# Coordinating UAVs and AUVs for Oceanographic Field Experiments: Challenges and Lessons Learned

Margarida Faria[1], José Pinto[1], Frédéric Py[2], João Fortuna[1], Hugo Dias[1], Ricardo Martins[1]
Frederik Leira[3], Tor Arne Johansen[3], João Sousa[1] and Kanna Rajan[2]

***Abstract*— Obtaining synoptic observations of dynamic ocean phenomena such as fronts, eddies, oxygen minimum zones and blooms has been challenging primarily due to the large spatial scales involved. Traditional methods of observation with manned ships are expensive and, unless the vessel can survey at high-speed, unrealistic. Autonomous underwater vehicles (AUVs) are robotic platforms that have been making steady gains in sampling capabilities and impacting oceanographic observations especially in coastal areas. However, their reach is still limited by operating constraints related to their energy sources. Unmanned aerial vehicles (UAVs) recently introduced in coastal and polar oceanographic experiments have added to the mix in observation strategy and methods. They offer a tantalizing opportunity to bridge such scales in operational oceanography by coordinating with AUVs in the water-column to get in-situ measurements. In this paper, we articulate the principal challenges in operating UAVs with AUVs making synoptic observations for such targeted water-column sampling. We do so in the context of autonomous control and operation for *networked* robotics and describe novel experiments while articulating the key challenges and lessons learned.**

Fig. 1. Conducting UAV and AUV operations from the Portuguese Navy support vessel NRP *Bacamarte*, Sesimbra, Portugal, July 2013.

## I. INTRODUCTION

Sampling of the coastal ocean is substantially challenging, even at small observation scales. The state of the art entails deploying sensors distributed over buoys, unmanned marine vehicles, manned vehicles and adapting the sampling strategy to the observations. Adaptation is targeted at adjusting the spatial and temporal resolutions to the properties of the sampled field. This is not a trivial task. First, observations need to be communicated over communication challenged environments. Second, data once obtained requires assimilation that is computationally challenging as even prediction models do not fully capture domain knowledge from an expert. Third different sensors and vehicles may have different command and data interfaces making interoperability quite difficult. And finally, state of the art unmanned vehicles while automated, lack inferential reasoning capabilities, thus requiring human guidance for effective adaptation. These are some of the reasons why there is a significant gap between operational deployments and simulation studies.

Autonomous platforms such as powered autonomous underwater vehicles (AUVs) or slower moving gliders have extended the reach of traditional ship-board methods for obtaining oceanographic measurements. As a consequence, scientists are able to characterize a wider swath of a survey area in less time. Yet, such measurements are not *synoptic* since they do not match the mesoscale ($> 50km^2$) observation capability necessary to understand the bio-geochemistry and organism transport important for science, one example are coastal ecological studies. Coupled with limited ocean model quality, the capability to understand our changing ocean is called into question. With recent advances in unmanned aerial vehicles (UAVs), payload sensors, and their cost-effectiveness in field operations, these platforms offer a tantalizing hope of further extending the reach of oceanographers.

The paper describes the necessary infrastructure and software architecture to operate networked vehicles at sea. Operations are supported by the use of mixed-initiative control, a novelty in our field. As a first, we detail the use of Artificial Intelligence (AI) based control to command both a UAV from the ground and an AUV onboard. The paper describes the results of a field experiment which used the architecture at sea from a support vessel. We conclude with lessons learned and

[1]M. Faria, J. Pinto, J. Fortuna, H. Dias, R. Martins and J. Sousa are with the Faculty of Engineering, Porto University, Portugal {margaridacf,zepinto,jtasso}@fe.up.pt

[2]F. Py and K. Rajan are with the Monterey Bay Aquarium Research Institute, California {fpy,kanna.rajan}@mbari.org

[3]F. Leira and T. Johansen are with the Center for Autonomous Marine Operations and Systems (AMOS), Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim {frederik.s.leira, tor.arne.johansen}@itk.ntnu.no

future work.

