Intelligent Autonomy for Collaborative Intervention Missions of Unmanned Maritime Vehicles

Carlos C. Insaurralde, Yvan R. Petillot
Institute of Sensors, Signals and Systems
Heriot-Watt University
United Kingdom
{c.c.insaurralde, y.r.petillot}@hw.ac.uk

***Abstract*—Current unmanned marine vehicles are increasing their autonomy so that more sophisticated maritime missions can be carried out. Remotely operated vehicles are no longer cost-effective since they are limited by economic support costs, and the presence and skills of the human operator. This paper proposes an Intelligent Control Architecture (ICA) to enable multiple collaborating marine vehicles to autonomously carry out underwater intervention missions. The ICA is generic in nature but aimed at a case study where a marine surface craft and an underwater vehicle are required to work cooperatively. Service-oriented computing as well as agent technology lay the ICA foundations. An ontological database captures the operator skills, platform capabilities, and potential changes in the environment. This knowledge enables reasoning agents to plan missions. The ICA implementation is verified in simulation. This paper also presents architectural details of the ICA, and experimental results from simulation.**

Keywords-cognitive control; maritime robotics; unmanned marine vehicles; agent techonology; service-oriented computing

I. INTRODUCTION

New generations of ocean systems are demanding more and more autonomous assistance in underwater activities, e.g. maritime rescue, inspection and light intervention on offshore structures, marine biology applications, etc. Most of the current industrial approaches still involve Remotely Operated Vehicles (ROVs) that are manually tele-operated from the surface. The deployment of such unmanned vehicles requires hiring costly support vessels to manage launch, umbilical cord, and recovery, as well as teams of dedicated and highly skilled ROV pilots.

This paper proposes an Intelligent Control Architecture (ICA) to enable multiple maritime vehicles to carry out autonomous multipurpose underwater intervention missions. The ICA is generic in nature but aimed at case study where an ASC and an IAUV are required to work cooperatively. They are capable of cooperating towards the execution of complex activities since they have different but complementary capabilities. The ICA proposed moves away from fixed mission plans and very elementary diagnostics scheme currently utilized to a more robust architecture to deal with the above missions. It needs to handle unexpected faults within vehicle, e.g. at the sensor and sensor processing levels, based on either hardware failure or environmental changes.

The architectural foundation to achieve the ICA lays on the flexibility of service-oriented computing. Each vehicle module provides basic services which advertise their capabilities to the system. The service also publishes regular updates of its current status.

In addition to the service orientation paradigm, a knowledge-based database captures the domain specific skills of the human expert (how to perform a specific task), as well as the dynamic information concerning the environment and platform capabilities. This makes it possible to include small atomic plans to test and validate the performance of specific services. The knowledge captured enables high-level reasoning agents to monitor, refine, or adapt mission plans based on the current situation. The resulting architectural solution proposed is a service-oriented agent-based approach which is suitable for integrating the vehicle modules as well as the capabilities of each marine vehicle in a collaborative manner. The agents are embedded in the marine vehicles. They are specialized in different disciplines, and provide different capabilities available as platform services (e.g. navigation, mapping, vision, etc.) to the overall system. The ICA is the backbone for the development of agents by providing a de-facto integration approach.

The combination of the above two technologies makes it possible to develop an ICA that is able to dynamically reconfigure and adapt itself in order to deal with changes in the operation environment. Thus, this ability to perform on-the-fly re-planning of activities when needed increases the chance to succeed in a given mission. The ICA performance is tested and demonstrated for a particular case study. However, it is a general solution for maritime autonomy that can be applied to other marine missions, and unmanned maritime vehicles. The reasoning is mainly involved at the high level tasks such as planning and re-planning to check the system consistency. However, there is no intention of reasoning on hard real time activities such as control of grasping/manipulation actions.

The rest of the paper is organized as follows. Section II presents the existing work related to this research. Section III shows an overview of the ICA. Section IV presents detailed aspects of the development of ICA. Section V describes the results obtained from simulation of a seabed survey in order to evaluate the ICA proposed for a particular fault situation. The last Section presents the conclusions and future research directions.

