ADVOCATE II: ADVanced On-Board Diagnosis and Control of Autonomous Systems II

Miguel Angel Sotelo[1], Luis Miguel Bergasa[1], Ramón Flores[1], Manuel Ocaña[1], Marie-Hélène Doussin[2], Luis Magdalena[3], Joerg Kalwa[4], Anders L. Madsen[5], Michel Perrier[6], Damien Roland[7], and Pietro Corigliano[8]

[1] University of Alcalá, Escuela Politécnica. Campus Universitario s/n, Alcalá de Henares, 28871, Madrid, SPAIN.
michael@depeca.uah.es,
http://www.depeca.uah.es
[2] GETRONICS, Europarc bat D, Technopôle de Chateau-Gombert, 13013 Marseille (France)
[3] ETSI Telecomunicación, Universidad Politécnica de Madrid (UPM) Madrid 28040 (Spain)
[4] STN-ATLAS Electronik GmbH, Sebaldsbrucker Heerstraße 235, 28305 Bremen (Germany)
[5] Hugin Expert A/S, Niels Jernes Vej 10, 9220 Aalborg (Denmark)
[6] IFREMER, Zone portuaire de Bregaillon, BP 330, 83597 La seyne-sur-mer (France)
[7] E-MOTIVE, Marseille (France)
[8] INNOVA S.p.A., Via della Scrofa 117, 00186 Rome (Italy)

Abstract. A way to improve the reliability and to reduce costs in autonomous robots is to add intelligence to on-board diagnosis and control systems to avoid expensive hardware redundancy and inopportune mission abortion. According to this, the main goal of the ADVOCATE II project is to adapt legacy piloting software around a generic SOAP (Simple Object Access Protocol) architecture on which intelligent modules could be plugged. Artificial Intelligent (AI) modules using Belief Bayesian Networks (BBN), Neuro-Symbolic Systems (NSS), and Fuzzy Logic (FL) are coordinated to help the operator or piloting system manage fault detection, risk assessment, and recovery plans. In this paper, the specification of the ADVOCATE II system is presented.

1 The ADVOCATE II Architecture

ADVOCATE II introduces intelligent techniques for diagnosis, recovery and replanning into UUVs (Unmanned underwater vehicles) and UGVs (Unmanned Ground Vehicles). The global objective of the project is to enhance the level of reliability and efficiency of autonomous robotic systems, as described below:

– To construct an open, modular, and generic software architecture for autonomous robotic systems diagnosis and control.

R. Moreno-Díaz and F. Pichler (Eds.): EUROCAST 2003, LNCS 2809, pp. 302–313, 2003.

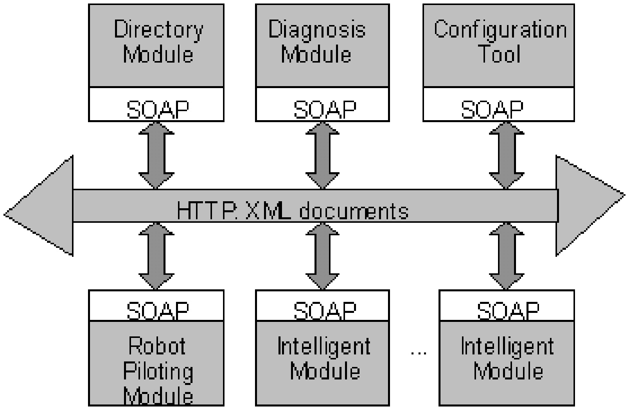

Fig. 1. The ADVOCATE II Architecture

- To develop or improve a set of intelligent diagnosis modules fully compatible with this architecture and tested in operational applications.
- To carry out practical tests and demonstrations on a set of operational prototypes in order to prove operationality and efficiency of this solution in several application fields, and particularly for Autonomous Underwater Vehicles (AUVs) and Autonomous Ground Vehicles (AGVs).

ADVOCATE II is based on a distributed architecture, and a generic protocol (SOAP/XML technology implementing HTTP) for communication between the different modules. The ADVOCATE II architecture is distributed around a SOAP bus as depicted in figure 1. The architecture is modular, easy to evolve and to adapt to legacy piloting systems [1] [2]. It comprises five different types of modules.

