.

The Maritime Simulation Framework (MSF) developed by the NATO STO CMRE is a modelling and simulation framework based on HLA that covers the simulation of maritime operations with a special focus on the underwater domain and the autonomous systems (AS). It includes an extensive set of specialised simulation models developed in conjunction with subject matter experts (SMEs) offering complex simulation capabilities covering the main aspects of a maritime scenario.

MSF provides a more comprehensive environment than the traditional robotics simulation or machine learning frameworks, which have a reduced simulation scope. This wider simulation capability allows the discovery of not foreseen relationships or factors to consider during the learning phases of behaviours for an AS.

The authors have presented the potential of the MSF as a simulation environment for machine learning algorithms with a proof of concept in a distribute simulation between several partners. In particular, one of the partners in the consortium used the distributed simulation, where MSF was part, to train and test their machine learning algorithms.

Having successfully demonstrated its ability to support autonomous system development via both technical training and testing along with improved support to multiple SME communities, MSF is an evolving simulation environment that is continuously improving.

This work is now progressing onto future opportunities such as the inclusion of additional sensors types, the integration of specialized models to improve communications, the inclusion of more environmental conditions, and the interoperability with C2 systems by standards interfaces.

REFERENCES

[1] A.C. Madrigal, “Inside Waymo's Secret World for Training Self-Driving Cars,” The Atlantic, August 2017. [Online]. Available: https://www.theatlantic.com/technology/archive/2017/08/inside-waymos-secret-testing-and-simulation-facilities/537648/. [Accessed 25 August 2021].

[2] A. J. Hawkins, “Welcome to Simulation City, the virtual world where Waymo tests its autonomous vehicles,” The Verge, 6 July 2021. [Online]. Available: https://www.theverge.com/2021/7/6/22565448/waymo-simulation-city-autonomous-vehicle-testing-virtual. [Accessed 31 August 2021].

[3] J. Leudet, F. Christophe, T. Mikkonen and T. Mannisto, "AILiveSim: An Extensible Virtual Environment for Training Autonomous Vehicles," in *Computer Software and Applications Conference*, Milwaukee, USA, 2019.

[4] S.K. Choi, S.A. Menor and J. Yuh, "Distributed virtual environment collaborative simulator for underwater robots," in *Intelligent Rodots and Systems*, Takamatsu, Japan., 2000.

[5] A. Carrera, A. Tremori, P. Caamaño, R. Been, D. Crespo Pereira and A. Bruzzone, "HLA interoperability for ROS-based autonomous systems," in *International Workshop on Modelling and Simulation for Autonomous Systems*, 2016.

[6] A. Tremori, P. Caamaño Sobrino, A. Carrera Viñas, G. L. Maglione, D. Solarna, R. Been and T. Mansfield, "A Verification, Validation and Accreditation Process for Autonomous Interoperable Systems," in *Modelling and Simulation for Autonomous Systems*, 2018.

[7] Marine Simulation, "ROVsim Gen3," Marine Simulation, [Online]. Available: http://marinesimulation.com/rovsim-gen3/. [Accessed 10 08 2021].

[8] PaleBlue, "VR ROV Simulator: Remotely Operated Vehicles," PaleBlue, [Online]. Available: https://pale.blue/simulators/subsea-and-diving/vr-rov-simulator/. [Accessed 10 08 2021].

[9] Fletcher and Barbara, "ROV simulation validation and verification," in *OCEANS*, Halifax, NS, Canada, 1997.

[10] B. Buckham, F. Driscoll, N. Meyer and B. Radanovic, "Three Dimensional Dynamics Simulation of Slack Tether Motion in an ROV System," in *International Offshore and Polar Engineering Conference*, Honolulu, Hawaii, USA, 2003.

[11] J. Collins, S. Chand, A. Vanderkop and D. Howard, "A Review of Physics Simulators for Robotic Applications," *IEEE Access*, vol. 9, pp. 51416-51431, 2021.

[12] D. Cook, A. Vardy and R. Lewis, "A survey of AUV and robot simulators for multi-vehicle operations," in *2014 IEEE/OES Autonomous Underwater Vehicles (AUV)*, 2014, pp. 1-8.

[13] M. Prats, J. Perez, F. J.J. and P. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[14] M. M. Marcusso Manhães, S. A. Scherer, M. Voss, L. R. Douat and T. Rauschenbach, "UUV Simulator: A Gazebo-based package for underwater intervention and multi-robot simulation," in *OCEANS 2016 MTS/IEEE Monterey*, 2016, pp. 1-8.

[15] P. Cieślak, "Stonefish: An Advanced Open-Source Simulation Tool Designed for Marine Robotics, With a ROS Interface," in *OCEANS 2019 - Marseille*, 2019.

[16] P. Katara, M. Khanna, H. Nagar and A. Panaiyappan, "Open Source Simulator for Unmanned Underwater Vehicles using ROS and Unity3D," in *2019 IEEE Underwater Technology (UT)*, 2019.

[17] M. Paravisi, D. H. Santos, V. Jorge, G. Heck, L. M. Gonçakves and A. Amory, "Unmanned Surface Vehicle Simulator with Realistic Environmental Disturbances," *Sensors*, vol. 19, no. 5, pp. 1424-8220, 2019.