978-0-933957-40-4 ©2013 MTS

II. Literature Review

The state of the art presented in this paper focuses on the ocean engineering and robotics areas. In particular, it only involves mobile agent systems for marine vehicles. Addressing the above context, two main classifications can be made. The first one is according to the amount of agents implemented per vehicle. The second one is according to those approaches that rigorously follow the basic agent architecture and in particular, those which really propose approaches of intelligent agent architecture.

There are many proposals implementing several agents per individual mobile platform. These approaches generally assign an agent to a functional system module (navigator, pilot, vision processor, etc.). Thus, the agents are defined as key components of the infrastructure that supports the system [1] - [7]. An alternative approach some researchers have taken is to implement only one agent per mobile platform [8] - [13].

Researchers at the Center for Intelligent Systems Research (CISR) are working on decentralized control for multiple Autonomous Underwater Vehicles (AUVs). They proposed an ontological approach for collective behavior of intelligent agents that can be applied to IAUV fleets [12]. Their proposal is based on collective intentionality in agents. It is an alternative to the traditional Beliefs-Desires-Intentions model of individual agents that capitalized on the commitments agents had to one another rather than the commitments the agents had to maximize their own individual utility functions [14].

The Ocean Systems Lab at Heriot-Watt University has started working on situation awareness in service oriented agents for AUVs. The research mainly focuses on a semantic knowledge-based representation for improving situation awareness [15], a distributed ontological world model [16], an adapting mission planning [17], and an initial integrated proof of concept [18]. The above approaches pave the way towards reconfigurable control architectures for autonomous maritime vehicles based on service-oriented agents.

There are successful stories of underwater operations, in particular applications for formation control in environmental tracking [19]. In addition, defense systems are demanding autonomous air and marine vehicles as reported in a survey on networked multi-vehicle systems [20]. It shows, from a control systems viewpoint, some current issues common to networked multi-vehicle systems, and how they have been solved.

III. System Architecure Overview

This Section presents the architectural aspects of the ICA with an emphasis on key ICA elements and agent capabilities.

A. Key Architectural elements

Figure 1 shows the architectural dependency among the key elements of the ICA proposed. The central entity is the agent. In the top of the figure, the agent deals with missions with goals that can be divided into sub-goals, and a plan that can also be divided into sub-plans. On the right, the agent is endowed with capabilities that are required when planning. A capability is built of activity and functionality (or services).

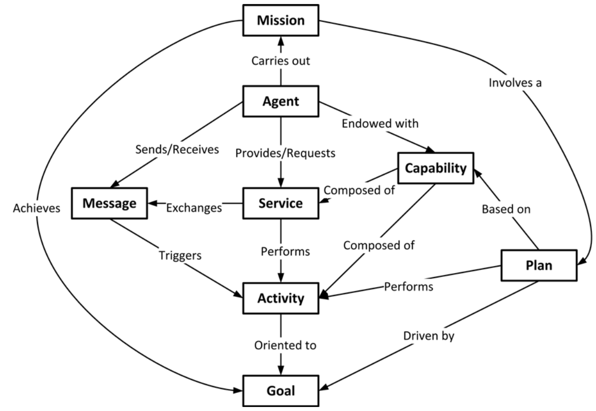

Figure 1. Elements dependency of intelligent control architecture.

A plan is based on agent capabilities, and performs a list of activities in Figure 1. The planning of activities is carried out by matching agent capabilities with required activities. On the left, an agent sends and receives messages that trigger activities. Services are able to exchange messages as well.

Figure 2 shows the operation principles of the Autonomous Marine Vehicle (AMV) system. This figure is explained by dividing it into two areas: the top part and bottom part. The former depicts how the system works at the planning level. The former depicts how the system works at the execution level. This figure presents the exiting connections between the planning and execution levels. The concept shows in this figure can also be applied to the internal operation of an agent. (Section V.C).

