Advancing Multi-Vehicle Deployments in Oceanographic Field Experiments

António Sérgio Ferreira · Maria Costa · Frédéric Py · José Pinto · Mónica A. Silva · Alex Nimmo Smith · Tor Arne Johansen · João Borges de Sousa · Kanna Rajan

Received: 26 February 2018 / Accepted: 3 October 2018

Abstract Our research concerns the coordination and control of robotic vehicles for upper water-column oceanographic observations. In such an environment, operating multiple vehicles to observe dynamic oceanographic phenomena, such as ocean processes and marine life, from fronts to cetaceans, has required that we design, implement and operate software, methods and processes which can support opportunistic needs in real-world settings with substantial constraints. In this work, an approach for coordinated measurements using such platforms, which relate directly to task outcomes, is presented. We show the use and operational value of a new Artificial Intelligence (AI) based mixed-initiative system for handling multiple platforms along with the networked infrastructure support needed to conduct such operations in the open sea. We articulate the need and use of a range of middleware architectures, critical for such deployments and ground this in the context of a field experiment in open waters of the mid-Atlantic in the summer of 2015.

Keywords Mixed-Initiative control, Marine Robotics, Control, Artificial Intelligence, Ocean science, Operational Oceanography, Upper water-column biology

1 Introduction

Oceanographic field experiments, targeting large-scale dynamic phenomena, typically require the coordination of multiple manned and unmanned assets to deal with the spatio-temporal variability of the upper water-column in the ocean. The coordinated execution of these multiple assets, critical for co-temporal observations, is a difficult challenge due to a myriad of operational issues ranging from limited communication range, bandwidth and uptime, to difficulty in creating an accurate global view for asset location, supervision, awareness and operation. In the past, we have undertaken a series of field experiments with unmanned vehicles under such challenging unstructured conditions in the context of inter-disciplinary applications not only of scientific interest to oceanographers [12,18] and biologists [39,52], but also to first responders [9] and the military [36]. Uncertainty and unpredictability are often the norm, thus making operations extensively challenging. This is further compounded when the target applications involve responding to dynamic and unpredictable phenomena in such unstructured environments, through the combined use of autonomous underwater vehicles (AUVs) and unmanned aerial vehicles (UAVs).

António Sérgio Ferreira · Maria Costa · Frédéric Py*, · José Pinto · João Borges de Sousa
Faculdade de Engenharia da Universidade do Porto, Portugal
E-mail: {asbf,mcosta,zepinto,jtasso}@lsts.pt,fredpy@gmail.com

Mónica A. Silva
IMAR-Açores & MARE
Marine and Environmental Sciences Center, Horta, Portugal
E-mail: monica.silva.imar@gmail.com

Alex Nimmo Smith
School of Biological & Marine Sciences,
Plymouth University, UK
E-mail: alex.nimmo.smith@plymouth.ac.uk

Tor Arne Johansen
Center for Autonomous Marine Operations and Systems,
Department of Engineering Cybernetics,
Norwegian University of Science and Technology (NTNU), Trondheim, Norway
E-mail: tor.arne.johansen@ntnu.no

Kanna Rajan
Center for Autonomous Marine Operations and Systems,
Department of Engineering Cybernetics,
NTNU, Trondheim, Norway &
Faculdade de Engenharia da Universidade do Porto, Portugal
E-mail: Kanna.Rajan@ntnu.no
* – now at SINTEF Digital, Trondheim, Norway

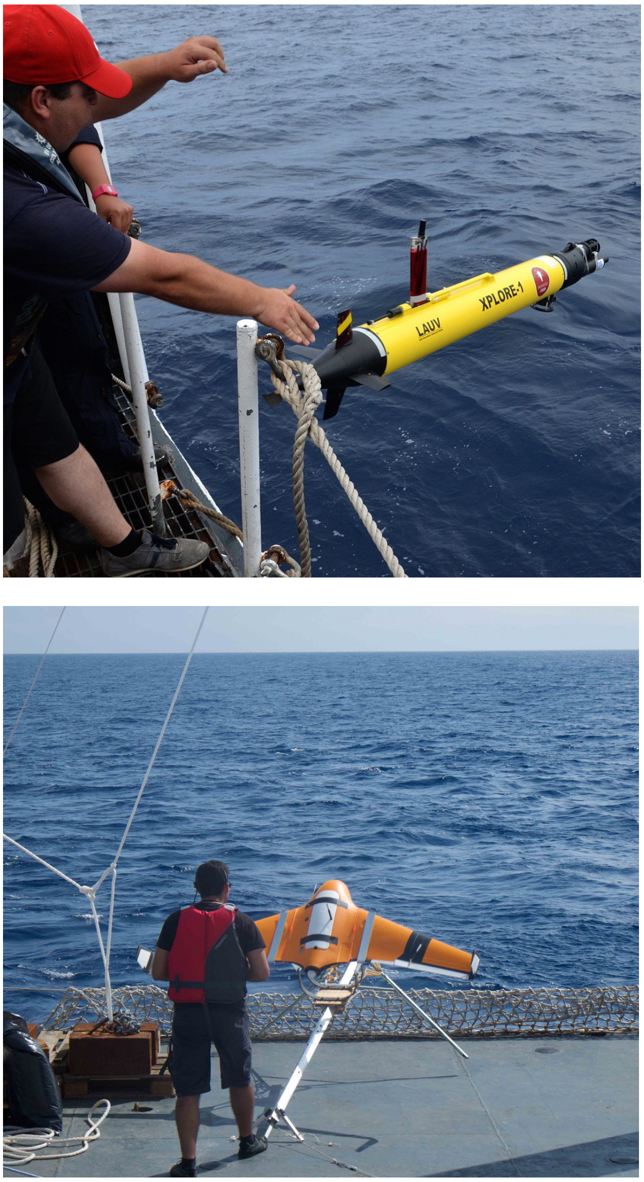


Fig. 1 AUV and UAV operations from the NRP Gago Coutinho in the Azores in July 2015.

Coordination implies the ability to envision task completion in light of unpredictability while dealing with all remaining operational constraints. Typically, coordination in marine robotics has been viewed as a means to demonstrate nominal engineering principles [17,27,2]. Our experiments use networked robotic platforms tied together with a mature set of tools for decision support, situational awareness, control, planning, data visualization and archiving for inter-disciplinary experimentation [20,40].

While the scenarios described have yield science data the focus of the work presented here is not on it's analysis, but rather to showcase advances in operational methodologies for marine robotics for inter-disciplinary experiments at sea, while drilling down to the feasibility of successfully applying such methods. Furthermore, it highlights the changing nature of oceanographic experiments, by articulating the tools, methods and operational artefacts which comprise a portable laboratory, including heterogeneous robotic platforms and their infrastructure. In addition, we articulate the need for a novel mixed-initiative constraint-based planner which aids operational robotic field experimentation. This tool, EUROPtus, surfaces as an extension to [9] by using temporal planning methods for operational control of vehicles, while assigning them specific coordination tasks. The EUROPtus planner is a ship/shore situated temporal constraint-based automated planner for field experiments. The planner does not attempt to model all constraints comprehensively and is used primarily for task assignment, information gathering and situational awareness of AUVs, all the while keeping a simple resource model of UAV operations.

We do so in the context of an actual field experiment in the deep ocean, off the Azores in the mid-Atlantic[1], where UAVs were used to spot cetaceans (Fig. 1) providing a GPS fix on their coordinates, as a means for targeted oceanographic measurements with AUVs in the water column. With this operational model, we aim to understand how far automation of field operations involving multiple coordinated assets can aid or hinder operational effectiveness. While EUROPtus has been field tested only in the specific operating environment for our experiment, we believe the principles behind it are general.

This paper is organized in the following manner: first we introduce the problem and challenges of multi-vehicle oceanographic deployments in Section 2; next, we briefly place the context of this work in Section 3 and describe the tools, techniques and process we use in Section 4. Section 5 explains the approach in automated planning we use in this work and, in Section 6, we document experimental results from the field deployment in the Azores. We conclude with Section 7, discussing the results and future work.

2 Scientific Motivation and Challenges

Oceanographic field experiments typically combine remote sensing with in situ observations using research vessels off of which sensors and robotic platforms are deployed. The research vessel is guided towards an operational area of interest, where opportunistic needs drive what and how the operations on-board are conducted. More recently, as mobile robotic platforms have become more robust and able to carry a diverse and useful scientific payload, experiments are driven by scientific hypothesis, typically covering the meso-scale (i.e.

---

[1] http://rep15.lsts.pt/

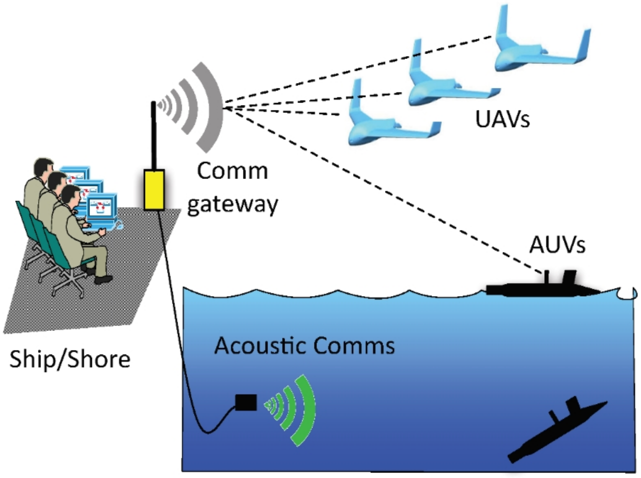


Fig. 2 Typical setup of operators on ship/shore for vehicle control in oceanographic field operations.

$> 50km^2$) regions repeatedly for weeks with such assets. Campaigns like the Autonomous Ocean Sampling Network (AOSN-I/-II) [10,47] and the Controlled Agile and Novel Observation Network (CANON) field program [13,11,12] are prominent examples, with a large number of robotic and/or manned assets for sampling and observation.

The traditional approach in operational oceanography, is for the use of individual robotic vehicles controlled by a planning tool with decision-support capability, with humans making all decisions a priori, to take measurements in a very uncertain, harsh and dynamic setting (Fig. 2). In these scenarios, situational awareness is often opaque, increasing operator workload [24] while prompting an increased need for operator-to-operator coordination. Our objective in this work, is to not only augment such methods with automated planning [30], but also attempt to allow a single operator to command multiple heterogeneous vehicles while keeping humans in the loop and provide maritime domain awareness.

While tele-presence techniques have had a visible impact in the ocean science community [53,32], a key issue continues to be that of situational awareness, namely where a sensor or platform is, and what it is observing, over space and time. Unpredictable conditions over and below the sea surface and its typically harsh environment, preclude full knowledge at all times. Adaptability and high-level guidelines are thus preferred, so as to allow accommodation to local perceived phenomena, while still fulfilling the experiment's objectives and enforcing safety constraints.

As robotic platforms become more ubiquitous, the complexity of dealing with multiple assets for simultaneous, coordinated and co-temporal observations, has added to this challenge. This is prevalent in the operational scenario addressed in this paper, of cetacean tracking that underpins the focus of this work.

