Planning and Supervising Autonomous Underwater Vehicles through the Mission Management Tool

Afshin Ameri E.
School of Innovation, Design, Engineering
Mälardalen University
Västerås, Sweden
afshin.ameri@mdh.se

Baran Cürüklü
School of Innovation, Design, Engineering
Mälardalen University
Västerås, Sweden
baran.curuklu@mdh.se

Branko Miloradović
School of Innovation, Design, Engineering
Mälardalen University
Västerås, Sweden
branko.miloradovic@mdh.se

Mikael Ektröm
School of Innovation, Design, Engineering
Mälardalen University
Västerås, Sweden
mikael.ekstrom@mdh.se

***Abstract*—Complex underwater missions involving heterogeneous groups of AUVs and other types of vehicles require a number of steps from defining and planning the mission, orchestration during the mission execution, recovery of the vehicles, and finally post-mission data analysis. In this work the Mission Management Tool (MMT), a software solution for addressing the above-mentioned services is proposed. As demonstrated in the real-world tests the MMT is able to support the mission operators. The MMT hides the complex system consisting of software solutions, hardware, and vehicles from the user, and allows intuitive interaction with the vehicles involved in a mission. The tool can adapt to a wide spectrum of missions assuming different types of robotic systems and mission objectives.**

Keywords— Human-robot interaction, multi-robot systems, underwater robotics, mission planning

I. Introduction

The seas and the oceans cover 70% of the planet. These habitats are host to large industries and economical activities such as gas and oil, transportation, shipping, marine food production, tourism, and other service-oriented businesses and secondary sectors [1][2]. In many of these industries, operations that are carried out at the sea may potentially affect the marine habitats and the coastlines. In some cases, the consequences of these operations can be critical, e.g. when the outcome is an oil spill, a leak of wastewater, or a problem associated with an underwater construction, which needs detailed inspection or even replacement of a construction part. In addition to that, constant human presence, which is evident in many coat lines, can also affect the underwater habitats.

Thus, both during such critical events and as part of routine activities, seamless collaboration between a team that is distributed between two separate constellations, i.e. the first on land and the second in water may be needed. In this configuration, the former may consist of experts that can give advice to the team that performs the operations in the water, and also be responsible for the orchestration of the mission.

The assumption in this paper is that the latter consists of Underwater Vehicles (AUVs) and/or Remotely Operated Underwater Vehicles (ROVs). Assuming this configuration and mission orchestration from a location on land, e.g. a mission control station, through a Graphical User Interface (GUI) solution, one can define the actions of the orchestration team as (i) monitoring a mission; (ii) inspection of a specific infrastructure; (iii) supervision of a mission with the objective of affecting the mission's progress if required; and finally; (iv) complex collaboration with the ambition of problem solving with the support of the AUVs/ROVs, which are performing the underwater mission. All these different types of actions require that the operators, which are at the mission control station located above the sea, have a sufficient degree of situational awareness regarding the activities that are taking part underwater. Risks analysis for complex missions involving AUVs have been addressed previously and shows that such missions can be assessed [4].

Given this context, it is plausible to assume that underwater operations can greatly benefit from the possibilities provided by robotic solutions, similar to the case of industrial robot solutions in other sectors [5]. However, the underwater environment is challenging especially with respect to mission planning [18][19], and communication [20].

However, solutions in Human-Robot Interaction (HRI) can improve the overall performance of a team and also allow carrying out advanced missions. HRI solutions may also reduce the need for human presence, i.e. divers, at the mission site, which results in less risk-taking, and improved working conditions.

Another factor in favor of robotic solutions is that the access to skilled divers is not always possible. Missions that are located in remote places, or have specific requirements, assume specialized competences and rigorous planning. Thus, from the perspective of work planning, or being able to act on emergency situations, a solution that assumes robotic underwater systems in pair with appropriate HRI solutions may be essential in many operations and business.