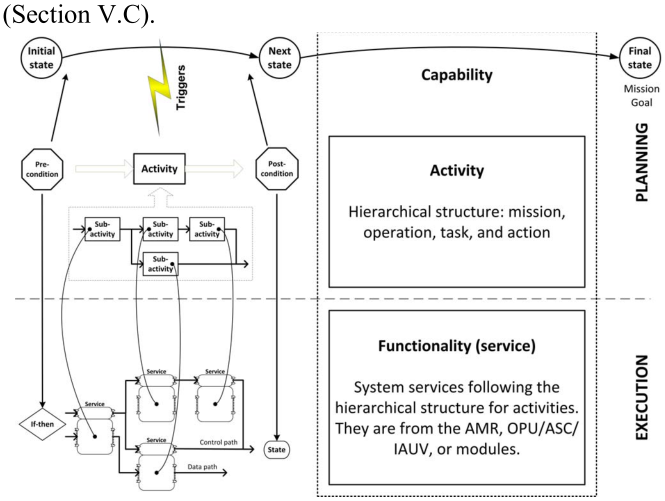

Figure 2. High-low-level agent integration.

At the left bottom of Figure 2, the network of platform services performs the activities required by the plan (the left top). There is a one-by-one relation between activities and services as shown in the figure. The activities are only triggered when pre-conditions are met. They also generate post-conditions. Pre-conditions are usually evaluated by “if-then” conditional sentences on states of data, and objects. Post-conditions normally result in new states of data, and objects that are used to evaluate the next pre-conditions. Goals are states, so every intermediate state reached can be considered as sub-goals achieved.

On the left of Figure 2 is a description of what a capability is, and the two levels it covers. At the right bottom, the network of platform services is the functionality that the system (AMV), subsystems (OCU, ASC, or IAUV), subsystem nodes, and node components provide. At the right top, the activities are hierarchically categorized as missions, operations, tasks, and actions. Thus, a capability is built of activities and functionalities (services).

B. Maritime Robot Capabilites

Capabilities are basically divided into two main parts: activity and functionality (service), and are triggered by messages. Activities are carried out by services. Thus, capabilities can be seen from a dynamic viewpoint; activities, and from a static viewpoint; services.

As capabilities are built of services, they also follow the classification for services as to hierarchical decomposition. Therefore, the ICA has two types of capabilities:

- Basic capability: They are indivisible. Therefore, they are the atomic elements of AMV platform in which the ICA is based on.
- Composite capability: They are composed by in other capabilities (basic or composite capabilities).

Following the above classification, capabilities are classified following the hierarchical categorization:

- A mission capability is composed of one or more Operation capabilities
- An operation capability is composed of one or more Task capabilities
- A task capability is composed of one or more Action capabilities
- An action capability are the atomic part of the capability hierarchy

The first three capability classifications are composite capabilities, and the last one is a basic capability. The mission and operation capabilities are designed by wrapping high-level functionalities and operability. The task and action capabilities are designed by wrapping low-level functionalities and operability that provide the platform components.

IV. SERVICE-ORIENTED AGENT ARCHITECTURE

This Section presents architectural aspects of the ICA design with emphasis on the service-oriented agent.

A. Overview of the Cognitive Concepts

There are a lot of literature and research works that describe the agent concept. The purpose of this Subsection is to briefly summarize the key aspects of the agent-based systems. This Subsection begins with some conceptual definitions within the robotics domain, in particular, for problem-solving processes that involve intellectual functions such as communication, planning, and reasoning. The following definitions are essential for the current development context (adapted from [21]).

Capability. System ability to effectively fulfill a specific activity (mission, operation, task, or action) in an environment.

Autonomy. System ability to autonomously make choices (select activity/ies) and enforce the decision made. This ability is accomplished under self-government and self-determination conditions.

Intelligence. System ability to accurately determine what activity will maximize the likelihood of achieving a goal successfully.

Knowledge. Data with meaning is information which in turn becomes knowledge when a purpose and the potential to generate action are attached. Knowledge is the intellectual machinery used to achieve goals (carrying out actions), and create new information.