In such a domain, understanding the processes which influence the distribution and ecology of animals is a fundamental problem with important implications for our understanding of ecosystem function and dynamics, and for conservation and management of natural resources. Most studies investigating the environmental drivers of marine animal distribution have relied on remote-sensed data or existing oceanographic models. However, a more comprehensive understanding of the movements and behavior of marine animals and how they interact with the environment requires more than averaged observations, often collected at coarse spatial and temporal scales – animal derived data needs to be integrated with more synoptic oceanographic approaches [19]. Standard methods to collect concurrent observations of oceanographic conditions and of the distribution and behavior of cetaceans use large, expensive ocean going vessels. As a consequence, oceanographic sampling is limited to a few stations providing but a snapshot of oceanographic parameters.

In addition, operation of large vessels may interfere with natural animal behavior. Recent advances in electronic transmitters and data-storage tags have made it possible to collect 3D high-resolution biological and oceanographic observations from animals in their habitats. However, because of satellite availability and transmission costs, tag recovery is usually necessary in order to access collected data, limiting the use of these tags to animals that reliably return to the same location or remain within a small geographic area, and shortening deployment durations. Moreover, as these tags remain relatively large they cannot be attached to most free-ranging cetaceans without the potential to impact their behavior. Because of these limitations, research into the oceanographic drivers of cetacean distribution continue to rely on the analysis of sighting data or low-resolution satellite tags and remote sensed variables [5,48]. These data can be used to correlate estimated positions from surfacing cetaceans with remote sensed or model-derived oceanographic variables, usually available at sparse temporal resolutions.

Using unmanned assets, deployed at sea, enable cost-effective near real-time monitoring of ocean processes and cetacean movements and behavior. UAVs can be deployed in the areas of interest and tasked to follow cetacean positions in an autonomous manner taking into account detection, classification and tracking methods [28]. AUVs can then be deployed to sample the water column on the surfacing locations, in order to provide complete synoptic observations around and above the target individual(s) – AUVs are also less prone to interfere with normal behavior of animals due to their reduced footprint. Finally, if AUVs and UAVs can perform co-temporal observations, a portable observatory can be created where animals and surrounding conditions are observed continuously in their natural habitat.

However, the current generation of AUVs can face challenges of either speed or endurance when tracking fast moving cetaceans. Typically, vehicles that can stay in the water for extended periods of time are only capable of speeds below 1 m/s [54,31]. Moreover, communication with these vehicles is also limited: only satellite communications are available at remote locations and only when vehicles are at the surface. Equally, acoustic communications are noise prone and have limited bandwidth and range. Thus the current capabilities of vehicles can only provide sparse location information or engineering telemetry at best. In addition, although UAVs can be continuously connected to control stations by using short-range (tens of kilometres) point-to-point communications, or alternatively by using expensive intermediate (satellite) gateways, their endurance ranges to a few hours at best.

Finally, coordination of UAVs and AUVs carries its own challenges, requiring the fusion of two very different vehicle types, with specific communication, locomotion and sensor payloads, into a unified domain where they share common objectives. Due to the high variability of ocean conditions, it may be required to adapt plan execution on board the vehicles according to the perceived environmental conditions. The computational power available on board these vehicles, however, is limited for running on board planning algorithms.

3 Related Work

Coordination of multiple heterogeneous robots requires a common communication infrastructure that is used to share the state of the different intervening assets to perform collaborative behavior. There are various open source robotic frameworks that can be used to coordinate multiple robots, but usually these target controlled environments, where reliable and continuous communication are available.

The Mission-Oriented Operating Suite (MOOS)[38] relies on a robust communications library to share data between applications, using a centralized node called MOOSDB. This node acts as a knowledge information base where all other nodes can send information to and receive information updates. The widely used Robotic Operating System (ROS), uses a publish/subscribe communication system that relies on a central node service for node registration and look-up, called the master node [43]. As such, whenever separate robots are to be coordinated, they should use a common master node running in either of the two or in another accessible network location turning it into a central point of failure. More recently, the possibility to use ROS in a multi-master architecture was introduced where multiple master nodes co-exist and exchange information being, however, still required to be aware of each other prior to their execution. The recently-developed GAMS middleware supports collaborative autonomy amongst heterogeneous teams of agents using non-blocking, best-effort communications protocols, which allows operation even in communications constrained environments [15]. All nodes under a GAMS collaboration system have their own knowledge base which, using MADARA, synchronizes relevant information with peers when that information is changed [16]. The LSTS Toolchain [40,20], used in this work, was developed to support different robot and communications hardware, mission purposes and operator needs. Reference implementations are freely available for AUVs, ASVs, ROVs and UAVs[1]. In its case, the Inter-Module Communication protocol (IMC) defines transport-agnostic discovery mechanisms, allowing it to be used in unreliable and heterogeneous networks using any available links such as acoustic, Iridium or Wi-Fi. Dealing with uncertainty in communications, especially in the context of oceanographic field experiments, is therefore key to the tool-chain. For instance, IMC targets unreliable communication environments and provides dynamic discovery algorithm implementations. All messages in IMC are time-tagged and include information about its origin and destination sub-systems.

Besides considering the communication needs for coordinated operation of these systems, vehicle autonomy is also a factor. Onboard autonomy can vary from the very basic tele-operation to fully intelligent systems that generate and adapt plans on board, in response to objectives and environment. In oceanographic experiments, simultaneous operation of multiple vehicles is typically along institutional lines, with a group of individuals focusing on one specific robotic platform [47]. Our work is unusual in that a small group is focused not only on a collection of vehicles, but that they are also heterogeneous; in such cases, the amount of attention that can be devoted to each entity is often limited. Compounding this, autonomous operation of UAVs is usually done by defining a sequential list of waypoints that the vehicle must follow. This is a very efficient approach for static and collision-free environments while requiring very limited onboard autonomy. Moreover, some execution environments also have triggers that use contingency waypoints whenever errors are diagnosed. However, in more dynamic and unknown scenarios or whenever the phenomena of interest is unpredictable, a more deliberative planning is required. There are a number of approaches to deliberation.

MOOS-IvP Helm is a behavior-based architecture that uses multi-objective optimization [3]. In MOOS-IvP, different behaviors may have different priorities

---

[1] http://github.com/LSTS

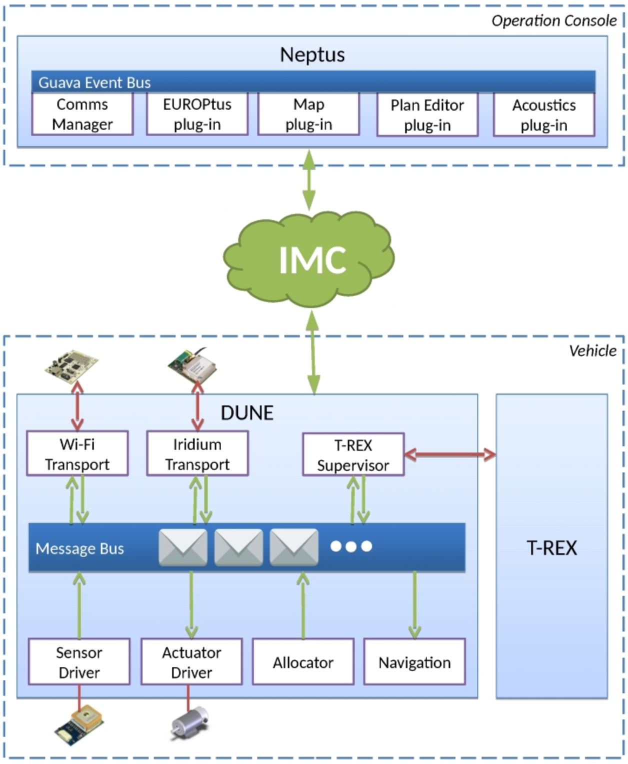


Fig. 3 The software components used to control unmanned vehicles with the LSTS toolchain.

and triggering mechanisms and can be simultaneously active, in which case they compete to define the ongoing actions of the vehicle. If all behaviors have different priorities, the resulting architecture can be equivalent to Brooks' subsumption architecture [8]. However, whenever two competing behaviors become active, the Helm selects an action by maximizing a resulting value. NASA developed MAPGEN (Mixed Initiative Activity Plan Generator) combining the EUROPA planner [26] with a user interface that allows human operators to interactively define plans, by providing an initial state, prioritized goals, standard constraints and daily constraints [1,6,7]. Such Artificial Intelligence (AI) based mixed-initiative methods for planning, continue to be novel in the oceanographic domain. Typically, a map-based user interface is used as a planning tool, where the 'plan' is essentially a sequence of waypoints carefully constructed by the human operator, dispatched to a vehicle out at sea. Such software systems offer basic decision-support capability, with little in the way of machine intelligence with human decision-making driving robotic activity. NEPTUS [40], which we later describe, is one such mature tool in this category and we build on its functionality.

In this work we extend the shore-based mixed-initiative planner demonstrated in [9] for controlling multiple AUVs, where the underlying automated planner did not allow re-planning to deal with environmental uncertainty or dynamic events, nor reasoning about time and resources, as this work does. Doing so, allows us to use active simulation on shore or ship to be coupled with the proscribed execution trace while keeping the operator situationally aware. Furthermore, in this work, we use the same temporal formalism on board our AUVs as we used in T-REX in the context of upper water-column exploration [42,46]. Finally, we are informed by efforts on the command/control of the Opportunity rover on Mars [1,6] using a similar planning approach we use here. The operating domain at sea however, is far harsher and substantially more constrained with a far more dynamic pace and operational fluidity.

In summation, the application deals with coordinated measurements using robotic platforms, where the coordination is related to task outcomes and not specific navigational needs. While [49] also demonstrates the use of multiple robotic assets in the water-column, albeit in shallow near-shore waters, to the best of our knowledge using automated planning [30] as a means to provide abstraction in control over single or multiple vehicles in the oceanographic domain, as this work does, continues to be novel.

The work in this manuscript builds upon the LSTS toolchain which has extensive operational use in recurring large scale exercises [34,51], is open-source and with a growing group of users and contributors such as Norwegian University of Science and Technology (NTNU), Monterey Bay Aquarium Research Institute (MBARI), Laboratory for Underwater Systems and Technologies (LABUST), Natural Environment Research Council (NERC) and Centre for Maritime Research and Experimentation (NATO-CMRE), among others. Furthermore, the use of the LSTS toolchain as a baseline proves advantageous for scalable use of the tool due to the ongoing adaptation of the toolchain to other communication protocols (e.g., MAVLink, JANUS, MOOS) and compatibility with vehicles from other manufacturers (e.g., LRAUV, Waveglider, IVER, SPARUS II). These adaptation and integration efforts open up the range of what can be done, operationally with this work.

4 Technical Approach

In order to control a set of heterogeneous platforms, a number of components needed to be developed or adapted. Our software architecture is built on top of the LSTS Toolchain [40], and uses the DUNE onboard software for control and navigation of all UAVs and AUVs. IMC is used across all systems to exchange information and NEPTUS has been extended for providing global situational awareness and supervision for human operators (Fig. 3).

4.1 The Command and Control System

The DUNE (DUNE Uniform Navigation Environment) onboard software has been developed to support all its unmanned vehicles and other embedded systems such as communication gateways and data loggers [40]. It provides a uniform navigation environment that is irrespective of sensors used to estimate a system position (localization) or actuators used to interact with the environment; there exist standardized structures that can be used for abstract representations of estimated and desired system states. This allows sharing most of the code between all vehicles and being required to adapt only low level actuation and sensing.

