PANDORA: Persistent Autonomy through Learning, Adaptation, Observation and Re-planning *

David M. Lane * Francesco Maurelli * Tom Larkworthy *
Darwin Caldwell ** Joaquim Salvi *** Maria Fox ****
Konstantinos Kyriakopoulos †

* Ocean Systems Lab, Heriot-Watt University, Edinburgh, UK, contact: d.m.lane@hw.ac.uk
** Advanced Robotics Department, Italian Institute of Technology (IIT), Genova, Italy
*** Depart. d’Arquitectura i Tecnologia de Computadors, Campus Montilivi, Universitat de Girona, Spain
**** Department of Informatics, King’s College London, UK
† Control Systems Laboratory, National Technical University of Athens (NTUA), Greece

Abstract: Autonomous robots are not very good at being autonomous. Operating in real environments, they easily get stuck, often ask for help, and generally succeed only when attempting simple tasks in well-known situations. PANDORA is a three year project that will develop and evaluate new computational methods to make underwater robots Persistently Autonomous, significantly reducing the frequency of assistance requests. The key to this is an ability to recognise failure and respond to it, at all levels of abstraction and time constant. Under the guidance of major industrial players, validation tasks of Inspection, cleaning and valve turning will be trialled with partners’ AUVs in Scotland and Spain.

1. INTRODUCTION

Whilst humans and animals perform effortlessly doing complicated tasks in unknown environments, our human-built robots are not very good at being similarly independent. Operating in real environments, they easily get stuck, often ask for help, and generally succeed only when attempting simple tasks in well-known situations. We want autonomous robots to be much better at being autonomous for a long time (persistent autonomy), and to be able to carry out more complicated tasks without getting stuck, lost or confused. Following the Deep Water Horizon disaster in the BP Macondo oilfield in the Gulf of Mexico in 2010, Oil Companies are developing improved ways to cost effectively and safely carry out more frequent inspection, repair and maintenance tasks on their subsea infrastructure. This is particularly challenging in deep water. To date, Autonomous Underwater Vehicles (AUVs) have been deployed very successfully for various forms of seabed and water column transit survey. First commercial units will soon be applied to simple hovering inspection tasks, with future units expected to address much harder intervention where contact is made to turn a valve or replace a component. Because these vehicles reduce or remove the need for expensive ships, their adoption is expected to grow over the next 5 to 10 years.

To be successful commercially, these hovering AUVs must operate for extended periods (12-48 hours +) without the continual presence of a surface vessel. They must therefore demonstrate persistent autonomy in a challenging environment. We therefore choose this application focus to evaluate the projects research, with guidance from BP, Subsea7 and SeeByte Ltd. on the projects Industrial Advisory Group. We have identified three essential areas where we believe core research is required in order to provide the essential foundations for Persistent Autonomy:

- Describing the World
- Directing and Adapting Intentions
- Acting Robustly

1.1 Describing the World

Whilst current generations of autonomous robots use semantic descriptions of themselves, their environment and their tasks as a basis for directing their actions, these descriptions are typically fixed and certain. We wish to develop and evaluate semantic representations (using ontologies) that are explicitly uncertain, and that continually evolve, change and adapt based on sensor data that observes the environment and the robot as it acts. This builds on low level perception detecting features in sensor data, and on status assessment that estimates the actual state of the robot and task execution as it progresses. It also involves directing the sensors during execution to make observations appropriate to the task and situation (focus
Fig. 1. PANDORA: Computational architecture to develop and study persistent autonomy

1.2 Directing and Adapting Intentions

By observing the effects of actions in this changing probabilistic world above, we wish to develop and evaluate mechanisms to create and refine task plans and primitives. By maintaining and modifying models of actions and their effects, the autonomous robot formulates plans, updates them or repairs them on the fly when they don’t work, accounting for uncertainty and within time and resource budgets. This ability to model cause-and-effect and reason about state and goal satisfaction relies on these adaptive representations of the world. Thus equipped, the autonomous robot can change its strategic approach, its tactical plan and perhaps eventually its purpose to remain safe, operate efficiently and survive over an extended period without recourse to a human operator.