Autonomy and intelligence are not the same. While autonomy makes reference to act or respond autonomously by making and enforcing decisions as solutions, intelligence provides multiple solutions for the same problem.

A robot walking randomly has autonomy in choosing direction and distance but could fall down a staircase if it selects a wrong activity. A robot that can successfully execute more complex activities may be more capable but it does not mean it is more autonomous. A robot doing wrong things successfully does not mean it has reduced capabilities but selecting wrong activity to be performed means reduced intelligence.

The above definitions do not mention explicitly what happen if faults are present in the system but they implicitly consider some failures generated by faults as well as any some problems that can cause the system is not able to successfully accomplish the mission given. Both issues are taken into account when the mission plans.

An agent is an autonomous computational entity that is able to interact within an environment. Its flexible behavior is due to three main types of activity-based working:

- Reactive behavior: the agent interacts with the environment by reacting when facing certain situations.
- Proactive behavior: the agent is able to be guided by its own goal in order to carry out different activities.
- Social behavior: the agent is able to communicate with other agents by sharing a common language.

In addition to the above behavioral issues, the agents of the ICA proposed are expected to be intelligent. This adds a fourth behavior type which is the following:

- Deliberative behavior: the agent has the capacity to perform intellectual functions such as planning, reasoning, communicating, etc. in order to be able to solve problem (e.g. how to achieve a mission goal) by making decision.

Based on the above types of behavior, an agent has several attributes such as autonomy, sociability, rationality, reactivity, proactivity, adaptability, mobility, veracity, and benevolence.

B. *Architectural Foundations*

The foundations of the ICA have multi-disciplinary nature. It comes from the robotics, cognitive science, and computer science. Therefore, the ICA development is based on the following architectural representations: robotic, cognitive, and agentic models. There are currently different reference models for each of the above representational descriptions. In particular, the ICA combines the following approaches.

A **robotic architecture** which is a hybrid approach composed of three-layer architecture (Planning, Sequencing, and Skill) plus a knowledge block; World Model. Figure 3 shows this combined architecture (top left).

A **cognitive architecture** built of two blocks: TBox and ABox which are part of the knowledge representation based on description logics.in Figure 3 shows the elements of this cognition process (top right).

An **agentic architecture** based on the Belief-Decide-Intention (BDI) software model. The agent structure is shown in Figure 3 (bottom). It is built of well-defined blocks, i.e. Belief, Desire (goal), and Intention blocks. Additionally, there are Interpreter and Plan blocks.

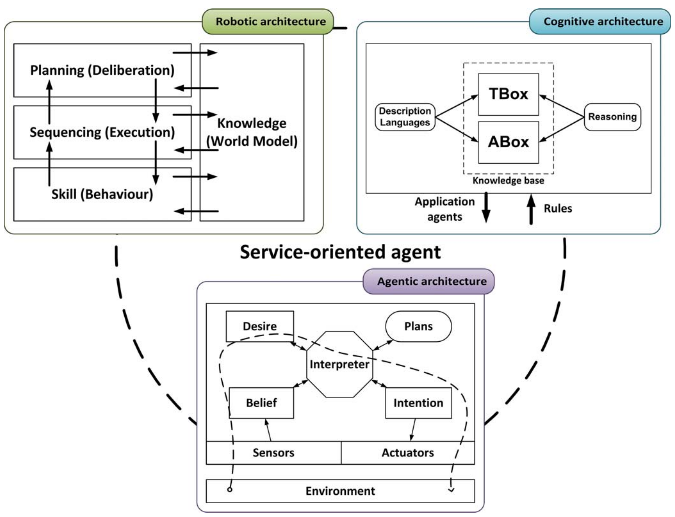

Figure 3. Architectural drivers for the agent structure.