DUNE uses a publish/subscribe system for message-passing between modules that run concurrently in a local system or distributed over the network. All messages flowing within and outside a DUNE system are defined in IMC . A conceptual view of a DUNE based system, where device drivers and transport components (which handle communication with the world) is illustrated in Fig. 3.

NEPTUS is a software infrastructure for developing user applications that interface Networked Vehicle Systems [14]. It is currently used for operating fleets of heterogeneous autonomous vehicles by diverse end-users such as biologists, oceanographers, archaeologists, students or the military. It is used not only during planning, but also to supervise execution and revise data after mission execution [41]. It uses an extensible architecture which supports different plug-in types: Console Widgets, Console Daemons, Interactive Map Layers, Interactive Data Visualizations, Plots, etc. Inside the software, all plug-ins interact with each other via Google's Guava Event Bus[1], a library that enables loosely coupled communication between components via asynchronous message-passing, similarly to what happens in DUNE. Operation consoles, for instance, can be configured to match the needs of specific mission objectives or operator preferences by defining active plug-ins and their configurations . Even though DUNE and NEPTUS were developed in two different programming languages (C++ vs Java, respectively), the two systems share most of the design patterns and development approach.

4.2 Onboard Plan Deliberation

In order to cope with uncertainty and environment interference during execution, vehicles need to adapt their actions according to the perceived environment. In addition, they also need to potentially change the order and set of actions to execute, in order to maintain vehicle safety while attaining the most important mission objectives.

The problem of running computationally-intensive deliberation systems on board cpu-constrained AUVs has been approached by developing a new mission execution system named Teleo-Reactive EXecutive (T-REX) [35,42,45,46]. T-REX [2] allows creation of deliberative agents by having simultaneous execution of multiple planners with heterogeneous latencies (time to produce a new plan based on new observations), as well as different planning windows (time span of produced plans). This "divide-and-conquer" strategy for problem-solving is encased in this decomposition of the T-REX agent into different software components (reactors) making it possible to have planning and reactive behaviors running in parallel and continuously.

T-REX does this by providing a software framework where different reactors interact with each other using timelines: a flexible representation to describe the evolution of a given state variable over time. Reactors provide internal (controlled) timelines where they publish state updates (observations) and use external timelines controlled by other reactors in order to read its state and post desired goal states. Deliberative reactors produce a plan to attain the desired goal states if they fall within their planning window, and do so with the underlying use of the EUROPA planner.

EUROPA is an AI-based planning system used to generate plans executed onboard NASA's 1999 Deep Space One probe with the Remote Agent Experiment (RAX) [4,37,26,44] and on the ground as a mixed-initiative planner for the 2003 Mars Exploration Rovers mission with the MAPGEN system [1,6,7]. EUROPA is a vastly re-factored, higher performance open-source[3] version of the systems used here and uses a domain model written in NDDL (New Domain Description Language), together with initial conditions and goals (also in NDDL), to construct a set of temporal relations that must be true at start time. These models include assertions about the physics of the vehicle, i.e., how it responds to external stimulus and internally driven goals. An in-depth description of how domain models are defined and plans deliberated is provided in Section 5.

A simplified version of our AUV's domain model is depicted in Fig. 4. The "Medium" timeline may be controlled by an Interface reactor that simply translates sensor readings into observations of the vehicle, being either underwater or at the surface. Since the AUVs are positively buoyant, whenever the vehicle is 'Idle' it will float towards the surface. Moreover, a deliberative reactor controls the "Navigator" and "Yoyo" timelines. Whenever the vehicle needs to reach a destination x, the Navigator reactor will produce a plan

---

[1] https://github.com/google/guava

[2] https://github.com/fredpy/trex2-agent/

[3] https://github.com/nasa/europa

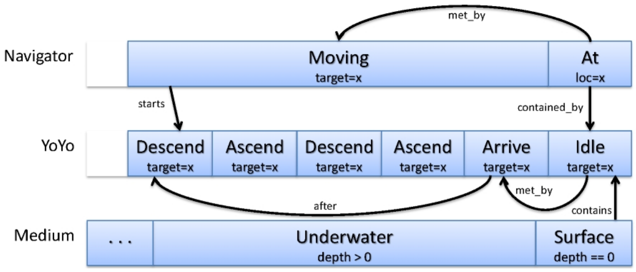


Fig. 4 An example of timelines (with their names on the left) and token relationships used on the domain model for AUV deliberative control.

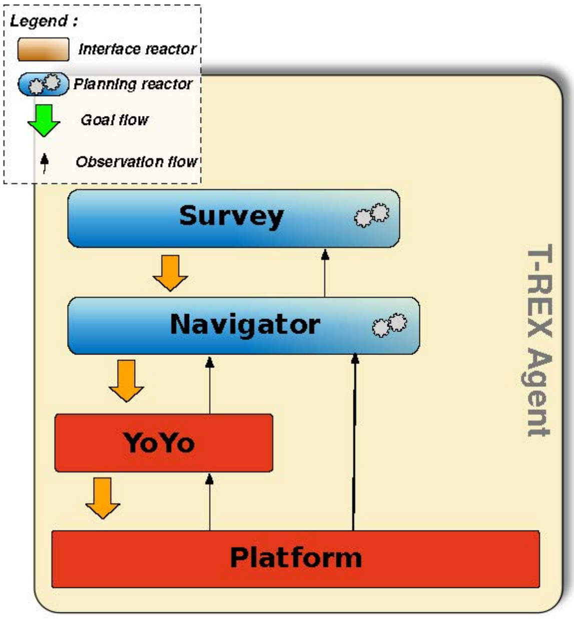


Fig. 5 The principal reactors running on the LSTS AUVs inside the T-REX framework for the July 2015 off of the Azores.

that projects the vehicle being at x at a future time. For that, the vehicle should start descending and then iteratively switch between Ascend and Descend modes, defined in the "Yoyo" timeline. The semantics of "arriving" at the destination implies stopping, in order to float to the surface. The undulating "Yoyo" behavior itself, is important for all underwater vehicles to capture the variability in the upper water-column across the vertical rather than the horizontal dimension.

Fig. 5 shows how different reactors can interact by exchanging goals and observations. From bottom to top, the reactors further abstract the environment and tend to have larger planning windows and latencies. The "Survey" reactor will produce plans that have several waypoints that the vehicle should travel to. These waypoints are passed as objectives to the "Navigator" reactor that, as described previously, will trigger a yoyo behavior in the vehicle.

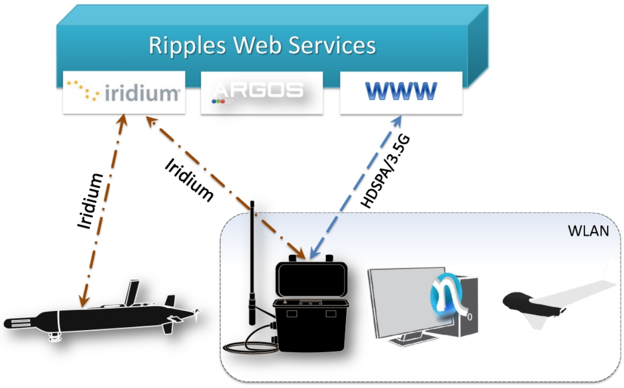


Fig. 6 Communication links between consoles and vehicles.

To integrate T-REX, running on an auxiliary CPU, with the LSTS Toolchain, a special DUNE task was developed that translates DUNE data into T-REX observations and parses incoming requests into vehicle actions. When T-REX receives a goal, its objective is to synthesize partial plans and dispatch these to DUNE, while simultaneously tracking their execution onboard a vehicle.

In order to translate T-REX requests into DUNE actions, DUNE was extended to allow reception of external control references such as desired speed, depth/altitude, target position and/or heading via a "back-seat driver" API. The API accepts both complete or incomplete reference definitions. For instance, if so desired, the external controller can specify a speed to be maintained while travelling towards the target pose or choose to not specify it. As a result, the back-seat driver will use an optimal (and possibly varying) speed while going to the target. Something similar occurs with altitude, depth and hovering radius[1] actions. The flexibility so provided allows for the development of external controllers, which are not tied to specific vehicle hardware, allowing DUNE to use provided slack to improve vehicle safety, navigation or battery optimization.

4.3 Communications Infrastructure

A cornerstone of multi-vehicle operations is its communications infrastructure. In the case of oceanographic field operations, heterogeneous and opportunistic links using a mix of Wi-Fi, satellite communications, GSM and acoustics are critical. Our approach has been to use the IMC protocol across all systems, irrespective of the available underlying transport mechanism. This provides additional flexibility, in that functional parts of the software can be either implemented on board the vehicles or distributed on the network and accessed us-

---

[1] The acceptable distance to the desired end point

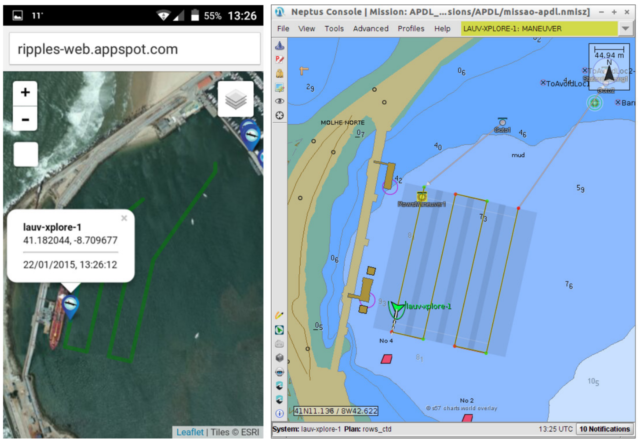


Fig. 7 The Ripples HTML interface visualized in a mobile browser (left) and the same data being accessed by Neptus (right)

ing any of the available communication methods. Moreover, the portability provided by DUNE allows it to run not only on vehicles but also on communication gateways, which can be used to extend the network and bridge different communication methods. These gateways allow operator consoles to establish a connection via Wi-Fi or Ethernet and then use its acoustic, Iridium or GSM modems to communicate IMC data.

IMC embraces heterogeneous means of communication by allowing nodes to be discovered over different interfaces and announcing its provided capabilities and names using transport-agnostic identifiers. For instance, a vehicle may be discovered by another using UDP multicast but afterwards, communication can be maintained between the two using TCP, UDP, HTTP, acoustic modem, etc. Deciding which communication method to use is the responsibility of the software and not the protocol; however, efficient implementations are available in NEPTUS and DUNE that can try all different reliable protocols (in order of degrading bandwidth) until the message is successfully delivered at its destination [33].

Vehicle operations must ingest the data coming from multiple sources and allow variable latency for incoming data and outgoing commands. For aggregating data from multiple sources we use a centralized communications system, named Ripples, running in the cloud and developed using the Google App Engine. This cloud-based application ingests data pushed through Iridium, Globalstar and Argos satellite communications, as well as several web entry points, for posting real-time and/or historical information (Fig. 6). All incoming data is stored in the cloud and can be forwarded to other recipients, via Iridium or Web Sockets. Ripples also provides a web interface where the telemetry and real-time logs of field events are accessible via a browser (Fig. 7).

To allow operations even in situations where the Internet is not available, such as the open sea, Ripples allows on-demand subscription of Iridium updates. A special entry point is used to subscribe/unsubscribe via Iridium. Whenever a message is sent to all systems (special IMC broadcast destination), it is forwarded to all subscribed systems.