1.3 Acting Robustly

Noisy sensor data and disturbances on the robot and the world produces uncertainty and errors in the location of objects and the robot itself. Attempts to execute atomic actions from planning above will be prone to failure therefore, unless local real-time adaptation of motions and contacts can compensate for these errors. Further, robots have to be instructed on how to execute these actions, and how to respond to these local uncertainties and disturbances.

2. ARCHITECTURE

Figure 1 outlines the computational architecture designed for development and study of persistent autonomy. Key is the notion that the robots response to change and the unexpected takes place at one or a number of hierarchical levels. At an Operational level, sensor data is processed in Perception to remove noise, extract and track features, localise using SLAM, in turn providing measurement values for Robust Control of body axes, contact forces/torques and relative positions. One of the goals is to further explore some of the current approaches (Aulinas et al. [2011], Lee et al. [2012]) and integrate them on a real vehicle. In the cases where a map is given, localisation techniques will be used (Petillot et al. [2010]), with a specific attention to active localisation (Maurelli et al. [2010]). Relevant work on robust control can be found in Panagou and Kyriakopoulos [2011], Karras et al. [2011]. At a Tactical Level, Status Assessment uses status information from around the robot in combination with expectations of planned actions, world model and observed features to determine if actions are proceeding satisfactorily, or have failed. Alongside this, reinforcement and imitation learning techniques are used to train the robot to execute pre-determined tasks, providing reference values to controllers. Fed by measurement values from Perception, they update controller reference values when disturbance or poor control causes action failure. The learning block will be lead by IIT, with relevant expertise in the field (Kormushev et al. [2011], Calinon et al. [2010]). Finally at a Strategic level, sensor features and state information are matched with geometric data about the environment to update a geometric world model. These updates include making semantic assertions about
the task, and the world geometry, and using reasoning to propagate the implications of these through the world description. Task Planning uses both semantic and geometric information as pre-conditions on possible actions or action sequences that can be executed. When Status Assessment detects failure of an action, Task Planning instigates a plan repair to assess best response, if any. Where there is insufficient data to repair, Task Planning specifies Focus Areas where it would like further sensor attention directed. These are recorded in the World Model and propagated through Status Assessment as Focus of Attention to direct the relevant sensors to make further measurements. Relevant work on Planning has been performed by Fox et al. [2011, 2012].

3. DOMAIN AND TEST SCENARIOS

We have chosen autonomous underwater inspection and intervention in the oilfield as our application focus. The specific domain challenges that require these advanced features to realise persistent autonomy are (Table 1):

1. Coupled Dynamics: Underwater vehicles have dynamics that are coupled across the axes, significantly time varying, with large inertias (stored energy). Adequate control across a range of payloads and operating conditions (e.g. pitch angles of the vehicle) remains challenging. This is essential for inspection where imaging sensor position and orientation relative to the structure must be maintained.

2. Noisy Sensors: Much sensor data is acoustically derived (e.g. navigation, imaging). It is therefore inherently noisy and of low resolution with some latency resulting from the speed of sound in water. This leads to uncertainties in the position of the vehicle and the objects around it.

3. Currents: The medium the vehicle moves through (the water) also moves in unpredictable ways and disturbs its motion. This causes navigation to drift, and perturbs in eddies around structures.

4. Limited Communication: E/M waves do not propagate underwater at useful frequencies. Communication is therefore acoustic, with severely limited bandwidth and noise corruptions. AUVs cannot be tele-operated from the surface. Removal of the umbilical cable (c.f. a tethered Remotely Operated Vehicle ROV) enforces the requirement for autonomy therefore.

5. Reaction Forces: During contact, the vehicle can only compensate reaction forces by thrusting. It therefore behaves like a passively compliant system subject to disturbances in applications where precision is required e.g. for grasping. Docking first to a subsea structure assists this, but docking bars or other fittings are not always available on the structure, and are expensive to install.

6. Object Motion: Subsea objects to be inspected move with time constant slower than the vehicle. In some cases (e.g. cleaning marine growth) they move as a result of the vehicles actions. The vehicle must compensate for these unexpected movements.

7. Finite Energy: AUVs carry batteries for power. The mission and the vehicle systems have to be managed effectively to keep the vehicle safe and complete useful tasks. Where energy is being consumed more quickly than expected, the mission plan must adapt.