Autonomous robotic systems in underwater operations has been proposed earlier, however, the main body of work is within

978-1-7281-5446-6/20/$31.00 ©2020 IEEE

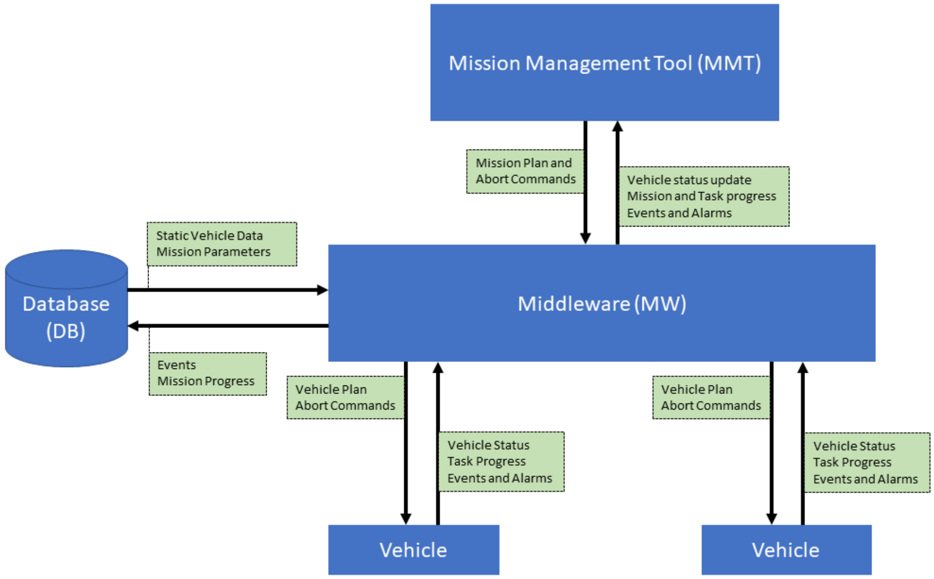

Figure 1. Overview of the system architecture showing the main components required for interacting with the underwater and surface vehicles.