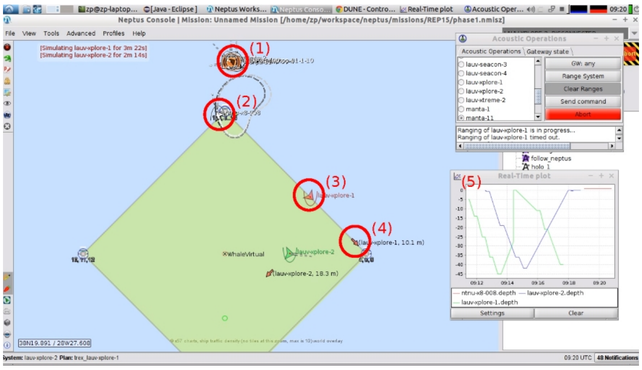
Irrespective of how data arrives at the Manta gateway – Wi-Fi, Web, acoustic modem or Iridium – it is dispatched to all connected NEPTUS consoles. The received information is then processed by any NEPTUS plug-ins that have subscribed to that message type. To distinguish the messages arriving from multiple sources and means, the IMC message header includes information such as identity of the generating system and time of message generation. The Mantas can then be used to send commands back to the vehicles using the available links. In practice, message flow from the vehicles to consoles using Iridium is as follows:

1. Vehicle arrives at surface and is not connected to any console;
2. Vehicle sends state update via Iridium not addressing it to any system;
3. An Iridium service provider forwards Iridium messages to Ripples using a POST request;
4. Ripples verifies the list of active Iridium subscribers and forwards the message to them, serially;
5. The message arrives at an Iridium device inside the console's WLAN (manta gateway);
6. The gateway forwards incoming Iridium messages to all connected consoles.

A similar (in reverse) route is used to send commands to the vehicles but, in this case, messages are addressed to the target system or else they are forwarded to all Iridium subscribers as before.

4.4 The EUROPtus Mixed-Initiative Planner

Maritime field experiments usually require having multiple vehicles disconnected for substantial periods of time (in excess of $15-30$ minutes). In order to improve the operators' situational awareness of vehicle positions and progress, we used a number of simulated vehicles running off of commands sent to the actual platforms. This was done by running DUNE and T-REX simulators side-by-side with NEPTUS. NEPTUS was configured to forward any commands sent to the actual vehicles also to their respective shore-side simulators. As a consequence, the simulated vehicles execute the same set of commands albeit in an idealized environment. While doing so and in periods of loss of contact, operators can determine with clarity, what each asset is expected to be doing – clearly in off-nominal situations, this simulation does not reflect reality.


Neptus Console | Mission: Unnamed Mission [/home/zp/workspace/neptus/missions/REP15/phase1.nmisz]

[Simulating lauv-xplore-1 for 3m 22s]
[Simulating lauv-xplore-2 for 2m 14s]

Ranging of lauv-xplore-1 is in progress...
Ranging of lauv-xplore-1 timed out.

System: lauv-xplore-2 Plan: trex_lauv-xplore-1

Fig. 8 NEPTUS operator console during the cetacean-tracking experiment, July 2015. (1) is the location of the control station (onboard ship), (2) is the real-time location of a flying UAV, (3) is an AUV position received via acoustic modem 3 minutes before, (4) is the simulated position of the same AUV and (5) is a plot of data received via acoustic modem from the AUVs.

In the case of AUVs, any incoming updates (received over any available communication channels) are used to reset the simulated vehicle's localization filter, similarly to what is done when the vehicles arrive at the surface, as shown in Fig. 8.

In order to coordinate the actions of multiple vehicles, operators can use NEPTUS to override the current behavior on the vehicles using any of the available communication means. However, manual control and supervision of multiple heterogeneous vehicles becomes infeasible as the number of vehicles and latency increases. This happens due to the increased uncertainty and discontinuous changes regarding the perceived state of the system, in addition to information overload on the operator.

In order to help operators coordinate fleets of robotic vehicles, we have designed, built and tested EUROPtus, a ship/shore situated temporal constraint-based automated planner which deals with operational constraints for field experiments. EUROPtus leverages existing work in fully-autonomous AUV operations and couples that with the need to operate multiple (often heterogeneous) robotic vehicles. It has a simple resource model which is integrated into the planner to enable as complex a coordination model as necessary; by using EUROPtus, the operator offloads a portion of the planning task, which is quite relevant given networks with very high variability of vehicle configurations and capabilities (while interacting operational constraints and substantial operational telemetry from vehicles with varying speeds). The operator, however, is still in charge of providing high-level goals, supervising the plans sent to the vehicles and override them if necessary.

A special NEPTUS plug-in was developed to act as a routing device for incoming and outgoing messages. The plug-in redirects data from EUROPtus to controlled/simulated AUVs and, at the same time, reports all received updates to EUROPtus; conversely this plug-in also notifies UAV operators about new incoming requests – a new deploy request or survey area from EUROPtus, for instance.

When EUROPtus determines so, it can request new objectives for vehicles deployed in the field or prompt humans behind an operation console. For instance, operators are expected to execute some task and provide input (e.g., inspect collected data and determine a list of waypoints to be visited) or, EUROPtus generates a high-level objective that is sent to an autonomous vehicle to replan in situ accordingly. If a new plan is found, this is reported to the operator from which they can both provide new objectives and inputs, and/or recall existing objectives. This is the focus of such mixed-initiative interaction between vehicle(s) and operator.

5 Planning and Execution Approach

Managing a mission with interdependency of multiple assets, some human operated, is often a challenge. It becomes even more problematic when assets like AUVs are often out of communication range – either because they are underwater or simply because they have to operate at a safe distance from a control station – and acting for the most part autonomously. Coupled with dynamic operational constraints, including the impact of weather and sea state, a substantial part of the operation requires human decision-making. Equally, synchronizing the operation of multiple robotic assets with ship-based operational constraints can overwhelm the operator. For instance, in our experiment, our research vessel had to be oriented to have the aft deck facing a tail wind for stable UAV take-offs; but UAV landings on a net facing starboard, required the ship to pivot towards the wind coming from the port side. Unifying such complex constraints (which require ship movement, and coordination with the bridge, for instance) while robotic vehicles are in operation, continues to require human judgement and involvement.

We used a mixed approach to autonomous platform operation, where a planner and its executive on the ship are used to track and supervise as much as the execution of the mission as possible, to alleviate these operational constraints. This calls for an approach different from fully autonomous embedded planning/execution. Equally, in structured laboratory experiments, it is assumed that execution can be tracked and every observation from the agent(s) are fully observed and in a timely fashion; this is often not so in real-world environments, especially at sea.

We believe that planning and execution interact, and tightly so. Such interaction goes beyond the classical conception of interleaving planning and execution, such as in [29], which continues to regard planning as an off-line process interrupting execution until it identifies a complete plan. We see execution as an integral part of the overall deliberation process that both interrupts and enriches the planning as the world evolves. This key concept provides the backbone of how EUROPtus is designed.

5.1 EUROPA Planning Essentials

In EUROPtus, a plan is composed of a temporally scoped predicate called a token. A token can be defined as a property – described as a first-order logic predicate – with its associated temporal scope $(start, duration, end)$, using flexible interval arithmetic. All the attributes of a token are described as a domain of possible values for this token in the plan context. In order to be part of the plan, a token needs to be associated with one of the plan timelines. A timeline is a sequence of tokens describing the evolution of a state variable. Concurrency between timelines and therefore between tokens on separate timelines, is the basis for concurrent state variable evolution. Tokens are causally linked by rules in a domain model that describe temporal relations and/or causality links between tokens [22,42] (see Fig. 4). Finally, a token can be marked optionally as either being a Fact or Goal; while a Fact requires no justification, a Goal will need not only to be inserted in the plan but necessarily have a causal chain connecting it to one or more Facts. Like T-REX, the underlying planner for EUROPtus is EUROPA [26,22]; while EUROPtus's concepts are not tied to this specific planner, our implementation relies heavily on both its flexible and rich representation, as also the basic principle in the way search is implemented in plan-space planners [30,23].

The planner works by continuously repairing flaws in a plan until no more flaws are present. Prior to execution, all partial plans must not have any flaws. Typically we deal with two types of flaws [1]:

- Open condition: A token is not yet associated with a timeline. It can be resolved by either inserting the token into a specific timeline or merging it with a compatible token;
- Threat: Once a token has been inserted it may impact other tokens indirectly through possible overlapping requirements. A timeline by definition, enforces a strict sequence of tokens with no concurrency within the timeline. The solver then needs to enforce a scheduling constraint on those potentially conflicting tokens, so they cannot overlap; for example, by enforcing that one should occur before the other.

The solver resolves these flaws until either it reaches an inconsistency i.e., a situation where constraints of the plan cannot be satisfied), in which case the planner will backtrack to explore alternate solutions; or a consistent solution is found and the plan presents no further flaws generating a valid solution.

Note that while the plan might be complete, it does not have to commit to the value of its variables. For example, the start time of a token can be left to be the interval $[1, 10]$ as long as it does not present a threat to the partial-plan. This leaves the decision of the start time to the executive, which is critical for operating in uncertain real-world environments.

5.2 The Cetacean Tracking Domain

We can briefly describe the cetacean tracking domain primarily as a way to motivate the use of EUROPtus and thus, as a way to ground the methodology in this section and experiments in Section 6.

In this domain we consider two types of assets: a UAV that is externally operated (with human-in-the-loop waypoint-based control) and for which the only observable outcome is a cetacean sighting; and AUVs (two in our case) which have in situ goal-oriented commanding using T-REX and can receive survey objectives via a timeline goal. The UAV launch and recovery operations involve substantial human involvement and its operation currently requires close monitoring. Consequently, the basis of interaction with EUROPtus was simplified as follows:

- EUROPtus can request a new deployment through NEPTUS directed at the UAV operator;
- the maximum UAV operation time is assumed to be 30 minutes, given available battery life;
- recharging a UAV battery on shore/ship takes approximately 15 minutes [2];
- a cetacean position update is expected to be the observed outcome of a successful UAV survey and, when available, generates a goal within EUROPtus which is instantiated on a timeline.

As a result, EUROPtus approximately models UAV operational constraints and its interactions are indirect as it only sends an IMC message to the UAV operators console to trigger the operators response; and its observations similarly are driven by a human operator's event when s/he manually identifies and "marks" a cetacean position on NEPTUS.

---

[1] There can be additional flaw types depending on the solver being used; but these are the minimum needed to converge to a solution.

[2] Both the launch/recovery operation and charging times where approximate, yet reasonable estimates.

The EUROPtus AUV model was based on the following:

- a timeline representing the vehicle position updated whenever a position update was received and placed in the timeline according to its observation timestamp;
- A set of possible surveys both parameterized with their scale in meters (representing the outer box surrounding the survey), its centroid (represented by a latitude and longitude) and its orientation (a rotation angle) [12];
- The high level operational state of the vehicle being either Inactive, Operating and Survey, the latter which takes as an argument a fully instantiated survey as above.

Typically within EUROPtus the AUV's overall state cycles between Inactive, Operating and Survey while executing a survey. The duration of Operating depends on the scale of the survey and the distance from the survey start point (which should be the last position observed when the vehicle was Inactive and on the surface). Survey being a goal state, its duration is very short (equivalent to 1 sec on the embedded T-REX) since it is a feedback confirming the successful completion of the survey request. This timeline – along with the AUV position – effectively produced by T-REX onboard, were the means of interaction between the AUV and EUROPtus on shore/ship. Time-stamped messages from the AUVs, as noted, come with significant time delays. This means that the executive should not only be able to dispatch the actions in a timely manner, but also to integrate observations from the past and, when required by the model, delay the execution of a specific action until its conditions are effectively observed.