[18] T. Watanabe, G. Neves, R. Cerqueira, T. Trocoli, M. Reis, S. Joyeux and J. Albiez, "The Rock-Gazebo Integration and a Real-Time AUV Simulation," in *2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR)*, 2015.

[19] O. Kermorgant, "A dynamic simulator for underwater vehicle-manipulators," in *International Conference on Simulation, Modeling, and Programming for Autonomous Robots*, 2014.

[20] IEEE, "1516-2010 Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) -- Framework and Rules," IEEE, 2010.

[21] IEEE, "IEEE Recommended Practice for Distributed Simulation Engineering and Execution Process (DSEEP)," 2011.

[22] J. L. D. Dantas and E. A. d. Barros, “A Real-Time simulator for AUV development,” *ABCM Symposium Series in Mechatronics*, vol. 4, pp. 499-508, 2010.

[23] Open Robotics, “About ROS,” [Online]. Available: https://www.ros.org/about-ros/. [Accessed 08 10 2021].

[24] Oxford Mobile Robotics Group, “MOOS Introduction,” [Online]. Available: http://www.robots.ox.ac.uk/~mobile/MOOS/wiki/pmwiki.php/Main/Introduction. [Accessed 10 08 2021].

[25] “TensorFlow,” [Online]. Available: www.tensorflow.com. [Accessed 20 08 2021].

[26] Pytorch. [Online]. Available: https://pytorch.org. [Accessed 10 08 2021].

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and É. Duchesnay, “Scikit-learn: Machine Learning in {P}ython,” *Journal of Machine Learning Research*, vol. 12, pp. 2825-2830, 2011.

[28] “H2O,” [Online]. Available: www.h2o.ai. [Accessed 20 08 2021].

[29] O. AI. [Online]. Available: http://gym.openai.com. [Accessed 08 10 2021].

[30] RAY, “RLlib: Scalable Reinforcement Learning,” [Online]. Available: https://docs.ray.io/en/master/rllib.html#rllib-index. [Accessed 10 08 2021].

[31] A. a. S. M. a. F. K. Kuhnle, “Tensorforce: a TensorFlow library for applied reinforcement learning,” [Online]. Available: https://github.com/tensorforce/tensorforce. [Accessed 10 08 2021].

[32] J. Gauci, E. Conti, Y. Liang, K. Virochsiri, Z. Chen, Y. He, Z. Kaden, V. Nayayanan and X. Ye, “Horizon: Facebook's Open Source Applied Reinforcement Learning Platform,” arXiv prepint:1811.00260, 2018.

[33] C. Strode and M. Oddone, “ASW Optimization using the Rapid Acoustic Prediction Service (RAPS),” CMRE-MR-2018-010, NATO UNCLASSIFIED, La Spezia, 2018.

[34] C. Strode and M. Oddone, “The Multistatic Tactical Planning Aid (MSTPA) - User Guide V4.1.2R,” *CMRE-SP-2013-001*, p. 1, 2013.

[35] Harrison, “User guide for Artemis with convergence and range dependence,” *CMRE SP-2014-001*, 2014.

[36] C. Harrison, “Propagation and bistatic reverberation with convergence in a range-dependent environment: implementation in Artemis,” *CMRE-FR-2014-014*, p. 14, 2014.

[37] C. Harrison, “Ray convergence in a flux-like propagation formulation,” *Journal of the Acoustical Society of America* , vol. 133, no. 6, pp. 3777-3789, 2013.

[38] C. Harrison, “Closed form bistatic reverberation and target echoes with variable bathymetry and sound speed,” *IEEE Journal of Oceanic Engineering*, vol. 30, no. 4, pp. 660-675, 2005.

[39] M. B. Porter, “The BELLHOP Manual and Users Guide,” Heat, Light and Sound Research Inc, La Jolla, CA, USA., 2011.

[40] Copernicus, “Copernicus: Europe's eyes on Earth,” 08 August 2021. [Online]. Available: https://www.copernicus.eu/it.

[41] Meteo-France, “Meteo-France,” 22 May 2020. [Online]. Available: http://www.meteofrance.com/accueil.

[42] EMODet, “Central Portal: Your gateway to marine data in Europe,” 22 May 2020. [Online]. Available: https://www.emodnet.eu/.

[43] R. Petroccia, G. Zappa. T.C. Furfaro, J. Alves and L. D'Amaro, “Development of a software-defined and cognitive communications architecture at CMRE,” in *OCEANS 2018*, Charleston, SC, 2018.

[44] S. Petillo, T. Schneider, H. Schmidt and C. Murphy, “The Dynamic Compact Control Language version 3,” in *OCEANS 2015*, Genova, 2015.

[45] OCEAN2020, "OCEAN2020 Webinar: "Focus on Modelling & Simulation" - full record of the event," OCEAN2020, 14 April 2021. [Online]. Available: https://ocean2020.eu/2021/04/14/cean2020-webinar-focus-on-modelling-simulation-full-record-of-the-event/. [Accessed 10 August 2021].