Situation Awareness (SA) is the ability to be aware of and understand what is happening in the surroundings of an agent, both at the present time, and in the future through prediction. This capability allows systems to understand dynamic and complex environments, and operate with them. It can be divided into three separate levels: perception of the elements in the environment, comprehension of the current situation, and projection of future status. SA involves the events, states, condition, and activities of the environment dynamics as to time and space from which some situations arise (in particular those changes that occurred in the environment over some time interval). A situation is defined by a specific state after a sequence of events (with intermediate states, and activities with pre and post conditions). The situation is concerned with the comprehension of the environment features, and with the evolvement of these features over time [18].

SA is essential for decision makers. Within an agent, the decision making cycle is defined by four basic stages: Observation-Orientation-Decision-Action (OODA) loop. The Observation stage is the SA perception level. The Orientation stage takes into account the information acquired from the Observation stage and the knowledge of the agent, to understand the situation (SA comprehension level). The Decision stage is carried out at the SA projection level. The Action stage closes the OODA loop by carrying out actions according to the environmental adaption made in the previous stage.

The mapping of the SA and OODA concepts onto the BDI agent architecture is as follows. The Belief block represents the informational state of the agent, and describes the known state of the world (the world model). It matches the SA perception and comprehension levels or OODA observation and orientation stages. The Desire block represents the motivational state of the agent (goals or situations that the agent would like to accomplish). The Intention block represents the deliberative state of the agent (what the agent has chosen to do). It corresponds to the SA projection level or OODA decision and action stages. The Interpreter block maps to the agent reasoner.

The above approach endows the agent with initiative. Decision making mechanisms are critical for problem-solving processes that are preformed every time an agent receives a mission to be carried out.

C. *Agent Anatomy*

The agent anatomy is depicted in Figure 4. It shows the internal structure of the service-oriented agent. There is one agent per marine vehicle. This figure encompasses three architectural models mentioned above.

(1) A block-layered robotic model as shown in Figure 3 with the following blocks: units of planning (deliberation), sequencing (execution), skill (behavior), and a world model. In addition, a user interface is taken into account.

(2) A description-logics model as shown in Figure 3 which involves the following blocks: deliberation unit (mission reasoner), and world model (mental model; ontology).

(3) A model with the logical structure of BDI agents as shown in Figure 3: beliefs, desires, interpreter, intentions, and plans.

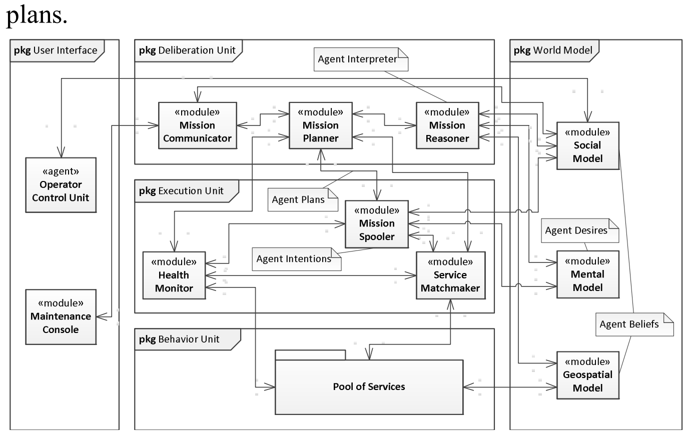

Figure 4. Agent anatomy.

User interface. The end user is able to deal with the mission, and visualize the mission results through an Operator Control Unit (OCU), e.g. seabed map (image mosaicking), scene and objects characterization, etc.

Deliberation Unit. This has basically three components: the mission communicator, the mission planner and the mission reasoner. The mission communicator, which includes the communication manager (wired and wireless communication channels), communicates with the human operator and with the marine vehicles through the social model. The mission planner, which includes the resource manager, helps to selects the agent capabilities required to take actions according to the decisions made by the agent interpreter. The mission reasoner, which includes the agent interpreter, reads the data perceived from sensors, interprets them according to the knowledge embedded in the mental model, and makes the decision of what to do next. The mission planner output is a plan (list of activities to be carried out by the spooler).