5.3 Planning and Execution with Asynchronism

Typically in fully autonomous systems such as T-REX, planning and execution are intertwined. Both manipulate the same plan representation and plan execution is required to occur at every clock cycle. This influences in turn, the outcome of the planning process while ensuring that any plan produced by the planning agent (EUROPtus in our case) is taking into account world state evolution. And so, it assumes that the agent is within a synchronous and fully observable world. Such design, while appropriate for embedded systems, is incompatible with the oceanographic domain where operations are asynchronous and where coordination on launch and recovery operations can be complex. Furthermore, in our case, observations from AUVs arrive sporadically, and observability is limited by the availability of an acoustic channel to the AUVs.

For EUROPtus, computing power is not a critical issue, but communication limitations and observational updates need to account for the fact that world evolution will often be observed later than occurrence, if at all. Execution can still be integrated into a deliberation process as planning and we go a step further in considering that execution is a part of planning [23].

5.4 Tracking Execution within Deliberation

Execution feedback is integrated into the the deliberation process by taking full advantage of the way the planner works. Specifically, the planner will stop searching as soon as no more flaws are found; the introduction of a new token (including Facts) in the plan, creates new flaws in the plan.

Let's consider that the planner has no more flaws and that the next command to be executed has been already dispatched. Because of communication disruption, we receive feedback from AUV_a that indicates that its state changed from Inactive to Operating an hour ago. Our approach is then to create the Fact token Operating for AUV_a starting at the time corresponding to an hour ago. This token is added to the plan generating a new Open condition flaw that the solver needs to resolve.

The resolution can be either as simple as a merge, if the token reflects exactly what was planned, or it is conflicting with the partial plan requiring in turn, the planner to backtrack. The insertion of such a token as a Fact however, is akin to plan recognition in the context of the currently maintained plan. As the planner operates continuously keeping its search state alive, this recognition can impact the search by forcing the planner to backtrack over past decisions until it finds an alternative new solution, given the injected Fact. As long as we assume that the decisions impacted by the new observation are not close to the root of the search tree, it allows for a plan resolution in few steps without an adverse impact to performance. Further, such an assumption is reasonable as often past observations arrive in relative chronological order rarely impacting the distant past in plan history. These steps also highlight how execution tracking can be a pure deliberative task within the same planning engine.

The remaining problem is to decide which part of the plan is ready to be executed and sent to NEPTUS. Many actions in a partial plan can be conditioned by the need to observe a situation. Actions are specific tokens introduced in EUROPA, that instantiate a causal relationship between their condition and effect tokens. It reflects the classical approach to describe a plan domain (similar to STRIPS [21] but substantially more expressive) and allows its use as additional semantics for our need. In EUROPA, an action is a special kind of token with temporal relations expressed either as conditions necessary for its execution, or expected effects of this actions [46].

To command the AUVs for a survey we needed EUROPtus to observe the following:

- Both AUVs are ready and in the Inactive state;
- We have a cetacean position update that is at most 30 minutes old – this heuristic was imposed to ensure "freshness" of cetacean tracks.

When deciding on dispatching a partial plan for an AUV survey, EUROPtus does an analysis of the causality structure of an action with its tokens. We do so by introducing the notion of a Justified token as follows: a token is Justified if either it is a Fact token, the condition of a Justified action or it is an action for which all the effects are Justified.

In a complete plan, an action in the plan can be dispatched for execution when all its conditions are Justified and its start time interval contains the current time. While action justification is reasonable, we need to ensure that we do not dispatch the action before its valid start time. However, we could potentially be in a situation where an action does not have all its justified conditions due to one or more missing messages and yet its start time is before the current time (i.e., it should have been dispatched for execution in the past). Dispatch is then postponed and we address this as a Pending action flaw, which applies to any action of the plan that could start at the current time but does not have all of its conditions Justified nor is it Justified itself. Its default resolution is to restrict the start time to be postponed. Consequently the planner needs to postpone dispatching this action until either new observations justify the action or the start time can no longer be pushed; the latter triggers a backtrack for an alternate solution.

Any pending action that has all of its conditions Justified is dispatched to NEPTUS for execution, which eventually will receive the observation of its completion (from the vehicle) and report it to EUROPtus. This results in a control loop that is managed as a pure continuous deliberation process governed by the principles just described.

The position update then comes into play for the AUV operations model. As the AUVs are driven by an embedded T-REX agent, EUROPtus could be further extended to directly leverage such a positional update for "direct" control of the vehicles. Yet the limitation in terms of communication had to be taken into account. Our AUVs can communicate with the ship only if they are at the surface and either in WI-FI range (which might also be not desirable as the ship can present a threat if operated too close to the vehicle) or is at the surface long enough to initiate a satellite connection. When the vehicle is underwater, it was assumed that there is no means to communicate [1].

Fig. 9 The location of the cetacean tracking experiment off of the island of Pico in the Azores in the mid-Atlantic.

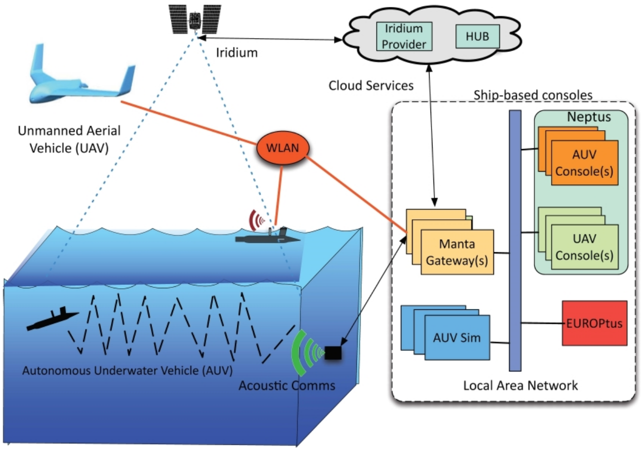


Fig. 10 Architectural block diagram of coordinated observations of a UAV with two AUVs in the water-column in July 2015 off of the Azores.

6 Experiment Domain and Setup

In 2015, a key objective of our annual inter-disciplinary field experiment was to perform co-temporal surveys for synoptic observations on the habitat of sperm whales. Cetacean tracking experiments were conducted in open waters south of Pico Island in the Azores (Fig. 9).

---

[1] Note: acoustic communication was only used for vehicle telemetry.

(a) Launching the LAUV-Xplore-1 off of a pier in the island of Horta for testing. Inset image shows the nose cone of the AUV with an experimental miniature holographic camera.

(b) A test flight of the X8 Skywalker UAV prior to offshore operations.

Fig. 11 Tests of platforms on-shore prior to the cruise.

Fig. 12 Video feed captured by X8-03 UAV with infrared imagery on the right.

Operations were conducted on board the Portuguese Navy research vessel NRP Almirante Gago Coutinho[1], which was used for launching and recovering AUVs and UAVs from the aft deck (Fig. 1). All operators and pilots were on board the vessel which had multiple antennas which provided 802.11 Wi-Fi coverage via Ubiquiti radios. The antennas were connected to multiple “Manta” communications gateways that relayed all data between unmanned vehicles and NEPTUS consoles. Inside the ship, a local area network was created to which all systems (mantas and consoles) were connected to. This network setup allowed operators to be aware of the AUV and UAV operations simultaneously (Fig. 10).

Two upper water-column Light AUVs [50] (LAUV-Xplore-1 & LAUV-Xplore-2) were used, both equipped with WHOI acoustic modems and Iridium SBD modems. LAUV-Xplore-1 was additionally equipped with an RBR XR620 CTD (Conductivity, temperature and Density) and an experimental holographic particle imaging system developed at Plymouth University (Fig. 11(a)) [25]. LAUV-Xplore-2, which is a similar vehicle in size and endurance, included only a Turner Designs Cyclops-7 wet-probe with a fluorometer. Both these vehicles can travel at a nominal speed of 2 knots ($\sim$1 m/s) for up to 24 hours. Moreover, while deployed in the open sea, they can receive new commands and report their state over Iridium SBD messages (up to 250 bytes using RockBlock [2]).

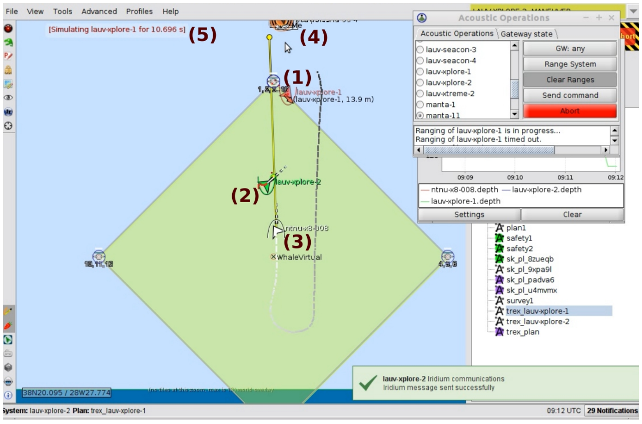
Additionally, there were multiple Skywalker X8 UAVs, seen in Fig. 11(b), at our disposal. These low-cost vehicles are adaptation of COTS (Commercial Off-The-Shelf) units which LSTS has augmented with specific components including System-On-a-Chip (SOC) CPUs, communications hardware and specific sensor payload. These frames, normally used for RC-flying, are inexpensive and made a resilient styrofoam which can be be further customized if needed. The system has a maximum flight time of around 45 minutes at an average cruise speed of $\sim$17 m/s giving it a useful (round trip) range of almost 25km which are typically not utilized in our kind of scenario. For this experiment, the UAVs carried both video cameras providing real-time feed to operators and still cameras taking periodic imagery of the water at high resolution. While some cameras captured visible light, others sampled in the LWIR spectrum (Fig. 12). We also deployed a new light-weight hyper-spectral imager on a separate UAV for purposes of testing; the results of that test are not part of this work.

As noted earlier, the key scientific objective of this field experiment was to characterize the upper water column properties within sperm whale foraging areas to understand what drives their foraging behavior. Our scenario called for either an experienced human observer on a separate boat, or airborne UAVs with real-time imagers, to be able determine target cetaceans on the surface. Researchers on board the ship or small vessels acted as spotters.

---

[1] http://goo.gl/hEVmFU

[2] http://www.rock7mobile.com/products-iridium-sbd


[Simulating lauv-xplore-1 for 10.696 s]

Ranging of lauv-xplore-1 is in progress...
Ranging of lauv-xplore-1 timed out.

lauv-xplore-2 Iridium communications
Iridium message sent successfully

System: lauv-xplore-2 Plan: trex_lauv-xplore-1

09:12 UTC 29 Notifications

Fig. 13 A NEPTUS console snapshot taken during the experiment showing a submerged AUV (1), an AUV at the surface (2), a UAV (3), the ship position (4) and the simulation time for the AUV (5).

As individual sperm whales surfaced, a reference point was determined either from imagery or from the spotter, which then was communicated to the operators on the vessel. If AUVs were not already in the water, they were launched to start autonomous execution. Repetitive AUV surveys around this targeted spot were to commence, using two AUVs to measure spatial variations in the water column.