## II. Related Work

Coordinating AUVs and UAVs vehicles in oceanic conditions has proved to be challenging. While vehicle-specific commercial solutions with proven field capabilities are in routine use especially within the military, they rely on mature waypoint-based command and control techniques.

Demonstrated applications with autonomous vehicles in realms spanning from civil engineering [1], agriculture [2], oceanography [3] and others [4] exist. A control law for cooperation between UAVs and terrestrial robots for large area coverage is described in [5]. These approaches use limited forms of autonomy which is prone to human errors since missions are defined in low-level behavior patterns.

The use of deliberative AI methods is less common, however. A number of different approaches for multi-vehicle task assignment in field operations have been attempted. Extensive work has been done by [6] developing a human readable Mission Specification Language based on Temporal Action Logic. In his delegation-based framework, an operator can specify behavior through high level goals that get distributed among agents. A partial order planner is then used to search for task assignment among UAVs, while an onboard forward-chaining planner generates actions for the agent in temporal order. A hierarchical 3-tier architecture has been developed and tested by [7]. It uses a collision avoidance layer at lower abstraction levels, a middle layer for path planning and an abstract layer for task assignment. This is where the optimal task scheduling is determined and allocated, it relies on a negotiation based algorithm for task distribution. The architecture developed by [8] has an on-board deliberative layer for distributed decision-making and an executive layer for task execution. Task planning can be done online or offline using a temporal planner that solves instances of the Travelling Salesman Problem using a brute-force method. The distribution of tasks among UAVs is then done using a market-based approach that does load-balancing across UAVs.

The approach described here builds on the work by [9]. These authors use an onboard planer to generate and execute Lagrangian survey plans around a moving drifter with an AUV.

In most of these instances, the system was designed and tested for a single vehicle type. Network and control of a heterogeneous group of vehicles has proven to be difficult especially in the field and we believe we are among the first to address it here.

## III. Architectural Approach

To overcome the challenge of adjusting to both the faster, short lasting UAV and the slower, longer lasting AUV we used the flexible network setup shown in Fig. 2. Operators are onboard a support vessel, interacting with the overall system via `NEPTUS` consoles. These are connected to a `Manta` communications gateway which provides long-range wifi, underwater (acoustic), satellite (Iridium) communications and internet access via 3G, when available. The consoles receive the state of all systems in the network, those connected directly to nearby `Manta` gateway as well as systems in remote locations. When removed from a gateway, the system sends its position to a central service through 3G/GSM or satellite. Execution of a plan is carried out by `DUNE`, deliberation of the plan is `T-REX`'s responsibility, while plan selection is in the hands of the human controller operating `NEPTUS`.

*a) Communications:* To cope with hardware heterogeneity we use Inter-Module Communications[1] (`IMC`) [10], a real-time message-based protocol that is used across all systems including unmanned vehicles, operator consoles, `Manta` and back-end servers. Typically, all `IMC`-compatible nodes use a discovery mechanism to disseminate available services to the network and, after discovery, communicate peer-to-peer.

During execution, all hardware nodes store incoming and outgoing messages by serializing them into a log file, for posterior mission analysis. The common format allows merging of different missions for a more integrated view of the data.

*b) Web services:* A centralization point, the `HUB` web service receives information from any source with a web, satellite or GSM connection. Data such as oceanographic conditions, vehicle or ship locations need to be parsed, integrated and made available through an API to `NEPTUS` and other applications. The `HUB` can in turn send data out through either satellite links or the web. Consequently, teams at remote locations and without direct wifi connection can coordinate positions of different vehicles using the `HUB` as a relay while exchanging "virtual positions" (a desired position is requested by one team and updated by another). Not only can all teams see requests and updates, but they can use multiple pathways to receive the same information, leading to a robust and reliable communication mechanism.