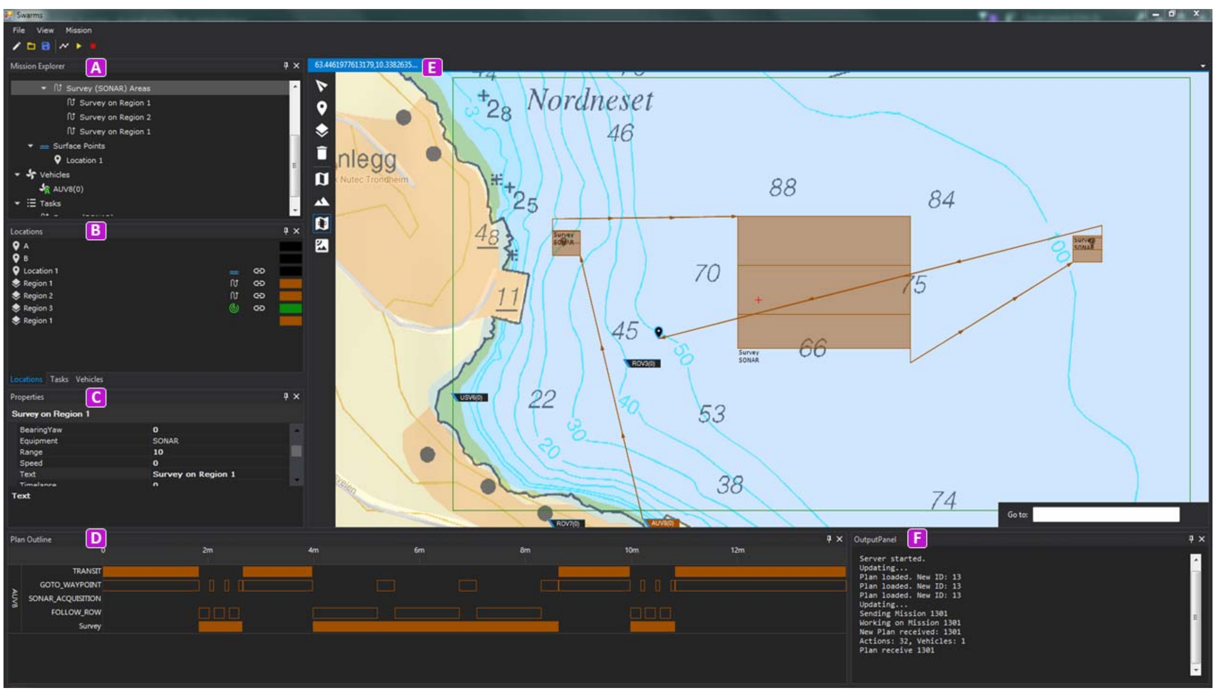

Figure 2. The Mission management tool overview showing the chosen mission plan. The sub windows are (A) Mission Explorer: The central place for defining the relationship between different mission components (locations, vehicles, tasks), (B) Asset Panel: Is a combination of three panels groups together. They are locations, vehicles and tasks. This is where the operator can see and interact will all mission related assets (locations/vehicles/tasks), (C) Properties Panel: Lists all the properties for the selected item, (D) Plan Outline: Shows the plan as a Gantt chart. The Y axis groups the tasks for each vehicle. The X axis is time. (E) Map, (F) Console: Text messages from the system are displayed here.

the systems and hardware design [6][7][8], which is not surprising since the underwater habitat is highly challenging for robotic systems, whereas regarding autonomous operations, the focus is manly on offline path planning [3]. However, aspects regarding the interaction between the human operator and the underwater vehicles have received limited attention, especially within the context of the categories of services that are provided by the MMT.

Low volume in combination with complex technological solutions result in high production costs. The trade-off between affordable solutions on one hand and robustness, reliability, and overall system capability on the other becomes a challenge. In underwater operations, whether divers are involved or not,

Figure 3. In total four vehicles were used in the missions: (Left) Telemetron USV, (Right) AUVs Noptilus-1, Noptilus-2, and Fridtjof.

robust, reliable, and domain-specific solutions are essential. Taken together, it is plausible to assume that one should adopt the HRI solutions when the system is designed.

Activities that assume the deployment of AUVs rely on planning, which is done pre-mission. Such plans consist of tasks that represent events in a mission, which may be defined as monitoring, inspection, supervision, and collaboration. Thus, planning is both a description of a mission and a language understood by the involved AUVs and the operators. Independent of the capabilities of the involved AUVs, their level of autonomy, a mission plan is required.

In this paper a software solution in line with the above-mentioned HRI needs is proposed. This solution is referred to as the Mission Management Tool (MMT) and supports the operators during the stages of mission definition, continuous monitoring, inspection, and supervision, as well as collaboration in the context of semi-autonomous missions involving teams containing AUVs, ROVs, divers, and operators. The MMT has been validated as part of the SWARMs European project [17].

The rest of the paper is organized as follows; in Sect. 2 the MMT design, its implementation, and the overall system are described, Sect. 3 explains the experimental setting and the results. Sect. 4 summarizes the conclusions.

II. Proposed Solution

The MMT is a human multi-robot interaction software aiming at extending the abilities of current robotic solutions for underwater missions. The MMT is developed for the Windows operating system using C# language and freeware libraries. The maps are handled by GMap.NET control library, which can fetch maps from different online sources, such as OpenStreetMap, Google, Bing, and equivalent.

In this paper the MMT is configured for underwater missions, however, it is a tool that can be used in other application domains as well. Its main objective is to provide tools for defining and carrying out a mission involving autonomous systems, in combination with other cyber-physical systems. The MMT can also be used for solely monitoring purposes, e.g. through showing the values of a sensor, which is connected to the MMT.