Both T-REX-enabled AUVs were tasked by EUROPtus to synthesize yoyo based survey patterns [12] in two concentric square patterns of $400X400$ meters^2 and $800X800$ meters^2 diving to a depth of 50 meters. The vehicles also surfaced on the corners of the squares in order to correct its localization with a GPS fix and (tentatively) report its progress via Iridium. The larger pattern was used to sample outside the sperm whale movement path, as a measure of understanding the variability in the upper water-column. The smaller survey was expected to take 25 minutes, while the larger 50 minutes, at about 2.5 knots speed over ground for the vehicles. Consequently, the inner surveys were executed twice to provide a dense coverage co-temporal to the single outer survey, (Fig. 14(a)). At the end of the survey, the next target spot was to be determined either via the spotter or a UAV, and the vehicles re-tasked by EUROPtus.

One extended objective was to attempt not only co-temporal AUV surveys, but to coordinate the survey of the sampling area with UAV overflights with Far-IR and the hyper-spectral imagers. In doing so, it was thought, we could obtain additional data of the ocean surface to be merged and subsequently studied to understand the bio-geochemical composition in the light of any effluents from the passing whales. This was successfully achieved by sharing of mission plans between consoles and by coordinating the execution of plans via EUROPtus . Vehicles operating in the same area can be seen in Fig. 14(b), one airborne, one at the surface and one AUV underwater (connected acoustically).

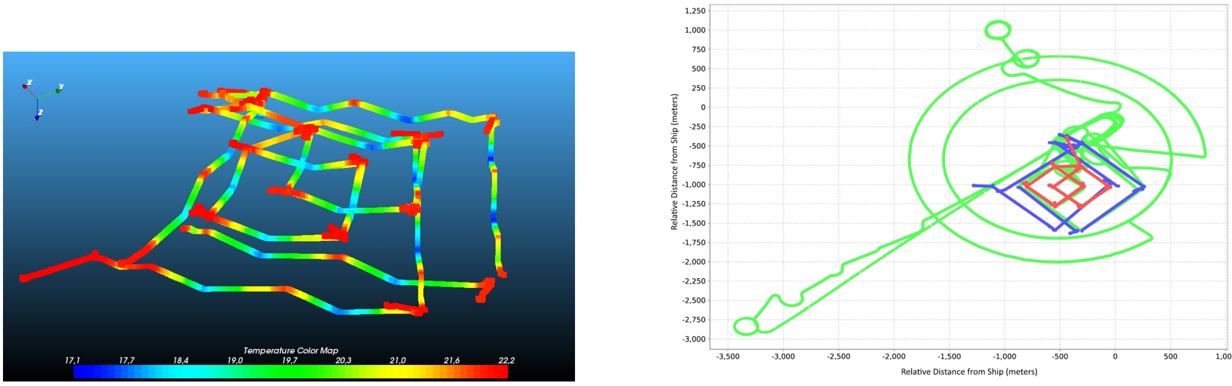
At the same time, multiple NEPTUS consoles were receiving and controlling the different vehicles from the ship, as seen in Fig. 13. One of the consoles was explicitly used for AUV operations and had the EUROPtus plug-in activated while also synchronizing all data received via Iridium and Wi-Fi from the two AUVs connected to their respective shipboard simulators. The data from the simulators was, however, available to all consoles, providing situation awareness on the position of the vehicles underwater, even for the UAV operators.

At the UAV consoles, operators could receive imagery in real-time and a plug-in was used to geo-locate points in the video stream. Whenever a frame of interest was detected in the video feed, the operator could pause the feed and click a point in the image. After attaching a name to the clicked point, these names, together with their coordinates were disseminated to the remaining consoles, so that identified cetaceans or other points of interest were available in all operator consoles. If the disseminated position was a cetacean to be tracked, the AUV operator could select it as a valid target which resulted in forwarding this location to EUROPtus. This additional step was added so that human operators can select preferred targets. Preference would generally be given to targets closer to current AUV locations.

In EUROPtus we tied the AUVs operational model with that of the UAV to ensure coordination; UAVs if not in the air were launched from the aft deck, to be surveying the same area overhead. For this experiment, launch was based primarily on readiness of the vehicle.

A cetacean position was considered “fresh” by EUROPtus if and only if it was received at most 30 minutes before the vehicles were to start a new survey. The surveys' relative scales between the two AUVs was identified in order to have both vehicle finishing the survey approximately at the same time for ease of recovery or relocation. All of this could be determined by EUROPtus including, for example, that having a “fresh” whale position would require an UAV launch and for a valid survey both AUVs are required to be available and sufficiently close to the survey area. While this domain appears to be straight-forward, for a human operator it is quite difficult to properly follow the execution of multiple robots and still make ahead-of-time decisions of what the vehicles should be doing next. EUROPtus, on the other hand, tracks execution automatically and deals with hard to predict and often very large latency of communications.

Fig. 15 shows EUROPtus deliberation steps and depth along the first hour of the July 19th mission. A single deliberation 'step' is associated with the system making a single choice of a binding for a variable in order to satisfy a constraint, which ideally moves the system towards a final plan (or outcome). This fig-



(a) Two AUV trajectories with CTD profiles (red: warmer surface waters, blue: colder deeper waters) executing two consecutive co-temporal surveys around a targeted cetacean position.

(b) UAV trajectory (green) superimposed on AUV tracklines (red and blue) from 14(a). UAV tracks show several search trajectories and long-distance tests.

Fig. 14 Coordinated observations of a UAV with two AUVs in the water-column on July 19th, 2015.

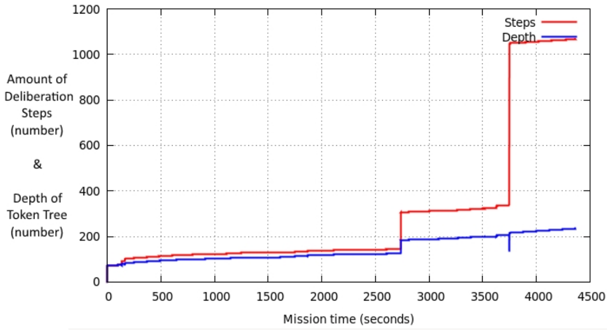


Fig. 15 Number of deliberation steps in EUROPtus and its search tree depth for a subset of July 19th operations. EUROPtus was running on a Linux Virtual Machine on a 2.8GHz Intel Core i7.

ure shows the number of steps increasing monotonically reflecting the integration of new observations as they were received from NEPTUS. The depth grows slower showing that backtracking occurs during execution. Except for the small downward spike, the depth tends to either remain flat or climb, indicating that in general backtracking was only impacted by chronologically recent observations and hence the planner recovered gracefully. The downward spike was due to erroneous timestamps on some AUV location messages before they reached EUROPtus. These could have been caused because of improper clock settings. Despite the large jump in steps ($\approx$ 700) the planner recovered in less than 3 seconds.

In tandem with all these deployments, the experimental holographic particle imaging system recorded high-resolution images of suspended particles (25μm to 5mm size) within a 2ml sample volume at 1 Hz. As visible in Fig. 11(a) the system was mounted on the nosecone of LAUV-Xplore1 to sample the particles in an undisturbed state. Digital reconstruction of the holograms provides focused images of the particles allowing them to be categorized, sized and counted. Fig. 16 shows example images of particles recorded during the experiment, comprising phytoplankton, zooplankton as well as detritus (marine snow). The distributions of the relative abundance of each of these particle categories will be related to the physical and biological properties of the water column (temperature, salinity, chlorophyll fluorescence). Such data presents an interesting opportunity, and serves to highlight the capacity such coordinated autonomous systems have to collect co-temporal and diverse data, which can then be cross-calibrated if needed.

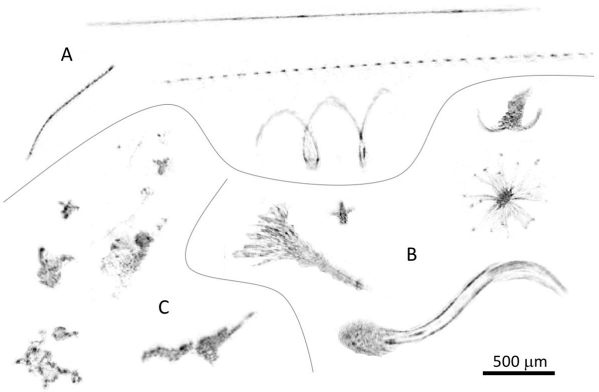


Fig. 16 Example particles imaged by the miniature holographic sensor on-board LAUV-Xplore-1. A) Phytoplankton – diatom chains. B) Zooplankton – crustaceans and fish larvae. C) Detritus – flocculated marine snow

7 Discussion and Conclusions

We present an infrastructure to supervise coordinated operations of multiple heterogeneous assets, while dealing with unpredictability of the environment and sporadic communications in the open sea. Many of these tools are the result of years of work iterating towards a balance between autonomous decision making and mission execution, as well as feedback for operator supervision.

The operation of different vehicles was carried out by a mix of operators and automated planners on board the vehicles and at the control station. The use of the LSTS Toolchain and its unified IMC protocol allowed coordination of systems with very different capabilities over heterogeneous links. The DUNE software portability and its small footprint allowed it to run across communication gateways, vehicles and shore-side simulators. NEPTUS was used simultaneously by different operators targeting different mission responsibilities. Despite using different user interfaces, relevant data could be shared between the consoles over Wi-Fi, which allowed for an effective localization and tracking flow having human operators in the loop.

EUROPtus was designed and first deployed in the context of this operation. Its ability to handle observations with temporal delays and to dispatch commands only when all their conditions are observed proved to be effective. Operators consequently, could focus on high level operational concerns and rely on NEPTUS to maintain situational awareness for understanding the current status of the mission.

The recent addition of EUROPtus relieves operators of the intricacies of synchronization between assets and focuses more on higher level issues such as the scientific goals and the safety of the assets. To address the latter, NEPTUS was extended to not only dispatch commands from EUROPtus to the assets but also to intercept them, in order to give an overview of expected behavior to the users. This ability to maintain situational awareness, proved to be critical for our at-sea operations.

Additionally, EUROPtus allowed for the reduction in logistics outside of the operational team. The simplification came in the streamlining of spotting tasks for cetaceans since systems coordinated by EUROPtus were also tasked to fulfill that role. This allows for added scalability when both the operation area and mission objectives increase.

Further work remains. Among those is the way we resolve Pending action flaws. The only way the system can resolve them currently, is by postponing the action until it has all of its conditions Justified. An interesting extension would be for the planner to actively enrich the plan by proactive search (for example by polling) for such a justification, perhaps even asking for operator input. This 'inquisitive' approach would then prevent the system from replanning just because there was no feedback before a certain deadline.

Another aspect we would like to explore is allowing EUROPtus to forget part of its search when it no longer impacts the plan. A novelty of EUROPtus, when compared to classical approaches, is that it keeps all of its search history. One benefit is that EUROPtus can revisit the past and restore a partial plan fragment in the light of a delayed observation or justification. A side effect, is that the receding planning horizon can result in computational search being slower as time advances. A solution to this potential issue would be to prune nodes from the search tree as they are justified by observations. This would require replacing the chronological backtracking search, allowing the system to run for a long periods without performance impacts.