*c) `DUNE`:* The Uniform Navigation Environment (`DUNE`)[2] [11] is the embedded software used onboard

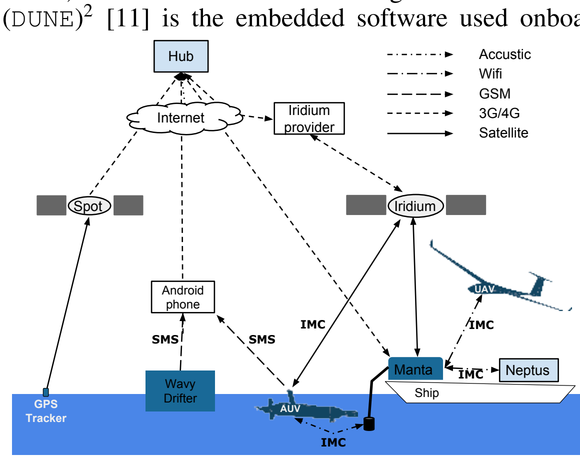



Fig. 2. Network connectivity across platforms.

[1] `https://github.com/LSTS/imc`
[2] `https://github.com/LSTS/dune`

our unmanned vehicles, data loggers and `MANTA` gateways. It is composed of independent tasks that communicate by broadcasting and subscribing to `IMC` messages. `DUNE` tasks can be diverse; some are responsible for interfacing hardware with `IMC` (either sensors or actuators), others are part of the chain of command either translating higher-level goals into low-level actuation commands or interpreting low-level sensor readings into state estimates. To integrate any sensor or actuator hardware, only the tasks dedicated to its management need to know the specific details. Each system uses a task configuration adjusted to its purpose, available sensors, actuators and communications.

The `DUNE` control architecture requires that a single maneuver controller is driving the vehicle at any given instant. To achieve complex behavior, a plan supervisor is capable of parsing a (scripted) plan and instantiating maneuver controllers according to that specification. Alternatively, external controllers like `T-REX` (described below) guide the vehicle by instantiating the *FollowReference* maneuver controller. When this controller is active, it accepts references with the desired state from authorized sources and transforms them into vehicle actions. The desired state has elements like coordinates, altitude/depth or speed.

*d) `NEPTUS`:* is an open-source software infrastructure[3] used to build graphical interfaces for networked vehicle systems. The main user interfaces are the Operator Console and the Mission Revision and Analysis tool [11]. Fig. 3 is an example of an AUV operator console, consisting of a set of plug-ins such as *visual widgets* (a map, a mission tree, a vehicle state display), *map layers* (cartography, live ship positions), *daemons* (operational limits monitor), *popups* (multi-system listing) and *map interactions* (planning, map editing, measuring, real-time strategy etc. ). Vehicles can be controlled using four interaction styles: point-and-click to manually execute a maneuver, scripted plans for subsequent execution, real-time planning where the user designs a plan local to `NEPTUS` that can be edited on the fly and finally deliberative planning, an advanced AI-based capability to synthesize, execute and re-plan on the fly.

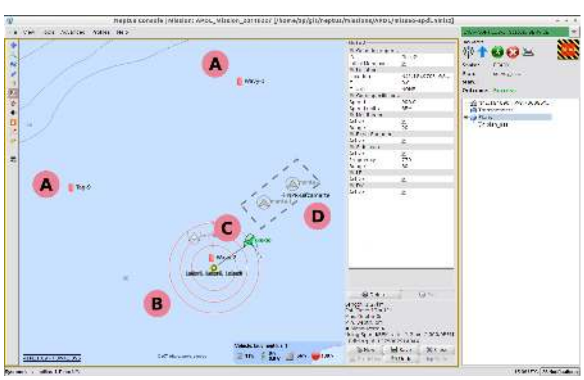



Fig. 3. Neptus console interface. '**A**' shows GPS trackers, '**B**' the path of the UAV, '**C**' icons for an UAV in green and an AUV in white, '**D**' for two `MANTA`s and an AIS marker for the *Bacamarte*

To improve operator responsiveness and situational awareness, `NEPTUS` allows each console to be adapted to a particular mission, vehicle or operator function. A mission file is a configuration that stores all the mission elements: map features, vehicle configurations, plans and checklists. All elements can be edited and shared as `IMC` messages.