In the proposed architecture the MMT is connected to a Middleware (MW) solution [12][13], which is the link between the former and the robotic systems. The MW aggregates all the information gathered from different robotic systems and other data sources (environment data, weather data, etc.). This information is communicated to the MMT. The two main components of the MW interact directly with the MMT:

- Mission & Tasks Register & Reporter (MTRR): This is the main component which is responsible for communicating with the vehicles and reporting the information related to the vehicles, mission, and the environment to MMT [12][13].
- Semantic Query (SQ): Stores the information gathered from different sources (including MTRR) related to the mission and provides the information to MMT when required [15][16][11].

Based on this architecture the MW's interaction with the MMT can be broken down into the following:

- Providing MMT with a list of all available vehicles and their capabilities (types of onboard sensors and actuators, and type of tasks that they can perform). Such information is required at the planning stage since vehicles are assigned to different tasks.
- Receiving status updates, events and alarms from the vehicles. Storing the information in a database to keep track of the vehicles and mission situation. The MW also forwards this information to the MMT, so that the vehicles' and mission status can be visualized through the MMT.
- Receiving the full mission from MMT, decomposing it to vehicle-specific list of tasks and forwarding these tasks to the vehicles.

Planning a mission requires access to a different set of resources. This includes maps, vehicles, and their capabilities. In addition, a set of domain-specific tasks must be defined. Obviously, the vehicles must have sensors, actuators, and other SW/HW (all related to vehicle's capabilities) so that at least one vehicle is able to perform a specific task, which is part of the mission. The operator can choose a set of desired tasks that are required for the success of the mission. The MMT will communicate with mission planners [9][10] to generate a list of possible plans, all having the included tasks as building blocks. In the next stage, the operator has the option to choose one of the plans for execution or save it for future use. When the plan is selected for execution, it is forwarded to the MTRR

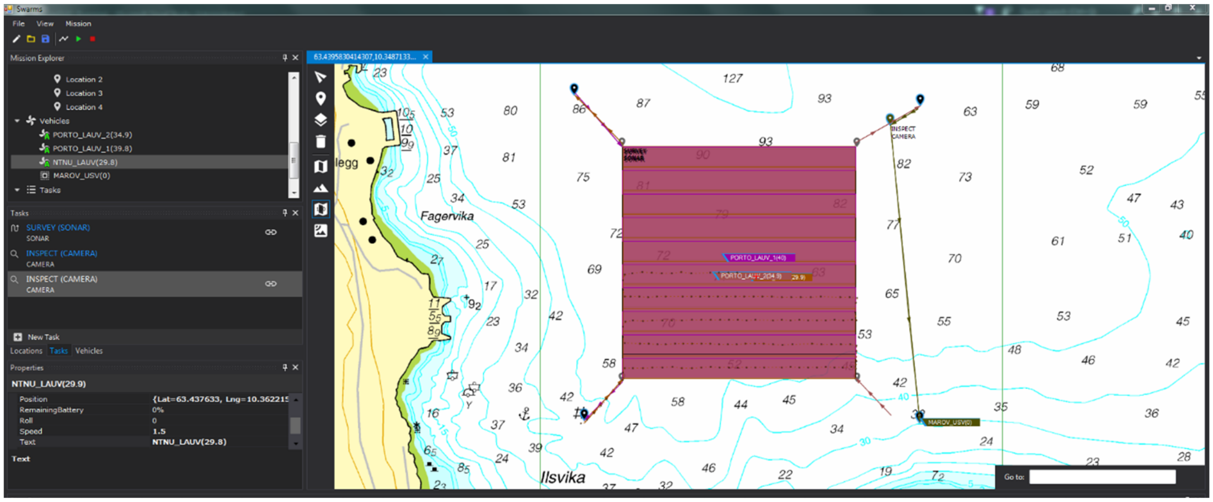

Figure 4. Mission execution shown in the MMT. There are 3 AUVs scanning the same area at different depths. The dotted lines show the path already taken by the AUVs. On the top right there is a USV which provides underwater modem communication and also acts as the recovery vehicle.

component of the MW for decomposition and forwarding to the vehicles.