Execution Unit. This is in charge of dealing with the execution of the agent services. The execution is according to the plan generated in the mission planner, and it is executed by the mission spooler. The mission spooler is responsible for parceling out activities listed in the mission plan for execution by platform services. The health monitor deals with the status of the platform services by keeping record of the vital working conditions. It implements the fault diagnosis techniques.

Behavior Unit. The pool of services of the agent depends on the marine vehicle it is deployed on. They are services at the vehicle level. In the case of the ASC the services provided are: navigation, behavior management, waypoint list setting, and acoustic/radio communication. In the case of the IAUV the services provided are: navigation, path plan setting, maps generation, seabed data collection, scene/object identification, visual docking, manipulation, grasp specification, and acoustic/radio communication.

World Model. This is a central repository built from the following models. (1) *Social Model.* It describes the social context which the agent inhabits and interacts with. It is built of the agent directory module which includes the service registry. (2) *Mental Model.* It describes what the agent is able to know about itself. (3) *Geospatial Model.* This contains environmental data collected by sensors (perception).

V. EXPERIMENTAL RESULTS

This Section describes the computer-simulated experiment (seabed survey) carried out to evaluate the ICA proposed.

A. *Computer Simulation*

The simulator and the agents are implemented in Robot Operating System (ROS) [22]. The seabed survey experiment only focuses on the guidance, navigation and control of the IAUV for the seabed survey mission. The exploration area is 50 m x 40 m (beginning at 30 m north, 30 m east). The seabed survey stages defined at the group-of-vehicle and vehicle levels (similar stages can be defined at the device and transducer level inside the vehicles) are as follows:

First Stage: the ASC and IAUV modules publish their capabilities as services available in their respective platforms. The advertisement and discovery of services are carried out by means of the service matchmaker. The matchmaker is designed to be used to match service providers to direct consumers of those services. It is the responsibility of each service consumer to use the matchmaker to resolve its own direct service dependencies. The IAUV module capabilities implemented for the simulation of the target intervention are: Path planning and Terrain following as operation services, and IAUV navigation, and IAUV motion control as task services.

Second Stage: The mission planner checks the plan consistency (based the capabilities required to carry out the mission) against the record kept by the matchmaker. The platform services are basically available or not available. When they are available, the mission planner must check their health status in order to know if he can really make use of them. If there is any problem to execute the services or if they are unavailable, the service matchmaker proceeds to find any other capability that can replace the required one. If no capability are available at all, the service matchmaker must decide what to do the plan (if it is still viable or not), and communicates to the rest of the system (AMV).

Third Stage: once the availability of the required services is confirmed, the IAUV is ready to begin the mission. The mission spooler retrieves all the information needed to execute the services from the matchmaker, i.e. based on the service name, the mission spooler makes a query for status, and invocation method for each service in order to create the execution queue. The mission spooler then invokes the platform services according to the mission plan, and takes into account the health of the services. The IAUV, initially located at the origin position (surface), goes down to a given altitude where it reaches the beginning position to start the data collection from the seafloor. Then, it follows the path given so visual or acoustic data can be collected. When the IAUV reaches the end of the path (end position) for the exploration area, it returns to the surface (origin position) with the stored data. Fig. 3 shows the path followed by the IAUV after being commanded to perform the activities for a seabed survey.

Fourth Stage: once the seabed survey is finished, it can be carried out again. The IAUV will be endowed with more mission, operation, task, and action services in order to include more capabilities.

B. *Experimental Results*

The seabed survey stages simulated are as described above. In addition, a fault is introduced on a service; the IAUV leader following service. It shows the greatest potential of the research contribution of ICA since the ASC and IAUV are able to make in-mission decisions (without contacting or getting back to the human operator for advice on what to do).

The AMVs, by themselves, are capable of dealing with abnormal situations based on the knowledge stored in the ontological database (the operator skills, platform capabilities and, possible changes in the environment).