*e) `T-REX`:* is an open source on-board adaptive control system[4] that integrates AI-based planning and state estimation in a hybrid executive. The overall system is a composition of application specific functional components called *reactors*, interacting with each other through timelines or state variables. Goals are dispatched to other reactors down the hierarchy while observations recording the state of the world are posted up this hierarchy. Timelines are populated with observations (temporal predicates) posted by their owner reactors. In this system, a goal is a projected future observation. When a reactor receives a goal in a timeline, it will attempt to decompose it by potentially deliberating over a series of other (sub)goals and eventually report it as an observation, in nominal execution. The role of the `T-REX` agent then, is to ensure that all the reactors will be able to interact *concurrently* so that they are informed of state evolution that may impact them, have a sufficient amount of time to synthesize plans and coordinate plan dispatch across reactor boundaries. Details of `T-REX` are beyond the scope of this paper and can be found in [12], [13], [14].

*f) UAVs:* Three identical low-cost UAV platforms (Fig. 1) were deployed and recovered from our support vessel. Fig. 4 shows the main components of this UAV platform. The `X8` flying wing is an electric vehicle based on a RC model which holds an ISEE IGEPv2 single board computer running `DUNE`. It has a wingspan of 212 cm, a take-off weight of around 3.5 Kg and is able to fly up to one hour continuously. It uses the `ArduPilot`, an open-source low-cost autopilot. To integrate it with `DUNE` a specialized driver task was created which translates telemetry and guidance commands. The main communication channel is a 5 GHz wifi with line-of-sight ranges of up to 10 Km. Different

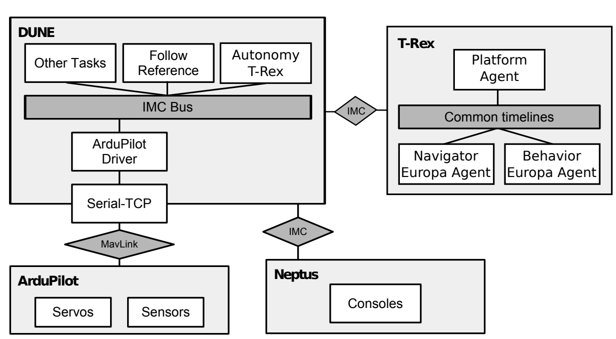



Fig. 4. The UAV toolchain used in the experiments.

[3]`https://github.com/LSTS/neptus`

[4]`https://code.google.com/p/trex2-agent/`

camera configurations were tested including an IP, a digital compact camera and an analog Infra-Red camera (Fig. 5(a)). The Infra-red camera used was a FLIR Tau2 336, connected to an analog sending device working at 1.3 GHz. An analog receiver device was located on the ground, where the video feed was captured and stored.

The UAV was deployed using a crossbow/catapult mechanism and can land either on a grass/dirt/asphalt strip or on a capture net used in confined spaces like ship decks such as seen in Fig. 1. Regular operations required one operator who uses NEPTUS and T-REX to control the vehicle while in autonomous operations mode and a safety pilot for take-off, landing and emergency manual control.

*g) AUVs:* Three AUVs designed at the Univ. of Porto were deployed, with two simultaneously in the water at most. These AUVs have $\sim$ 6 hours of operational capacity when running underwater at 3 knots. They weigh 20 Kg, have a 15 cm diameter and length of 180 cm. Each vehicle runs DUNE in its AMD Geode LX 800 processor with 1024 MB of RAM memory. It can communicate using Iridium, 802.11 wifi, GSM and acoustic modems. For navigation it relies on a Microstrain 3DM-GX3 AHRS and a NavQuest 600 MicroDVL (Doppler Velocity Log). For water column measurements it uses an RBR XR620 CTD. The vehicle is deployed and recovered manually, either from the side of a ship (Fig. 1) or from a RHIB.