Acknowledgements Johansen was supported in part by the Research Council of Norway through the Centers of Excellence funding scheme, grant number 223254 – NTNU AMOS, and by grant number 235348. Rajan and Py were supported by the US Office of Naval Research, ONR Grant # N00014-14-1-0536. Silva was supported by FEDER and COMPETE funds and by FCT grant # IF/00943/2013. Silva was also supported by FRCT from the Government of the Açores. Authors are grateful to the Cetacean Ecology Research Group including Pablo Chellavard Navarro, Rui Prieto, Cláudia Oliveira, Marta Tobena, and skipper Renato Bettencourt, and to NTNU's UAV team, including Lars Semb, Krzysztof Cisek, Frederik Leira and João Fortuna. This work was partially supported by the SUNRISE - "Sensing, monitoring and actuating on the UNderwater world through a federated Research InfraStructure Extending the Future Internet", project #611449, funded by the European Union's Seventh Framework Programme for Research and Technological Development - Large scale integrating project (IP). We are grateful to the captain and crew of NRP Gago Coutinho. Finally, we are grateful to the entire LSTS team that participated in the REP-15 exercise in the Açores.

References

1. Ai-Chang, M., Bresina, J., Charest, L., Chase, A., Hsu, J., Jonsson, A., Kanefsky, B., Morris, P., Rajan, K., Yglesias, J., Chafin, B., Dias, W., Maldague, P.: MAPGEN: Mixed Initiative Planning and Scheduling for the Mars'03 MER Mission. IEEE Intelligent Systems 19(1) (2004)
2. Arrichiello, F., Das, J., Heidarsson, H.K., Sukhatme, G.S., Chiaverini, S.: Experiments in autonomous navigation with an under-actuated surface vessel via the Null-Space based Behavioral control. In: Conference on Advanced Intelligent Mechatronics, pp. 362–367 (2009)
3. Benjamin, M.R., Schmidt, H., Newman, P.M., Leonard, J.J.: Nested autonomy for unmanned marine vehicles with MOOS-IvP. Journal of Field Robotics 27(6), 834–875 (2010)
4. Bernard, D.E., Dorais, G.A., Fry, C., Jr., E.G., Kanfesky, B., Kurien, J., Millar, B., Muscettola, N., Nayak, P., Pell, B., Rajan, K., Rouquette, N., Smith, B., Williams, B.: Design of the Remote Agent experiment for spacecraft autonomy. In: Proceedings of the IEEE Aerospace Conf. Snowmass, CO (1998)
5. Block, B.A., Jonsen, I.D., Jorgensen, S.J., Winship, A.J., Shaffer, S.A., Bograd, S.J., Hazen, E.L., Foley, D.G., Breed, G.A., Harrison, A.L., Ganong, J.E., Swithenbank, A., Castleton, M., Dewar, H., Mate, B.R., Shillinger, G.L., Schaefer, K.M., Benson, S.R., Weise, M.J., Henry, R.W.,

Costa, D.P.: Tracking apex marine predator movements in a dynamic ocean. Nature 475(7354), 86–90 (2011). DOI 10.1038/nature10082. URL http://www.nature.com/doifinder/10.1038/nature10082

6. Bresina, J., Jonsson, A., Morris, P., Rajan, K.: Activity Planning for the Mars Exploration Rovers. In: International Conference on Automated Planning and Scheduling (ICAPS). Monterey, California (2005)
7. Bresina, J., Jonsson, A., Morris, P., Rajan, K.: Mixed-Initiative Planning in MAPGEN: Capabilities and Shortcomings. In: Proceedings of the workshop on Mixed-Initiative Planning and Scheduling, ICAPS. Monterey, California (2005)
8. Brooks, R.: A robust layered control system for a mobile robot. IEEE Journal on Robotics and Automation 2(1), 14–23 (1986)
9. Chrpa, L., Pinto, J., Ribeiro, M., Py, F., Sousa, J., Rajan, K.: On Mixed-Initiative Planning and Control for Autonomous Underwater Vehicles. In: Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on. Hamburg, Germany (2015)
10. Curtin, T.B., Bellingham, J.G., Catipovic, J., Webb, D.: Autonomous oceanographic sampling networks. Oceanography 6(3), 86–94 (1993)
11. Das, J., Maughan, T., McCann, M., Godin, M., O'Reilly, T., Messie, M., Bahr, F., Gomes, K., Py, F., Bellingham, J., Sukhatme, G., Rajan, K.: Towards mixed-initiative, multi-robot field experiments: Design, deployment, and lessons learned. In: Proc. Intelligent Robots and Systems (IROS). San Francisco, California (2011)
12. Das, J., Py, F., Maughan, T., Messie, M., O'Reilly, T., Ryan, J., Sukhatme, G.S., Rajan, K.: Coordinated Sampling of Dynamic Oceanographic Features with AUVs and Drifters. Intnl. J. of Robotics Research 31, 626-646 (2012). April
13. Das, J., Rajan, K., Frolov, S., Ryan, J.P., Py, F., Caron, D.A., Sukhatme, G.S.: Towards Marine Bloom Trajectory Prediction for AUV Mission Planning. In: IEEE International Conference on Robotics and Automation (ICRA). Anchorage, Alaska (2010)
14. Dias, P.S., Fraga, S.L., Gomes, R.M., Goncalves, G.M., Pereira, F.L., Pinto, J., Sousa, J.B.: NEPTUS- a framework to support multiple vehicle operation. In: Proc Oceans MTS/IEEE Conference, pp. 963–968. Brest, France (2005)
15. Dukeman, A., Adams, J.A., Edmondson, J.: Extensible collaborative autonomy using gams. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing, pp. 281–283. ACM (2016)
16. Edmondson, J., Schmidt, D.: Multi-agent distributed adaptive resource allocation (madara). International Journal of Communication Networks and Distributed Systems 5(3), 229–245 (2010)
17. Encarnaçao, P., Pascoal, A.: Combined trajectory tracking and path following: an application to the coordinated control of autonomous marine craft. In: Decision and Control, 2001. Proceedings of the 40th IEEE Conference on, vol. 1, pp. 964–969. IEEE (2001)
18. Faria, M., Pinto, J., Py, F., Fortuna, J., Dias, H., Leira, R.M.F., Johansen, T.A., Sousa, J.B., Rajan, K.: Coordinating UAVs and AUVs for Oceanographic Field Experiments: Challenges and Lessons Learned. In: IEEE International Conference on Robotics and Automation (ICRA). Hong Kong (2014)
19. Fedak, M.: Marine animals as platforms for oceanographic sampling: a “ win / win ” situation for biology and operational oceanography. Memoirs of the National Institute for Polar Research, Special Issue 58, 133–147 (2004)
20. Ferreira, A.S., Pinto, J., Sousa Dias, P., de Sousa, J.B.: The LSTS software toolchain for persistent maritime operations applied through vehicular ad-hoc networks. In: 2017 International Conference on Unmanned Aircraft Systems (ICUAS), pp. 609–616. IEEE (2017)
21. Fikes, R., Nilsson, N.: STRIPS: A new approach to the application of theorem proving to problem solving. Artificial Intelligence 2, 189 – 205 (1971)
22. Frank, J., Jónsson, A.: Constraint-based Attribute and Interval Planning. Constraints 8(4), 339–364 (2003)
23. Ghallab, M., Nau, D., Traverso, P.: Automated Planning and Acting. Cambridge University Press (2016)
24. Gonçalves, G., Gonçalves, R., Ferreira, A.S., de Sousa, J.B., Pinto, J.: Supervisory Control: Optimal Distribution of Workload Among Operators for Mixed Initiative Control of Multiple UAVs. In: Dr. Leo van Breda (ed.) Supervisory Control of Multiple Uninhabited Systems – Methodologies and Enabling Human-Robot Interface Technologies, chap. 10, pp. 153 – 163. STO/NATO (2012)
25. Graham, G.W., Nimmo Smith, W.A.M.: The application of holography to the analysis of size and settling velocity of suspended cohesive sediments. Limnology and Oceanography: Methods 8(1), 1–15 (2010)
26. Jónsson, A., Morris, P., Muscettola, N., Rajan, K., Smith, B.: Planning in Interplanetary Space: Theory and Practice. In: Artificial Intelligence Planning and Scheduling (AIPS) (2000)
27. Lapierre, L., Soetanto, D., Pascoal, A.: Coordinated motion control of marine robots. Proceedings of the 6th IFAC MCMC, Girona, Spain (2003)
28. Leira, F.S., Johansen, T.A., Fossen, T.I.: Automatic detection, classification and tracking of objects in the ocean surface from uavs using a thermal camera. In: 2015 IEEE Aerospace Conference, pp. 1–10 (2015). DOI 10.1109/AERO.2015.7119238
29. Lemai-Chenevier, S., Ingrand, F.: Interleaving Temporal Planning and Execution in Robotics Domains. In: Assoc. for the Advancement of Artificial Intelligence, National Conference (AAAI) (2004)
30. M. Ghallab and D. Nau and P. Traverso: Automated Planning: Theory and Practice. Elsevier Science (2004)
31. Manley, J., Willcox, S.: The wave glider: A persistent platform for ocean science. In: OCEANS 2010 IEEE-Sydney, pp. 1–5. IEEE (2010)
32. Martinez, C., Keener-Chavis, P.: NOAA Ship Okeanos Explorer: Telepresence in the Service of Science, Education and Outreach. In: OCEANS 2006, pp. 1–5. IEEE (2006)
33. Martins, R., de Sousa, J.B.: An extensible networking architecture for autonomous underwater vehicles. In: 2013 OCEANS - San Diego, pp. 1–5 (2013)
34. Martins, R., de Sousa, J.B., Afonso, C.C., Incze, M.: REP10 AUV: Shallow water operations with heterogeneous autonomous vehicles. In: OCEANS 2011 IEEE - Spain, pp. 1–6. IEEE (2011). DOI 10.1109/Oceans-Spain.2011.6003568. URL http://ieeexplore.ieee.org/document/6003568/
35. McGann, C., Py, F., Rajan, K., Thomas, H., Henthorn, R., McEwen, R.: A Deliberative Architecture for AUV Control. In: IEEE International Conference on Robotics and Automation (ICRA). Pasadena (2008)
36. McGillivary, P., Sousa, J., Martins, R., Rajan, K., Leroy, F.: Integrating Autonomous Underwater Vessels, Surface Vessels and Aircraft as Persistent Surveillance Components of Ocean Observing Studies. In: IEEE Autonomous Underwater Vehicles. Southampton, UK (2012)
37. Muscettola, N., Nayak, P., Pell, B., Williams, B.: Remote Agent: To Boldly Go Where No AI System Has Gone Before. Artificial Intelligence 103, 5–48 (1998)
38. Newman, P.: A mission oriented operating suite. Tech. rep., Technical Report OE2007-07. MIT Department of Ocean Engineering (2003)
39. Pinto, J., Faria, M., Fortuna, J., Martins, R., Sousa, J., Queiroz, N., Py, F., Rajan, K.: Chasing Fish: Tracking and control in a autonomous multi-vehicle real-world experiment. In: MTS/IEEE Oceans. San Diego, California (2013)