1.1 Robot Piloting Module (RPM)

This module manages the mission plans and communicates directly with the vehicle sensors and actuators. Several RPMs, each of them working on a specific subsystem, can be plugged onto the ADVOCATE II architecture. Each end-user participating in the project (UAH, STN ATLAS, and IFREMER) is responsible for the corresponding piloting modules. A brief description of the different Robotic platforms used in the ADVOCATE II project is provided below.

IFREMER Vehicle Platform and Robot Piloting Module. The application deployed by IFREMER is based on an experimental underwater vehicle, called VORTEX, operated in a test pool or in simulation. The vehicle is a small experimental Remotely Operated Vehicle (ROV), but it can be considered as an AUV from a control point of view since fully automatic missions can be programmed and performed. VORTEX can be considered as a vehicle that belongs to the class of working AUVs, not dedicated to long survey tasks, but to

Fig. 2. VORTEX vehicle

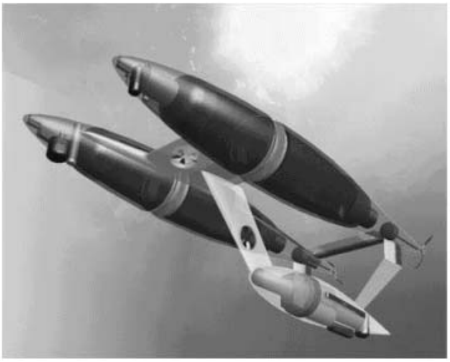

Fig. 3. DeepC Vehicle

intervention tasks (offshore application for instance). Mechanically, the vehicle structure consists of a basis tubular structure on which the different actuators are arranged, without pre-defined locations, as depicted in figure 2. Central to this structure is the main electronics package containing the vehicle electronics as well as the different set of sensors : attitude, depthmeter, gyrometers, video camera, sonar and sounders.

The control architecture of VORTEX is located on a VME-based system, reachable through Ethernet, where different software modules can be connected to the vehicle controller:

- Man-Machine Interface used to supervise and operate the vehicle.
- Mission Manager used to program simple or complex missions.

STN-ATLAS Vehicle Platform and Robot Piloting Module The DeepC vehicle developed under the support and promotion of the Federal Ministry of Education and Research of Germany is a fully autonomous underwater vehicle (AUV), depicted in figure 3, with the related components on the water's surface for oceanographic and oceanologic applications.

One of the outstanding features of the AUV is the "reactive autonomy". This property allows situation-adapted mission and vehicle control on the basis of

Fig. 4. UAH platform: BART (Basic Agent for Robotic Tasks) robot.

multi-sensor data fusion, image evaluation and higher-level decision techniques. The aim of the active and reactive process is to achieve high levels of reliability and safety for longer underwater missions in different sea areas and in the presence of different ground topologies. The processes involved include:

- highly accurate long-term underwater navigation
- autonomous obstacle recognition and avoidance
- autonomous operation monitoring system (system diagnostics)
- reactive mission management system
- case-sensitive track control
- situation-adaptive vehicle controller
- global and local communication and data management

Whithin Deepc Control Architecture capabilities, the monitoring system is of particular significance. It divides into mission monitoring and health monitoring. The mission monitoring feature is responsible for the mission sequence. It is used to analyse running missions and gives recommendations for any necessary replanning to mission control.

UAH Vehicle Platform and Robot Piloting Module. In the context of the ADVOCATE II project, UAH deploys a ground vehicle that works in a combination of autonomous and teleoperated mode. The vehicle is intended to perform surveillance tasks after hours in a large building composed of corridors, halls, offices, laboratories, etc. For this purpose, UAH is currently deploying the BART (Basic Agent for Robotic Tasks) robot, depicted in figure 4. The operator is in charge of global vehicle navigation by remotely commanding its actuators according to the images that are continuously transmitted through a wireless ethernet link from the vehicle to the base station. Information concerning proximity sensors (the vehicle is equipped with a ring of ultrasound sensors) is also transmitted for monitoring.