The planning process is divided into two levels: High-Level Planning (HLP), and Low-Level Planning (LLP). Whereas HLP is not vehicle specific, LLP is vehicle specific, thus it is implemented as part of vehicle control unit. Note further that this multi-level approach is hierarchical and can easily be adapted to different requirements, as well as vehicle compositions. While the HLP defines the plan as described by the operator as well as path planning, LLP is responsible for converting the set of tasks provided by HLP into vehicle-specific commands. Since not all vehicles have the LLP capability, the planners also decompose high-level plans into smaller building blocks, which are translated into vehicle-specific commands. For example, a high-level command to survey an area can be further decomposed into a set of waypoints to represent the lawn mowing pattern for the survey. A vehicle with an onboard low-level planner is capable of receiving this high-level command and producing the decomposed version internally. On the other hand, a vehicle that does not have an onboard planner will require that this decomposing be done on a higher level in the system. This is achieved by running a vehicle agnostic low-level planner which decomposes the plan for a virtual vehicle and forwards it to the MTRR component of the MW.

The communication between MMT, MW, and the planners is based on Apache Thrift Framework [14]. A shared description of missions and their components is used amongst MMT, MW, and planners and each provides services that can be accessed through the same framework. These services are:

- computePlan: This service is provided by planners and allows the MMT to send a mission to the planner. The mission object only contains the mission goals and is void of any detailed list of tasks that the involved hardware can/should perform. The planner then fetches this data and computes a high-level plan based on the provided mission goals. It then, in turn, calls back the sendPlan service provided by MMT. If any error occurs during the computation, then the sendError service is invoked instead.
- sendPlan: This service is provided by MMT and is activated by the planner when a plan is ready. This is a complete mission object which contains a detailed list of actions to be taken by involved parties in the mission to achieve the goals.
- sendError: This service which is provided by MMT is invoked when an error occurs in one of the external components (in this case the planner, but it can be other components that use Thrift interface to communicate with MMT, such as MW, as well).

The shared mission description is based on important parameters that describe a mission, namely: locations/areas, vehicles, tasks, and the relationships between them. The mission description defines a goal when the only relationships present are the relationships between locations/areas and tasks. Thus, before planning is performed, it only contains the information regarding the tasks that should be performed on a location/area.

The High-Level Planner is responsible for adding extra information to the mission object, defining the relationship between the vehicles, tasks, and the order in which the tasks should be performed. Other planners can then: plan the path for each vehicle and decompose the tasks into smaller so-called sub-tasks, e.g. a survey on an area needs to be decomposed into a set of waypoints producing the lawn mowing pattern. These sub-tasks that relate to a main task are named as actions in the mission object. When these relationships are defined and the list of actions is produced, planning is complete.

From the perspective of the operator, the MMT's core services are defining, planning, monitoring, controlling, and analyzing missions:

- Defining a Mission: Missions that will be carried out in the real-world must firstly be defined. This means that an area for the mission site is marked in the map shown in the MMT. Typically, during the mission, one or more

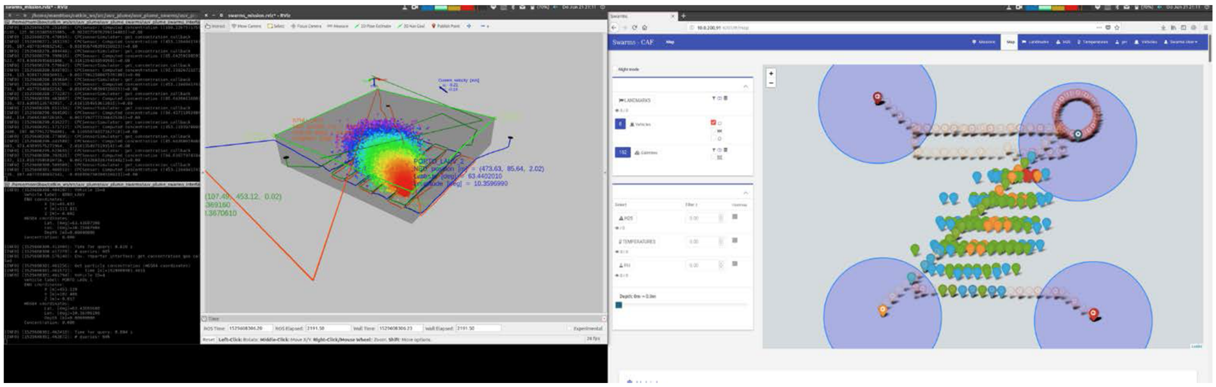

Figure 5. Visualization of computer simulated (left) and detected (right) wastewater plume.