40. Pinto, J., Martins, P.S.D.R., Fortuna, J., Marques, E., Sousa, J.: The LSTS toolchain for networked vehicle systems. In: MTS/IEEE Oceans, pp. 1–9. IEEE (2013)
41. Pinto, J.Q., Dias, P.S., Gonçalves, R., Gonçalves, G.M., Sousa, J.T.d.F.B., Pereira, F.L., et al.: Neptus a framework to support a mission life cycle. In: Proceedings of the 7th Conference on Manoeuvring and Control of Marine Craft (2006)
42. Py, F., Rajan, K., McGann, C.: A Systematic Agent Framework for Situated Autonomous Systems. In: 9th International Conf. on Autonomous Agents and Multiagent Systems (AAMAS). Toronto, Canada (2010)
43. Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A.Y.: ROS: an open-source robot operating system. In: ICRA workshop on open source software, vol. 3, p. 5. Kobe (2009)
44. Rajan, K., Bernard, D., Dorais, G., Gamble, E., Kanefsky, B., Kurien, J., Millar, W., Muscettola, N., Nayak, P., Rouquette, N., Smith, B., Taylor, W., Tung, Y.: Remote Agent: An Autonomous Control System for the New Millennium. In: Proceedings Prestigious Applications of Intelligent Systems, European Conf. on AI. Berlin (2000)
45. Rajan, K., Py, F.: T-REX: Partitioned Inference for AUV Mission Control. In: G.N. Roberts, R. Sutton (eds.) Further Advances in Unmanned Marine Vehicles, pp. 171–199. The Institution of Engineering and Technology (IET) (2012)
46. Rajan, K., Py, F., Berreiro, J.: Towards Deliberative Control in Marine Robotics. In: M. Seto (ed.) Autonomy in Marine Robots, pp. 91–175. Springer Verlag (2012)
47. Ramp, S.R., Davis, R.E., Leonard, N.E., Shulman, I., Chao, Y., Robinson, A., Marsden, J., Lermusiaux, P., Fratantoni, D., Paduan, J.D., et al.: Preparing to predict: the second autonomous ocean sampling network (AOSN-II) experiment in the Monterey Bay. Deep Sea Research Part II: Topical Studies in Oceanography 56(3), 68–86 (2009)
48. Roberts, J.J., Best, B.D., Mannocci, L., Fujioka, E., Halpin, P.N., Palka, D.L., Garrison, L.P., Mullin, K.D., Cole, T.V.N., Khan, C.B., McLellan, W.A., Pabst, D.A., Lockhart, G.G.: Habitat-based cetacean density models for the U.S. Atlantic and Gulf of Mexico. Scientific Reports 6(1), 22,615 (2016). DOI 10.1038/srep22615. URL http://www.nature.com/articles/srep22615
49. Shkurti, F., Xu, A., Meghjani, M., Higuera, J.C.G., Girdhar, Y., Giguere, P., Dey, B.B., Li, J., Kalmbach, A., Prahacs, C., et al.: Multi-domain monitoring of marine environments using a heterogeneous robot team. In: Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on, pp. 1747–1753. IEEE (2012)
50. Sousa, A., Madureira, L., Coelho, J., Pinto, J., Pereira, J., Sousa, J., Dias, P.: LAUV: The man-portable autonomous underwater vehicle. In: Navigation, Guidance and Control of Underwater Vehicles, pp. 268–274 (2012)
51. de Sousa, J.B., Pereira, J., Alves, J., Galocha, M., Pereira, B., Lourenço, P., Portuguesa, M.: Experiments in multi-vehicle operations: The Rapid Environmental Picture Atlantic exercise 2014. In: OCEANS 2015 - Genova, pp. 1–7. IEEE (2015). DOI 10.1109/OCEANS-Genova.2015.7271761. URL http://ieeexplore.ieee.org/document/7271761/
52. Sousa, L.L., López-Castejón, F., Gilabert, J., Relvas, P., Couto, A., Queiroz, N., Caldas, R., Dias, P.S., Dias, H., Faria, M., Ferreira, F., Ferreira, A.S., J.Fortuna, Gomes, R.J., Loureiro, B., Martins, R., Madureira, L., Neiva, J., Oliveira, M., Pereira, J., J.Pinto, Py, F., Queiroz, H., Silva, D., Sujit, P.B., Zolich, A., Johansen, T.A., Sousa, J., Rajan, K.: Integrated monitoring of mola mola behaviour in space and time. PLOS one (2016). DOI http://dx.doi.org/10.1371/journal.pone.0160404
53. Stoker, C.R., Burch, D., Hine, B.P., Barry, J.: Antarctic undersea exploration using a robotic submarine with a telepresence user interface. IEEE Expert 10(6), 14–23 (1995)
54. Webb, D.C., Simonetti, P.J., Jones, C.P.: Slocum: An underwater glider propelled by environmental energy. IEEE Journal of oceanic engineering 26(4), 447–452 (2001)

António Sérgio Ferreira is a Ph.D student and holds a Master's degree in Informatics and Computing Engineering from University of Porto. His research interests include unmanned vehicle systems, communications and coordination of multiple dynamic systems, hybrid systems, systems engineering, and management architectures for multi-vehicle systems. Since 2011 he has been part of the Underwater Systems and Technology Laboratory (LSTS) where his work focused on further developing the Neptus Command and Control Infrastructure for UAS as well as being a UAS field operator.

Maria Costa is a Robotics Engineer in a robotics research laboratory (LSTS). She is a main software developer (onboard software) and is also a part of both the AUV and UAS field operational teams (where she is the main UAS Operator: fixed-wing and multirotor vehicles). During the last four years she gained significant experience in field operations, mostly maritime environments (operating from ship) in cooperation with the Portuguese Navy. She also had the opportunity to participate in several national and European projects (Seagull, PITVANT, Safeport, NetOcean, NETMAR, Fire-RS, URready4OS, E-URready4OS) and annual big-scale exercises (REP14, REP15, REP16, REP17). Maria Costa holds an Electrotechnical and Computer Engineering degree and a Masters in Autonomous Systems from School of Engineering Polytechnic of Porto (ISEP). Her master thesis is entitled "Assisted Maneuvers for Underactuated ROVs". Prior to college education, she also graduated top of her class as a EU Level 3 Technician in Industrial Electronics and Automation. Throughout her college education she worked as an Electronics Technician and also integrated the Ise-Porto robotic football team (participating in a World

Cup Middle Size League competition (RoboCup 2009) and in a couple of National Middle Size League competitions - Robótica 2009 and Robótica 2010).

Frédéric Py holds a doctorate degree in Robotics and Artificial Intelligence from University Paul Sabatier, Toulouse. He has pioneered the use of advanced statistical methods using Clustering algorithms for detecting dynamic events in the water-column in the ocean and has lead the development and design of the T-REX software agent for high level planning and execution along with its application on scientific operations. Moreover, he has worked on automated model learning techniques for in situ analysis of physical oceanographic processes (such as Intermediate Nepheloid Layer, plume, ...) He has collaborated with the LSTS team during the last 2 REP projects, where we worked together integrating the T-REX architecture which he developed while working at MBARI and still supports up to this day.

José Pinto was born in Matosinhos, Portugal, in 1982. He received the B.S. degree in computer science from Porto University, Porto, Portugal, in 2005. He joined the Underwater Systems and Technology Laboratory (LSTS) in 2004 still as an undergraduate student when he started the developments leading to the creation of the Neptus Command and Control Infrastructure as well as other parts of the LSTS Software Toolchain. During his stay at LSTS he has been working towards the autonomous operation of water-column AUVs from satellite communications to underlying control layers, cloud services and user interfaces. Since 2006 he is an Assistant Professor at Porto University, teaching classes on Programming and Embedded Computers. Currently he is finishing his PhD (ABD) on Coordination Infrastructures for Networked Vehicle Systems, under the supervision of João Borges de Sousa.

Mónica A. Silva graduated in Biology from the University of Lisbon in 1996 and received my PhD degree from the University of St. Andrews (Scotland) in 2007. From the time of graduation until enrolling in a PhD, she developed research on marine mammal biology and ecology and Marine Protected Area design at the Nature Conservation Institute (Lisbon), Ministry of the Environment; worked as scientific consultant for the World Conservation Union in Guinea-Bissau, developing a Conservation Plan for the West African manatee; and was temporarily assigned (3 years) to the Department of Oceanography and Fisheries/University of the Azores (DOP/UAz), to work as a Research Assistant in the scope of an EC LIFE-Nature Project. She started a post-doc in 2007 at the Centre of IMAR of the University of the Azores (IMAR/UAz) and at the Woods Hole Oceanographic Institution (WHOI) (USA). In 2013 she was awarded a 5-year contract as Senior Research Associate (Investigador Auxiliar under FCT Investigator Programme) at IMAR.

Alex is an Associate Professor (Reader) in Marine Physics at Plymouth University, and is part of the Marine Physics Research Group (MPRG). His research interests are focussed around suspended particle dynamics, both in terms of how particles (plankton, sediment, bubbles, droplets) behave in a natural, turbulent environment, and also in the use of particles as tracers of turbulent flows. He has developed several in situ optical systems to directly image particles for measurement and identification, and visualise 3D turbulence structure. He's member of the Challenger Society for Marine Science, of the American Geophysical Union and fellow of the Higher Education Academy.

Tor Arne Johansen received the MSc degree in 1989 and the PhD degree in 1994, both in electrical and computer engineering, from the Norwegian University of Science and Technology, Trondheim, Norway. From 1995 to 1997, he worked at SINTEF as a researcher before he was appointed Associated Professor at the Norwegian University of Science and Technology in Trondheim in 1997 and Professor in 2001. He has published several hundred articles in the areas of control, estimation and optimization with applications in the marine, aerospace, automotive, biomedical and process industries. In 2002 Johansen co-founded the company Marine Cybernetics AS where he was Vice President until 2008. Prof. Johansen received the 2006 Arch T. Colwell Merit Award of the SAE, and is currently a principal researcher within the Center of Excellence on Autonomous Marine Operations and Systems (NTNU-AMOS) and director of the Unmanned Aerial Vehicle Laboratory at NTNU.

João Borges de Sousa is with the Electrical and Computer Engineering Department of the Faculty of Engineering from Porto University and he is also the director of LSTS. His research interests include unmanned vehicle systems, control and coordination of multiple dynamic systems, hybrid systems, systems engineering, and control architectures for multi-vehicle systems. Since 1997 he has been leading the design, implementation and deployment of advanced unmanned vehicle systems in projects funded by the Portuguese Foundation for Science and Technology, Portuguese Innovation Agency, Luso-American Foundation, NATO, Office of Naval Research and DARPA. He had several Visiting Scholar appointments at the University of California at Berkeley since 1997. In the last 20 years he has been involved in fostering and growing a world-wide research community in this field with yearly conferences and workshops in the areas of Hybrid Systems, Intelligent Control, Networked Vehicle Systems and Multi-Agent Systems. He has been lecturing and delivering seminars on networked robotics in universities in the United States, Europe, Asia and Africa. In 2006 he received the national BES Innovation National Award for the design of the Light Autonomous Underwater Vehicle. In 2008 he received an outstanding teaching award from Porto University. He was the chair of several IEEE and IFAC conferences and workshops. He has authored more than 300 publications.

Kanna holds an International Chair adjunct professorship at NTNU's Dept. of Engineering Cybernetics and affiliated to the Center for Autonomous Marine Operations and Systems. His background is in Artificial Intelligence Planning and Scheduling and has applied his work in the domains of spaceflight while at NASA and marine robotics while at MBARI, all in California. He is also a Visiting Professor at the Faculty of Engineering, Univ. of Porto's Underwater Systems and Technology Laboratory.