A key variable for energy monitoring and diagnosis is the estimated State of Charge (SOC) of the vehicle battery. SOC estimations are based on battery voltage measurements [6]. In order to provide stable SOC estimations, a kalman filter is used so as to remove gaussian noise in voltage measurements, according to the simplified battery model provided in [3] [4] [5], described by equation 1.

$$\begin{aligned} \begin{bmatrix} V_{c_k} \\ R_k \end{bmatrix} &= \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_{c_{k-1}} \\ R_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{-T_s}{C} \\ K_r \end{bmatrix} I_{bat_{k-1}} + \begin{bmatrix} w_{1_{k-1}} \\ w_{2_{k-1}} \end{bmatrix} \\ V_{bat} &= \begin{bmatrix} 1 & -I_{bat_k} \end{bmatrix} \begin{bmatrix} V_{c_k} \\ R_k \end{bmatrix} + v_k \end{aligned} \tag{1}$$

where I_{bat} and V_{bat} represent the current and voltage of the battery respectively, R stands for the internal battery resistance, C is the battery capacity, w_1 and w_2 represent the components of the process noise, v stands for the measurement noise, and K_r is a battery dependant parameter that describes the relation between current and voltage. After that, the SOC is computed using the model proposed in [6], as shown in equation 2.

$$\begin{aligned} V_{avg} &= \frac{V_{c_{k-1}} - V_{c_k}}{2} \\ R_{avg} &= \frac{R_{k-1} - R_k}{2} \\ V &= V_{avg} - I \cdot R_{avg} \\ r &= \left(\frac{V_{avg}}{2 \cdot R_{avg}} \right)^2 - \frac{P_{bat}}{R_{avg}} \\ I &= \frac{V_{avg}}{2 \cdot R_{avg}} - \sqrt{r} \\ SOC_k &= SOC_{k-1} - \frac{P \cdot \Delta t}{3600 \cdot C \cdot V} \end{aligned} \tag{2}$$

where P_{bat} is the rated battery power, P denotes the instant power, and Δt represents the time interval in the estimation process.

1.2 Decision Module (DM)

The Decision Module can be considered as the central unit of the ADVOCATE system architecture. The Decision Module needs to have a good knowledge about the whole system. That is why it collects information about all system modules during initialization. The decision module has a generic part and a specific part containing knowledge for decision making, according to diagnosis results. The Decision Module manages the overall diagnosis and recovery process, including control of the monitoring/diagnosis/recovery process, validation of diagnosis and recovery actions (if needed), interaction with human operators (if any) with regards to diagnosis and recovery, integration of uncertainty information provided by the intelligent modules, and conversion of the recovery actions into recovery plans, and subsequent processing of the recovery plans. It is made up of four functional blocks:

- Diagnosis Process Manager. This subsystem is in charge of managing all the interactions related to diagnoses and recovery, from a procedural point of view.
- Initialization Module. This subsystem gathers all actions that take place during initialization. This module is in charge of one action, which does not take place during initialization (answer life checking). This action is related to registration.

- Measures Integrator (Decisions Merger). As there could be several diagnoses or recovery actions attending a single request, once these different results get to the decision module they have to be integrated.
- Recovery Action to Recovery Plan Translator. It is necessary to translate the recovery action into a series of manoeuvres (a recovery plan). The task of this subsystem is to translate a recovery action (a result generated by the intelligent modules) into a recovery plan that can be executed by the piloting module. This module is internal acting upon request of the Diagnosis Process Manager.

The core of the module from a procedural point of view is the Diagnosis Process Manager. This block reacts once an alarm is received or once the operator launches a diagnosis, a monitoring, or a recovery process. From that point, it controls the whole process, by determining when, how, and what module (Intelligent Module, Robot Piloting Module, or MMI) to ask for information, and continuing the process until the final recovery plan is provided to the Vehicle Piloting Module.