AUV activities will be performed in the mission area. These tasks are directly based on vehicle capabilities and available sensors. The important part is that the resources, i.e. AUVs, and other hardware systems are known to the MMT, which requires that they are defined in the system's Middleware (MW).

- Planning a mission: Designing a system that can be used for planning multi-robot systems means that the system should provide means for transforming a mission defined and understood by humans, to a plan interpreted by machines. In the context of multi-robot systems, another assumption is heterogeneity with respect to the robotic systems. In the MMT the operator can define mission plans on a high-level, which later can be interpreted by the vehicles' onboard planners. The outcome of this three-step process, i.e. human to high-level plan to vehicle-specific low-level plan transformation. In the MMT the result of the high-level planning can be presented on the map and as a Gantt chart in the Plan Outline Panel (Fig. 2).

- Monitoring a mission: Monitoring is done using two main means, i.e. the Gantt chart at the bottom window of the MMT GUI, and the large window in the middle showing the map (the main window in the middle, Figs. 2 and 4). The former's objective is to show the progress of the mission, whereas the latter provides spatial cues such as the location of the vehicles.

- Supervising a mission: Whereas monitoring is a passive process, in supervision the operators can change the content of the mission. One critical case is when the mission needs to be halted or terminated due to an unexpected event. However, also during the normal progress of the mission there may be a need for interaction between the operator and any of the vehicles, or their equipment.

- Analyzing a mission: From the moment a mission starts the MW starts to collect data. In addition to the MW itself, which needs to analyze the collected data to monitor the mission, the operator will need to analyze data during and post-mission.

Planning functionalities of MMT are accessible through the planning dialog which lists current online planners. The operator can send the mission to one or more of the planners. When the planning is done, each planner sends the results back to MMT which in turn shows the plan parameters (vehicles used and the total time of executing the plan) to the operator on the same dialog. The operator can then choose the desired plan and view it on the map.

III. Experimental Results

The overall purpose of the experiments was to demonstrate that the proposed MMT-based solution can support the operators regarding preparing, planning, execution, and monitoring a mission, followed by recovering the deployed AUVs. The experiments took place in the waters of the Trondheim fjord, Norway (Coordinates: Lat. 63.4430, Lng. 10.3651), in June 2018.

The MMT was evaluated through a set of objectives, which were detection, inspection, and tracking of plumes of wastewater, at different depths. The plume, and hence the salinity levels were simulated during the mission, due to environmental considerations. The simulated plume and the involved vehicles were visualized in the Context Awareness Framework (CAF), which is a tool that can show simulated data (Fig. 5). Thus, CAF is also used for validation purposes. The main information flow is as follows: Operator → MMT → MW → vehicles → MW → (MMT & CAF). In the experiment three different depths were assumed, i.e. 30, 35, and 40 meters. This means that the AUVs had to scan an area at different depths by measuring the salinity of the water. The AUVs send their status vectors including location (Lat, Lng, and Depth), battery level, task status, and sensor readings to the MW. This allowed the MMT to fetch this data and present it to the operator.

The location was chosen because of its proximity to the Norwegian University of Science and Technology (NTNU) Trondheim's Biologic Research Center, which was providing the command and control station room. From a high-level planning perspective, the types of tasks that the AUVs could perform included simple point-tasks, such as taking a photo at a given location, or more complex area-tasks where the AUV has to scan an area while using its sensors and communicating with the MMT.