Figure 5 shows the behavior results obtained in simulation of the seabed survey with the path followed by the two vehicles (ASC and IAUV) when the IAUV leader following service stop working (68 m in the north position, 40 m in the east position). The above service cannot be recovered and there is not a similar capability to replace faulty one. Both vehicles come back to the beginning position (30 m north, 30 m east), then they come back to the origin position (0 m north, 0 m east); mission aborted.

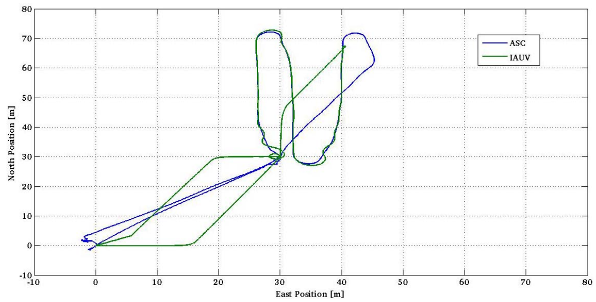

Figure 5. Execution of services for seabed survey without fault recovery.

Figure 6 shows the behavior results obtained in simulation of the seabed survey with the path followed by the two vehicles (ASC and IAUV) when the IAUV leader following service stop working (70 m north, 40 m east). In this case, the leader following service does recover itself (or there is a similar capability to replace faulty one). Both vehicles come back to the beginning position (30 m north, 30 m east), then they start the seafloor data collection again at the beginning since the leader following capability is recovered. Finally, they complete the mission given (30 m north, 73 m east) by returning at the origin position; mission successful.

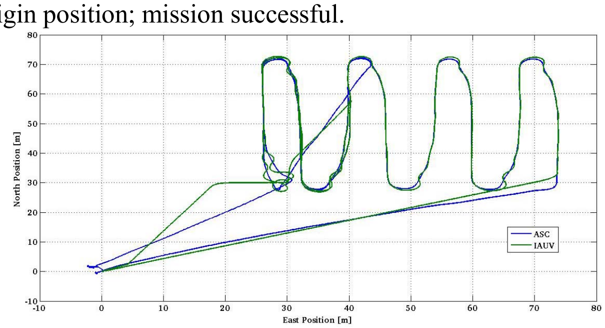

Figure 6. Execution of services for seabed survey with fault recovery.

VI. CONCLUSIONS

An Intelligent Control Architecture (ICA) for autonomous marine vehicles has been presented. The ICA foundation lies on the infrastructure of service-oriented computing, and agent technology. The approach also improves the situation awareness by relying on agents that combine their knowledge and skills with the environmental information. This make it possible to achieve adaptive and reflective mission planning based on a dynamic reconfiguration of plans. A demonstration of the ICA performance in a scenario including a fault has also been presented. There is still investigation to be done into this promising ICA approach. Future next steps of this research will take into account more complex evaluation scenarios, including more potential faults and unexpected situations.

REFERENCES

[1] L. Molnar, and S. M. Veres, “System Verification of Autonomous Underwater Vehicles by Model Checking”, in Proc. of the OCEANS, Bremen, Germany, 2008.

[2] Z. Yan, and S. Hou, “A Coordinated Method Based on Hybrid Intelligent Control Agent for Multi-AUVs Control”, in proc. of the Int. Conf. on Information Acquisition, Weihai, Shandong, China, 2006.

[3] J. Borges de Sousa, and F. Lobo Pereira, “Real-time Hybrid Control of Multiple Autonomous Underwater Vehicles”, in Proc. of the 37th IEEE Conf. on Decision and Control, Tampa, FL, USA, 1998.

[4] T. Y. Teck, M. Chitre, P. Vadakkepat, “Hierarchical Agent-based Command and Control System for Autonomous Underwater Vehicles”, in proc. Int. Conf. on Autonomous and Intell. Systems, Singapore, 2010.

[5] J. P. Pignon, and C. Bizingre, “Multiple Agents Architecture for Intelligent Command and Control System of AUVs: Application to the MARIUS Vehicle”, in proc. OCEANS, vol. 3, pp. III/126 – II/131, 1994.