8. Partially Known Environment: Whilst environments should be known a priori (subsea infrastructure, ship hull), the geometric world model data is often incomplete or incorrect, through previously undocumented maintenance, changes from as built drawings just before installation, and damage (mapping the debris after the Deep Water Horizon explosion was a serious challenge before intervention could commence). The AUV must therefore be ready to map unexpected structures, and adapt missions accordingly.

9. Training: Skilled underwater vehicle pilots are in short supply. They have skills of motion control and task execution that must be readily captured into the vehicle without extensive programming.

10. Failures: While all the symptoms of success may be present in the immediately available sensing and control data, in fact the task may have failed e.g. servo lock onto the wrong anchor chain or riser for inspection. This must be detected and addressed without recourse to the operator. With these specific challenges in mind. The following validation scenarios were selected for testing the ability of the PANDORA technology to deal with complex tasks.

3.1 Task A: Autonomous inspection of a submerged structure e.g. a ship hull (FPSO) or manifold (Fig. 2)

A hover capable autonomous underwater vehicle is equipped with a forward looking sonar, a video camera and dead reckoning navigation system. The structure is partially known, but there are inconsistencies between it and the geometric world model. The vehicles high-level goal is to autonomously inspect the entire structure with no data holidays, and bring back a complete data set of video and sonar for mosaicking and post processing. There may be a current running, and the optical visibility may be very poor. In some cases, the sonar inspection sensors must be kept at a constant grazing angle relative to the structure, for best performance. In the absence of a pan and tilt unit, the vehicle must dynamically pitch, yaw and roll to maintain this orientation.

<table>
<thead>
<tr>
<th>Id</th>
<th>Challenges</th>
<th>Describing the World</th>
<th>Directing and Adapting Intentions</th>
<th>Acting Robustly</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Coupled Dynamics</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>Noisy Sensors</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>3</td>
<td>Currents</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>4</td>
<td>Limited Communication</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>5</td>
<td>Reaction Forces</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>6</td>
<td>Object Motion</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>7</td>
<td>Finite Energy</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>8</td>
<td>Partially Known Environments</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>9</td>
<td>Training Failures</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 1. Relevance of the elements of PANDORA’s computational architecture to these challenges
3.2 Task B: Autonomous location, cleaning and inspection of an anchor chain (Fig. 3)

A hover capable autonomous underwater vehicle is equipped as above, but in addition carries a high-pressure water jet. Its goal is to locate the correct anchor chain of an FPSO and traverse it to remove the marine growth on all sides using the water jet. Thereafter it revisits the chain and brings back complete video inspection data for subsequent post processing. The reaction forces from the water jet introduce significant forces and moments onto the vehicle, and also disturb the anchor chain. Both are therefore in constant disturbed motion. The optical visibility drops to zero during jetting as the marine growth floats in the water. There may be sea currents moving over the anchor, creating minor turbulence downwind of the chain. The chain is located adjacent to flexible risers of slightly larger dimension bringing oil to the surface.

3.3 Task C: Autonomous grasping and turning of a valve from a swimming, undocked vehicle (Fig. 4)

A hover capable autonomous underwater is equipped as in Task A, with a simple robot arm at the front. Its goal is to locate the correct valve panel of a subsea manifold and open the correct valve. On each panel a selection of valve heads are exposed, each with a T bar attached for grasping. The vehicle must identify the state of the valves (open, close, in-between) from the T bar orientations, and if appropriate, use the robot arm to grasp the correct valve and open it. The vehicle does not dock, because there are no docking bars on the panel. It must therefore hover by swimming, counteracting any reaction forces from the turning. It must also ensure that the gripper position and orientation of the gripper after grasping does not cause significant shear forces in the T bar, and break it off. The visibility is generally good, but there may be sea currents running and minor turbulence down current from the manifold.
4. VEHICLE ASSETS

The testbed assets for these integrations and trials include HWU’s Nessie AUV, UdG’s Girona 500 AUV and HWU’s Cartesian robot mounted over a test tank. It can be animated with a 6DOF AUV dynamic model, and will be used to initially test the Perception and Status Assessment techniques in the project.

For in water trials with the real vehicles, two trials locations will be employed. For Task A, HWU’s Nessie vehicle will be used, featuring its unique bow and stern thruster arrangements enabling 360 rotation in yaw pitch and roll, and hovering in surge, sway and heave. Initial trials will take place in the small test tank at HWU. Thereafter, facilities at the Underwater Centre in Fort William, Scotland will be used, where larger structures are installed for offshore ROV pilot and diver training. For Tasks B and C, UdG’s Girona 500 vehicle will be used in the underwater test facilities that form part of the Girona laboratory. Some of the demonstration platforms are shown in Fig. 5.