An Unmanned Surface Vehicle (USV) was also deployed to act as a WiFi network link and recover the AUVs after the mission. Thus, the USV functioned as a relay station between the command control station and the AUVs. The maritime traffic was also being observed during the mission. Four vehicles were involved in this mission: 1 Telemetron USV from Maritime Robotics, 1 Fridtjof Light Autonomous Underwater Vehicle (LAUV) from NTNU, and 2 Noptilus LAUVs from LSTS Lab in the University of Porto (Fig. 3). All the vehicles were equipped with underwater acoustic modems.

In order to plan the mission, the operator executes the High-Level planners. Note that in the planning module there are 2 independent planners. The planning process involved defining the mission area and the type of the task to be performed (in this case scanning an area using salinity sensors). The High-Level planner then chooses the vehicles based on their capabilities and assigns different tasks to each vehicle. This plan is then provided to the operator through MMT. Since MMT can integrate with more than one planner, the operator has the option to choose the preferred plan (based on time, battery levels, etc.) and execute it. In Fig. 4 the execution of the plan is shown. Note that the actual path taken by the AUVs is shown as dotted lines while the solid lines present the planned path. Also note that since all the AUVs scan the same area at different depths, the area is seen as a single area on the 2D map. Vehicle depths are shown next to the vehicle name on the map and are also available through the properties panel to the user (see the Mission Explorer window, Fig. 4).

During this mission, the three AUVs were planned to scan the water at different depths (30 meters, 35 meters, and 40 meters). The USV was also involved in the plan which would provide underwater communication means, and meet the AUVs at the extraction point to collect them. During the mission, the operator was able to view the progress of the mission both on the map and also on the "mission outline" Gantt chart. The Gantt chart shows a timeline of the tasks that each AUV has to perform. The tasks which are finished are greyed out and the current task is highlighted to keep the operator informed of mission status for each vehicle. The vehicles can also send error messages to MMT if something goes wrong during the mission. The operator can also abort a vehicle's plan or the whole mission if necessary. These abort commands are categorized into two different sets:

- Soft Abort: Aborts the current action. The vehicle waits in location for further instructions. This is useful in cases where the operator wants to re-plan and continue with the mission.
- Hard Abort: All the actions are aborted and the vehicle floats to the surface. The telemetry information received from the vehicle allows the operator to know the location for recovery purposes if necessary.

The mission was performed on 3 different occasions. During one of the runs, one of the AUVs sent a Propulsion error message and aborted its mission and surfaced. The error and the surface extraction point were received at MMT and helped the extraction team to recover the AUV. It was later discovered that the reason for the error was algae which were stuck in the propeller. Upon receiving the error message on MMT, the operator had the option to cancel the mission, re-plan it, or continue with the mission with one less AUV. The operator in this case chose to continue the mission and no re-planning was performed.

The experiments have shown that MMT is capable of preparing, planning, executing, and supervising multi-robot missions. The planning algorithms integrated into MMT also made it easy for the operator to define their set of goals while leaving the actual planning to the planning algorithms. The map and the plan outline kept the operator aware of the vehicles' and mission status by visualizing the information in different forms. The AUVs detected the plum, which was the main goal of the mission (Fig. 5).

IV. Conclusion

The real-world experiments have shown that a complex system consisting of software and hardware solutions, and heterogenous types of underwater and surface vehicles can be orchestrated by the MMT, which provides services for mission definition, planning as part of pre-mission activities, and orchestration throughout the mission, and data analysis post-mission. Through the Apache Thrift Framework interface solution, it allows extension of its capabilities. to address specific mission requirements. The MMT is a generic solution, which can be extended to Unmanned Aerial Vehicles (UAV), as well as other types of (semi-)autonomous systems.

Acknowledgment

The research leading to the presented results has been undertaken within the SWARMs European project (Smart and Networking Underwater Robots in Cooperation Meshes), under Grant Agreement n. 662107-SWARMs-ECSEL-2014-1, which is partially supported by the ECSEL JU/Horizon 2020 and Vinnova.

References

[1] Fernández-Macho, J., González, P., & Virto, J. (2016). An index to assess maritime importance in the European Atlantic economy. *Marine Policy*, *64*, 72-81.