## IV. Experimental Results

Our field experiments were conducted off Sesimbra, Portugal from July 8th to July 17th onboard the Portuguese Navy vessel NRP *Bacamarte*. During this experiment, cameras onboard UAVs were used to detect features of interest and AUVs were used to provide contextual information around drifters entrained in these features. One AUV would sample around a drifter's Lagrangian frame of reference [15].

### *A. UAV flight operation*

For the UAVs to visually confirm a target, like a drifter in the water, we used a *spiral* pattern encoded as a number of increasingly larger circles. Doing so guarantees that the camera on the UAV will capture footage in an angle free of the Sun's specular reflection. It covers the area where the target might have drifted starting from its last known position. Human operators can task the UAV to engage this behavior by sending a point on the console's map as a target. Since the updated positions of all the systems are shown on the map, it is simple to send the desired position to the UAV. As a result of this interaction, NEPTUS creates a high-level goal encoded as an IMC message with coordinates, desired speed and altitude. This message is sent to a shore-side instance of T-REX that integrates the objective into its current sequence of planned UAV manoeuvres using three reactors: *Platform*, *Navigator* and *Spotter*. The *Platform* reactor translates IMC messages into T-REX observations and vice-versa. This reactor is also responsible for commanding DUNE using guidance messages sent to the *FollowReference* maneuver controller. The *Navigator* reactor is responsible for enabling the vehicle to travel from one waypoint to the next adapting the plan to hardware failures or navigation requirements. The *Spotter* reactor (specific to UAVs) is responsible for generating a sequence of circular motions that achieve the desired search pattern. T-REX reactors are kept updated with IMC messages about the position and speed of the vehicle, state of plan execution, operational limits and connectivity. Only the *Spotter* reactor knows the complete plan and issues commands to the *Platform* reactor based on the progress of the actions allowing plan adjustment and to queue of waypoints.

### *B. AUV under task deliberation*

AUV control, deliberation, execution and adaptation of plans is done onboard using T-REX with a number of reactors. The *Navigator* and *Platform* reactors are identical to those used in the UAV. The *Yoyo* reactor generates a set of waypoint references in between two locations. These references command the vehicle to change its depth while moving in the direction of the destination waypoint in a see-saw pattern for making high-density upper water-column observations. The *Drifter* reactor, also specific to AUV control, is responsible for generating a survey pattern that makes the vehicle encircle a drifter in its Lagrangian frame of reference [15]. The *Proxy* reactor allows a model to be distributed over multiple T-REX instances, allowing a timeline to be synchronized across the network (and through Iridium) – such an approach allowed us to maintain the human operator in control by having her/him selecting the drifter to be followed.

The generation of the survey pattern is shared between two reactors. The *Drifter* generates the horizontal path solely based on the last known drifter position and velocity (speed and heading). The generated path is usually a distorted box centered on the drifter position. The distortion applied to this box results from applying the velocity of the drifter to them, thus making the AUV keep up with expected movement of the drifter. The *Yoyo* reactor, generates the vertical references by following a state machine. Its latency of execution allows it to function in shallow areas with unknown bathymetry since it will stop descending as soon as the bottom is near the vehicle which is detected using Dopple Velocity Log measurements. The approach described here builds on the work by [9].

### *C. Other Operational results*

The toolchain consisting of DUNE, NEPTUS, HUB, T-REX and IMC made it possible to monitor and control several vehicles, often simultaneously. Gateways, running DUNE were capable of bridging different communication methods including underwater acoustics, wifi and the Web (through 3G/HDSPA). NEPTUS consoles allowed multiple operators to follow the operation and control vehicles individually. Additional information like cartographic maps, real-time ship traffic and meteorological forecasts were overlaid in the consoles, allowing operators and scientists to make accurate predictions and informed decisions.