5. VALIDATION

A comprehensive list of validation metrics have been developed to test the ability of each system to meet its criteria. Ultimately the best all-encompassing metrics of persistent autonomy is quantitative analysis of robot engineer interactions, which will be greatly reduced by PANDORA’s success. So overall, we measure our success by the reduction in the number of times the operator is called to assist a stuck robot during execution of tasks and sequences with noisy sensor data. We also count the number of successful automatic recoveries the robot achieves from an execution failure. These require world modelling to detect the failure, task planning to indicate corrective action, and reinforcement learning/adaptive control to successfully execute once more.

Beyond this, the overarching performance indicators within each core theme are:

- **Describing the World:**
  - Trends in numerical errors of vehicle location and object location/geometry (or other indirect measures such as residuals, covariance, OSPA error, Hellinger distance), and probabilities in semantic relationships (ontologies).
  - Number of correct and incorrect diagnoses of task execution failure.
- **Directing and Adapting Intentions:** Comparison of the number of failing actions in an automated planning/re-planning system with the number of failures produced by hand-built plans, assuming given world information (efficiency and resource utilisation are therefore implicit).
- **Acting Robustly:** Position and orientation error norm per unit distance, and the average wrench (force and torque) error norm.

6. SYSTEM INTEGRATION

Given the project is expected to further the state-of-the-art in a number of different research areas, we deemed it appropriate to adopt an Agile system development methodology. Agile methods promote a project management process that encourages frequent inspection and adaptation, a leadership philosophy that encourages teamwork, self-organization and accountability, a set of engineering best practices that allow for rapid delivery of high-quality solutions. Conceptual foundations of this framework are to be found in modern approaches to operations management and analysis such as lean manufacturing, soft systems methodology, speech act theory (network of conversations approach), and Six Sigma.

Agile methods choose to do things in small increments with minimal planning, rather than long-term planning. Iterations are short time frames (known as ‘timeboxes’) which typically last from one to four weeks. Each iteration is worked on by a team through a full development cycle, including planning, requirements analysis, design, coding, unit testing, and acceptance testing when a working prototype is demonstrated to stakeholders. This helps to minimize the overall risk, and allows the project to adapt to changes quickly. Documentation is produced as required. Multiple iterations may be required to release new features.

Team composition in an agile project is usually cross-functional and self-organizing without consideration for any existing hierarchy or formal roles of team members. Team members normally take responsibility for tasks that deliver the functionality of an iteration. They decide for themselves how they will execute during an iteration. Agile methods emphasize face-to-face communication over written documents, when working in the same location, or in different locations but having video contact daily, communicating by videoconferencing, voice, e-mail etc.

Agile methods emphasize working prototypes as the primary measure of progress. Combined with the preference for face-to-face communication, agile methods usually produce less written documentation than other methods. In an agile project, documentation and other project artefacts all rank equally with working product. Stakeholders are encouraged to prioritize them with other iteration outcomes based exclusively on application value perceived at the beginning of the iteration.

Specific tools and techniques such as continuous integration, automated or unit tests, pair programming, test driven development, design patterns, domain-driven design, code refactoring and other techniques are often used to improve quality and enhance project agility.

We will apply this methodology outside it’s normal business application to an open ended research problem. Our contribution will be evaluation and evolution of prescribed methods above to the practicalities of realising and experimenting a research challenge of some significance and complexity.

7. CONCLUSION

This paper has presented the challenges to be addressed by the FP7 Project PANDORA on persistent autonomy. Current existing autonomous systems require frequent operator intervention. The focus of Pandora is to enhance the long-term autonomy of AUV missions, through increased
cognition, at all the levels of abstraction. An agile development approach will be used which will allow frequent measurements of development metrics to rapidly optimize the system on the three main experimental scenarios. This frequent feedback will allow drive the expected increase of autonomy over the project’s lifetime.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Union Seventh Framework Programme FP7/2007-2013 Challenge 2 Cognitive Systems, Interaction, Robotics under grant agreement No 288273 PANDORA

REFERENCES