[2] Branch, A., & Stopford, M. (2013). *Maritime economics*. Routledge.

[3] Carreno, Y., Pairet, È., Petillot, Y., & Petrick, R. P. (2020, June). A Decentralised Strategy for Heterogeneous AUV Missions via Goal Distribution and Temporal Planning. In *Proceedings of the International Conference on Automated Planning and Scheduling* (Vol. 30, pp. 431-439).

[4] Brito, M. P., Griffiths, G., & Challenor, P. (2010). Risk analysis for autonomous underwater vehicle operations in extreme environments. *Risk Analysis: An International Journal*, *30*(12), 1771-1788.

[5] Ebel, K. H. (1986). The impact of industrial robots on the world of work. *Int'l Lab. Rev.*, *125*, 39.

[6] Bogue, R. (2015). Underwater robots: a review of technologies and applications. *Industrial Robot: An International Journal*.

[7] Jain, S. K., Mohammad, S., Bora, S., & Singh, M. (2015). A review paper on: autonomous underwater vehicle. *International Journal of Scientific & Engineering Research*, *6*(2), 38.

[8] Nicholson, J. W., & Healey, A. J. (2008). The present state of autonomous underwater vehicle (AUV) applications and technologies. *Marine Technology Society Journal*, *42*(1), 44-51.

[9] Miloradović, B., Çürüklü, B., Ekström, M. and Papadopoulos, A.V., 2019, February. A genetic algorithm approach to multi-agent mission

planning problems. In International Conference on Operations Research and Enterprise Systems (pp. 109-134). Springer, Cham.

[10] Landa-Torres I, Manjarres D, Bilbao S, Del Ser J. Underwater Robot Task Planning Using Multi-Objective Meta-Heuristics. *Sensors*. 2017; 17(4):762.

[11] Rodríguez-Molina, J., Bilbao, S., Martínez, B., Frasheri, M., & Cürüklü, B. (2017). An optimized, data distribution service-based solution for reliable data exchange among autonomous underwater vehicles. *Sensors*, *17*(8), 1802.

[12] Martínez, N. L., Martínez-Ortega, J. F., Rodríguez-Molina, J., & Zhai, Z. (2020). Proposal of an Automated Mission Manager for Cooperative Autonomous Underwater Vehicles. *Applied Sciences*, *10*(3), 855.

[13] Martínez, N. L., Martínez-Ortega, J. F., Castillejo, P., & Beltrán Martínez, V. (2020). Survey of Mission Planning and Management Architectures for Underwater Cooperative Robotics Operations. *Applied Sciences*, *10*(3), 1086.

[14] Slee, M., Agarwal, A., & Kwiatkowski, M. (2007). Thrift: Scalable cross-language services implementation. *Facebook White Paper*, *5*(8).

[15] Li, X., Bilbao, S., Martín-Wanton, T., Bastos, J., & Rodriguez, J. (2017). SWARMs ontology: A common information model for the cooperation of underwater robots. *Sensors*, *17*(3), 569.

[16] Li, X., Bilbao, S., Martín-Wanton, T., Bastos, J., & Rodriguez, J. (2017). SWARMs ontology: A common information model for the cooperation of underwater robots. *Sensors*, *17*(3), 569.

[17] SWARMS. Available online: http://swarms.eu/overview.html (accessed on 13 September 2020).

[18] Patrón, P., & Petillot, Y. (2008). The underwater environment: A challenge for planning. *UK PlanSIG Edinburgh*.

[19] Kothari, M., Pinto, J., Prabhu, V. S., Ribeiro, P., De Sousa, J. B., & Sujit, P. B. (2012). Robust mission planning for underwater applications: Issues and challenges. *IFAC Proceedings Volumes*, *45*(5), 223-229.

[20] Akyildiz, I. F., Pompili, D., & Melodia, T. (2004). Challenges for efficient communication in underwater acoustic sensor networks. *ACM Sigbed Review*, *1*(2), 3-8.