An unexpected outcome of the architecture, process and methods presented here, was the *flexibility* to dynamically target our combined assets towards an opportunistic science problem of tracking riverine fronts. Such fronts often have a steep thermal gradient which can be detected with an IR camera. Science input from oceanographers present showed the existence of such fronts, albeit weak in nature and close to shore. We used the X8 with the IR camera to map an area in near proximity to the support vessel and observed a thermal signature on the video feed suggesting the existence of such a gradient (Fig. 5(a)). A drifting buoy was then deployed at this location as a marker for the moving frontal mass.

A subsequent part of this experiment involved an AUV executing Lagrangian patterns around the drifting sensor, since the drifter was now entrained in the front. Using this method, it was possible to confirm the existence of frontal zones by comparing the temperature and salinity gradients of survey yoyo's. The acquired data strongly suggested the presence of different bodies of water (see figure 5(b)).

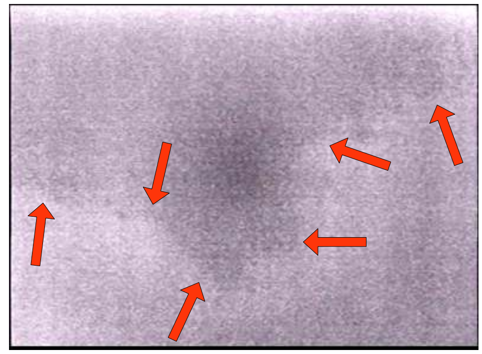
Considering the widespread area of operation and the opportunistic nature of an experiment of this type, the HUB 's web services were essential to aggregate information from disparate sources with all the associated diverse input channels.

## V. Lessons Learned

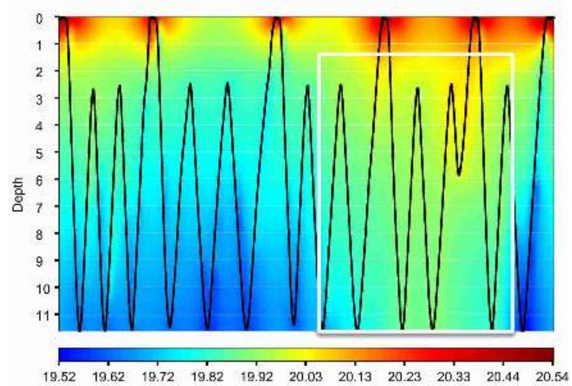
We describe the successful deployment of multiple heterogeneous robotic platforms for coordinated observation and tracking in the coastal ocean. The experiments were conducted from a support vessel under challenging conditions, not without failures, including last-minute procedure changes. The challenges encountered could be labeled mostly as logistical or operational; however some key technical issues were also found.

The experiment was part of a larger scale deployment encompassing tests and scenarios to be executed by the Portuguese Navy. This was both an advantage and a challenge; while it brought together people with differing skills and experience to do opportunistic science, it also meant multiple objectives had to be satisfied with a limited number of shared robotic assets. One solution to this problem was to have a tight schedule where each asset was used for a specific end; this however resulted in reduced flexibility for our engineering tests. Nature too played a part, especially since our UAVs relied on stable weather conditions including some wind lift to fly from the support vessel, impacting coordinated aerial/sub-surface measurements.

The IMC protocol has been designed so that different vehicles and consoles can exchange real-time data. However, due to the simultaneous operations of a number of vehicles and the available bandwidth, it became necessary to split the wireless network originating on the support vessel into two different frequencies, one operating at 5.8 GHz for the UAV team and the other at 2.4 GHz for the AUV. This isolation lead to loss of situational awareness as each team only had access to their own vehicles. With the HUB the position of all the system became visible supporting team synchronization. Yet another communication related problem encountered had to do with the radio transmitter used with the IR camera, which introduced static noise into the video feed. A potential solution would be to store the analog video signal directly on the UAV using a frame grabber.