[6] T. R. Cuff, and R. W. Wall, “Support Platform and Communications to Manage Cooperative IAUV Operations”, in Proc. OCEANS Asia Pacific, 2007.

[7] P. Patrón, E. Miguelañez, J. Cartwright, and Y. Petillot, “Semantic Knowledge-based Representation for improving situation awareness in service oriented agents of autonomous underwater vehicles”, Proc. OCEANS Europe, 2009.

[8] C. C. Sotzing, J. Evans, and D. M. Lane, “A Multi-Agent Architecture to Increase Coordination Efficiency in Multi-IAUV Operations”, in Proc. OCEANS Europe, pp. 1-6, 2007.

[9] P. Szymak, and T. Praczyk, “Control Systems of Underwater Vehicles in Multi-agent System of Underwater Inspection”, in Proc. 11th WSEAS Int. Conf. on Automatic Control, Modelling and Simulation, Vouliagmeni, Athens, Greece, 2009.

[10] H. Li, A. Popa, and C. Thibault, “A Software Framework for Multi-Agent Control of Multiple Autonomous Underwater Vehicles for Underwater Mine Counter-Measures”, in Proc. Int. Conf. on Autonomous and Intelligent Systems, Povoa de Varzim, Portugal, 2010.

[11] T. Fujii, and F. Ura, “Autonomous Underwater Robots with Distributed Behavior Control Architecture”, in Proc. IEEE Int. Conf. on Robotics and Automation, Nagoya, Aichi, Japan, 1995.

[12] X. Q. Bian, T. Chen, J. Zhou, and Z. Yan, “Research of Autonomous Control Based on Multi-Agent for IAUV”, in Proc. Int. Workshop on Intelligent Systems and Applications, WuHan, China, 2009.

[13] C. McGann, F. Py, K. Rajan, H. Thomas, R. Henthorn, R. McEwen, “T-REX: A Model-Based Architecture for AUV Control”, in proc. Int. Conf. on Automated Planning & Scheduling, Rhode Island, USA, 2007.

[14] P. Ray, M. O’Rourke, and D. Edwards, “The Ontological Status of Autonomous Underwater Vehicle Fleets”, in Proc. OCEANS MTS/IEEE Biloxi, Biloxi, Mississippi, USA, 2009.

[15] P. Ray, M. O’Rourke, and D. Edwards, “Using Collective Intentionality to Model Fleets of Autonomous Underwater Vehicles”, in Proc. of Oceans MTS/IEEE Biloxi, Biloxi, Mississippi, USA, 2009.

[16] J. Cartwright, P. Patrón, J. Evans, and D. Lane, “Developing a Distributed Ontological World Model for Autonomous Multi Vehicle operations”, 2nd SEAS DTC Technical Conf., Edinburgh, UK, 2007.

[17] P. Patrón, and D. M. Lane, “Adaptive Mission Planning: the Embedded OODA Loop”, 3rd SEAS DTC Technical Conf., Edinburgh, UK, 2008.

[18] C. C. Insaurralde, J. J. Cartwright, Y. R. Petillot, “Cognitive Control Architecture for Autonomous Marine Vehicles”, in proc. IEEE Int. Systems Conf., Vancouver, BC, Canada, 2012, pp. 117-124.

[19] D. G. Roberson, “Environmental Tracking and Formation Control for an Autonomous Underwater Vehicle Platoon with Limited Communication”, Ph.D. Thesis, State University, VA, USA, 2008.

[20] A. R. Girard, J. Borges de Sousa, and J. K. Hedrick, “An Overview of Emerging Results in Networked Multi-Vehicle Systems”, in Proc. 40th IEEE Conf. on Decision and Control, Orlando, FL, USA, 2001.

[21] J. P. Gunderson, and L. F. Gunderson, “Intelligence ≠ Autonomy ≠ Capability”, Gamma Two, Inc, Performance Metrics for Intelligent Systems, Gaithersburg, MD, 2004.

[22] Robot Operating System (ROS), available in http://www.ros.org.