(a)




(b)

Fig. 5. (a) Aerial infra-red imagery extracted from real-time video feed showing a riverine front with two distinct bodies of water indicated by the arrows. (b) Thermal gradient in the CTD data and the distinct water mass (within the white box) of the AUV survey of the frontal zone. Black lines indicate the yoyo pattern executed by the vehicle in the water column.

The large difference in operational speeds for UAVs, RHIBS and AUVs showed that it is difficult to re-task a collective of vehicles unless there is tight human operator coordination. In future deployments which might require co-located sampling in aerial and underwater environments, we believe that using several spatially distributed AUVs will lead to faster observations to compensate for their relative speed. However, this would require a careful balancing of assets' operating area and monitoring of resources such as onboard energy.

Deliberative planning onboard AUVs relieves the operator of validating any manually defined plans. However, we found that having the AUV fully autonomous makes it difficult for operators to follow the behavior of an AUV in real-time. Limited observability led to lack of opacity of what the AUV is doing at any instance of time. For this reason, we ensured that the operator was an integral part of the deliberation process. For instance, in scenarios associated with contextual surveys around a drifter, the operator console received and plotted all drifter positions. The operator was therefore situationally aware and made responsible for selecting which of the many drifters to target and which tasks and partial plans were to be sent to the AUV's onboard planner.

Overall, tolerance to communication faults and redundancy of communication means proved key in this field experiment. In particular, the UAV mitigation scenario in the event of communication loss (in which the UAV loiters around a 'safe point' and altitude) provided security for bolder tests.

## VI. Conclusions

In the context of this experiment, we successfully demonstrated the integration of deliberative techniques in `T-REX` with `DUNE` with the help of motion-control command sets based on concepts such as trajectories, paths, and waypoints. We are now working on tighter integration and underlying theory development related to guarantees for safe command execution to ensure that the planner will not lead to unexpected/unsafe behaviors. Towards this end, we are investigating the use of approximations of the vehicle's reach sets which are computed off-line [16] and of controllers based on the interpolation of the grid representation of underlying value functions [17] to address some of the questions of integrating different methods within an appropriate control and computation framework. Further, the integration of a deliberative on-board planner with a low level vehicle control system raises several questions at the intersection of AI and control systems; for instance what are 'good' abstractions of low-level controlled behaviors for planning purposes?

The complete toolchain was used for the very first time in the experiment. Logistical and operational challenges related to such experimentation in the field will continue to be a significant part of such deployments. However, we have taken preliminary steps to seamlessly insert humans and fully autonomous systems to control heterogeneous vehicles in real world environments. And in this context, the flexibility given by the `HUB` to integrate information coming from diverse sources proved to be a valuable asset for situational awareness.

The experiment also supported the possibility for multiple `T-REX` instances controlled by a centralized deliberation engine, which we hope to investigate further using such quasi-centralized schemes to decompose high-level objectives and constraints.

## Acknowledgements

This work was supported in part by the Portuguese Navy. We thank 1st Lt. Rúben Robalo Rodrigues and 1st Lt. Cordeiro Cavaleiro. Porto University authors were funded by ERDF - Transnational Programme - Atlantic Area (Project NETMAR), and EU FP7 ICT, (Project Noptilus). MBARI authors were funded by NSF grant No. 1124975 and NOAA grant No. NA11NOS4780055 as well as a block grant from the David and Lucile Packard Foundation to MBARI. NTNU was funded by the Research Council of Norway and Statoil through the Center of Excellence on Autonomous Marine Operations and Systems, grant 223254.

## References

[1] A. M. Jensen, T. Hardy, M. McKee, and Y. Q. Chen, "Using a multispectral autonomous unmanned aerial remote sensing platform (aggieair) for riparian and wetland applications," in *Proc. IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS)*, 2011.

[2] Y. Kaheil, M. Gill, M. McKee, L. Bastidas, and E. Rosero, "Downscaling and assimilation of surface soil moisture using ground truth measurements," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 46, no. 5, pp. 1375–1384, 2008.

[3] J. P. Ryan, S. Johnson, A. Sherman, K. Rajan, F. Py, H. Thomas, J. Harvey, L. Bird, J. Paduan, and R. Vrijenhoek, "Mobile autonomous process sampling within coastal ocean observing systems," *Limnology & Oceanograhy: Methods*, vol. 8, pp. 394–402, 2010.

[4] H. Chao and Y. Chen, *Cooperative Remote Sensing and Actuation Using Networked Unmanned Vehicles*. Wiley-IEEE Press, 2012.

[5] A. Jensen, B. Neilson, M. McKee, and Y. Chen, "Thermal remote sensing with an autonomous unmanned aerial remote sensing platform for surface stream temperatures," in *Proc. IEEE Int. Geoscience and Remote Sensing Symp. (IGARSS)*, 2012, pp. 5049–5052.

[6] P. Doherty, F. Heintz, and J. Kvarnström, "High-level mission specification and planning for collaborative unmanned aircraft systems using delegation," *Unmanned Systems*, vol. 1, no. 1, pp. 75–119, 2013.

[7] S. Moon, E. Oh, and D. H. Shim, "An integral framework of task assignment and path planning for multiple unmanned aerial vehicles in dynamic environments," *Journal of Intelligent and Robotic Systems*, vol. 70, no. 1-4, pp. 303–313, April 2013.

[8] I. Maza, K. Kondak, M. Bernard, and A. Ollero, "Multi-UAV cooperation and control for load transportation and deployment," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1–4, pp. 417–449, 2010.

[9] J. Das, F. Py, T. Maughan, T. OReilly, M. Messi, J. Ryan, G. S. Sukhatme, and K. Rajan, "Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 626–646, 2012. [Online]. Available: http://ijr.sagepub.com/content/31/5/626.abstract

[10] R. Martins, P. Dias, E. Marques, J. Pinto, J. Sousa, and F. Pereira, "IMC: A communication protocol for networked vehicles and sensors," in *OCEANS 2009 - EUROPE*, 2009, pp. 1–6.

[11] J. Pinto, P. Calado, J. Braga, P. Dias, R. Martins, E. Marques, and J. Sousa, "Implementation of a control architecture for networked vehicle systems," in *Proc. IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, 2012.

[12] F. Py, K. Rajan, and C. McGann, "A Systematic Agent Framework for Situated Autonomous Systems," in *9th International Conf. on Autonomous Agents and Multiagent Systems (AAMAS)*, Toronto, Canada, May 2010.

[13] K. Rajan and F. Py, "T-REX: Partitioned Inference for AUV Mission Control," in *Further Advances in Unmanned Marine Vehicles*, G. N. Roberts and R. Sutton, Eds. The Institution of Engineering and Technology (IET), August 2012.

[14] K. Rajan, F. Py, and J. Berreiro, "Towards Deliberative Control in Marine Robotics," in *Autonomy in Marine Robots*, M. Seto, Ed. Springer Verlag, 2012.

[15] J. Das, F. Py, T. Maughan, M. Messie, T. O'Reilly, J. Ryan, G. S. Sukhatme, and K. Rajan, "Coordinated Sampling of Dynamic Oceanographic Features with AUVs and Drifters," *Intnl. J. of Robotics Research*, vol. 31, pp. 626-646, 2012, April.

[16] J. E. da Silva and J. B. de Sousa, "A dynamic programming based path-following controller for autonomous vehicles." *Control and Intelligent Systems*, vol. 39, no. 4, 2011.

[17] ——, "Cooperative path planning in the presence of adversarial behavior." Fortuna, Luigi (ed.) et al., From Physics to Control through an Emergent View 4th international conference on physics and control (PhysCon2009), Catania, Italy, September 2009.