1.3 Intelligent Modules

Several Intelligent Modules for each End-User application are currently being developed, using different Artificial Intelligent techniques devoted to solve real problems on operational robots by making use of specific knowledge on them. Intelligent Modules include functionalities providing a diagnosis (identification of sub-system state), a proposed recovery action, or both. The main role of the intelligent modules is to serve as efficient modules for solving diagnosis and recovery action problems. Each intelligent model plugged onto the ADVOCATE II system architecture consists of a generic part and a problem specific part. The problem specific part will typically consist of the knowledge base (neural network, rule base, or Bayesian belief network) to be used to solve the diagnosis and recovery action problem and possibly "bookkeeping" functionality such as data pre- and post-processing. The generic part is the interface and the functionality common to all intelligent modules. An intelligent module may have two different modes of operation such as an off-line mode and an on-line mode. In on-line mode an intelligent module uses its knowledge base to process requests to produce diagnoses and recovery actions. In off-line mode the knowledge base to be used in the on-line mode is constructed, updated, or revised by learning, for instance. The functionality of the off-line mode will depend on the particular technique used in the intelligent module whereas the functionality of the on-line mode is defined by the interface of the generic intelligent module. The present implementation comprises modules based on:

- Bayesian Belief Networks (BBN).
- Fuzzy Logic (FL).
- Neuro-Symbolic Systems (NSS).

1.4 Directory Module

The Directory Module is a central point of the architecture. Developed in Java, it implements part of the UDDI registry. It registers dynamically all the modules plugged on the architecture, provides on request relevant module URLs, checks regularly if the registered modules are still operational, and in case some of them is not any more, informs the others. All the inter-modules communication is based on the SOAP standard, because of its flexibility and lightness. However, in order to support the soft real time constraints of such technical application a complete middleware has been developed including implementation of SOAP extensions to manage timeouts and priorities. This middleware provides the developers of the different modules all the "communication material" for an easy implementation of the communication interfaces. In addition, an Integration Tool and a Test Tool have been developed to provide support also during integration and test phases.

1.5 Configuration Tool

This is an offline, user-friendly application, which eases the production of the XML File describing all the communication interfaces to be developed and that constitutes an input for the SOAP middleware, as well as the XML Configuration File for every modules of the ADVOCATE II system, including the different necessary parameters. By generating a graphical view of the system the user will be able to check the concordance of the configuration files, and to foresee the behaviour of the modules in the system. All this set of tools has been designed to reduce at maximum and make easier the work to adapt ADVOCATE to a new or existing system.

2 Applications

Three end-users are involved in the ADVOCATE II project:

- IFREMER (France) designing Autonomous Underwater Vehicles (AUVs) for scientific applications.
- STN ATLAS Elektronik (Germany) designing Autonomous Underwater Vehicles (AUVs) and semi-AUVs for industrial applications.
- UAH (Spain) designing Piloting Modules for either Autonomous or Remotely Operated Ground Vehicles (AGVs or ROGVs, respectively) for surveillance applications.

2.1 IFREMER Application

Within the ADVOCATE II project, VORTEX will be considered as a working AUV, i.e. without human supervision during a mission execution. Thus, it must be demonstrated that the intelligent diagnostic architecture to be developed in the project can handle automatic problem resolution, without any guidance by an operator during a mission execution. The different Diagnosis Problems that need to be solved with ADVOCATE are the following:

– Thruster malfunctioning diagnosis
– Energy consumption monitoring

Diagnosis Problem: Thruster Malfunctioning. The VORTEX vehicle is controlled thanks to several thrusters. The controller commands *rpm* can help to perform some simple diagnosis. However, in some situations (damage to the propeller for instance) the vehicle controller will not detect the failure. In the case of thruster failure, the vehicle is no longer controllable as planned. Recovery actions have then to be applied in order to save the vehicle or to prevent the damaging of the environment. The diagnosis may have to be performed by doing specific test manoeuvres in order to get more information on the vehicle behaviour. As a function of the evaluated diagnosis, adaptation of the vehicle controller or the mission objectives may be envisaged. The following data are steadily monitored :

– Current consumption of all thrusters
– commanded *rpm*
– global navigation data of the vehicle

Diagnosis Problem: Energy consumption monitoring. The energy consumption monitoring is mandatory for an AUV since it will directly influence the correct execution of the mission and the safety of the vehicle. Any abnormal energy consumption should be detected and reported. The corresponding recovery actions in this situation will be to determine if the programmed mission can be continued and terminated (maybe by simply reducing the speed) or if some mission plan modification is requested. Data available for assessment are:

– the initial amount of energy
– the current consumption of energy

2.2 STN-ATLAS Application

Concerning DeepC AUV, three main diagnosis problems are stated:

– Assessment of Global Vehicle Behaviour (Motion Analysis)
– Propulsion and Thruster Diagnostics
– Sensor Malfunction

Diagnosis Problem: Global Vehicle Behaviour. An Autonomous Underwater Vehicle has to perform a long endurance mission about 60 hours or more. During that time it may be possible that the behaviour of the vehicle is not as it is expected in respect of the control input. For instance it shows a tendency to turn left although all controls are in a neutral position. Because of the fact that DeepC is fitted with an adaptive controller, it may be more likely that a control input necessary for a wanted behaviour is uncommon. The first aim of Advocate will be a monitoring and assessment of the motion characteristic and the control inputs. After the analysis of the measured behaviour an appropriate Recovery Action has to be proposed. The following input Data can be provided by the AUV system:

- Thruster Diagnosis
- Navigation Data: speed vector (North, East, Down), position (Latitude, Longitude, depth)
- Controller Data: flap angle, thruster *rpm*

The behaviour of the vehicle can be assessed by comparison with a numerical model or by a knowledge-based description or both. Using a model for comparison a high quality model must be implemented and it takes a huge amount of processing. A rule- or expert-knowledge based comparison is faster, maybe more robust, but does not comprise all vehicle states. Further it will be difficult to validate over the complete envelope of motion capability. But because DeepC will run 90 % of its mission time straight ahead it may be sufficient to regard this vehicle state. In this case the knowledge-based description is prefered.

Diagnosis Problem: Thruster and/or Actuator Malfunction. Propulsion and steering of the DeepC AUV is achieved by propellers (thrusters) and rudders (flaps). As mechanical devices they are prone to getting entangled with floating obstacles. In worst case parts may dismantle due to faulty installation or external forces. The vehicle controller will not detect a possible failure. In a possible failure case the vehicle is no longer controllable. It is absolutely necessary to perform recovery actions to save the vehicle or to prevent the damaging of the environment. Failures may vary in their significance: with increasing amount of foreign substances between (or clamped to) moving parts the more additional torque on the actuators will be generated. The following data are steadily monitored at a time interval *dt* or on request of the Motion Analysis Diagnosis or on request of a check module:

- Current consumption of all motors
- commanded *rpm* or angle value
- status of controller

From this data the health status of the thruster/actuators are generated by comparing current values with ideal values. The following Diagnosis is expected:

- Actuator ok, entangled, or lost
- Motor/controller damaged

Diagnosis Problem: Sensor Malfunction. As a fully autonomous system, DeepC has to rely on its sensors to survive operational. Among the most important sensors are:

– the Inertial Measurement Unit to measure the accelerations of the vehicle
– the CTD Probe to measure conductivity, temperature and depth
– the obstacle avoidance Sonar

It is possible that failures occur so that the values, the sensors currently reporting, are corrupted or too noisy. A special problem of the sonar is, that possible obstacles may be hidden in a cluttering environment. This is probably true for an operation near the seafloor. In this case images cannot be analysed automatically because the results of the image processing algorithms are poor. There is a risk of damaging the vehicle by an unforeseen collision. It can be considered to have a poor image whenever:

– The distance to the ground/surface is small so that a high portion of bottom reverberation is present, or
– High Seastate or other broad band noise or sound sources (e. g. ship noise) are in the vicinity of the sonar receiver

2.3 UAH Application

During the execution of the mission, several problems or failures can occur yielding the vehicle to abnormal behaviour with respect to the expected one, and thus, impeding proper finalisation. ADVOCATE II will be used to do diagnosis and/or recovery actions for those failures, using the modular and intelligent diagnosis systems developed in this project.

Diagnosis Problem: Energy problem. The vehicle electrical energy can run out during operation impeding the complete finalisation of the mission. That could become a serious disadvantage particularly if the vehicle is operating far away from the base station, or if several vehicles are simultaneously being remotely monitored by just one operator. From continuous monitoring of the vehicle power consumption along the planned track and prediction of the remaining power, the likelihood of mission success can be forecasted. In case the predicted power consumption exceeds the available battery capacity (plus some reserve), the mission conditions should be accordingly replanned. Then, the battery State Of Charge (SOC, hereinafter) requires continuous monitoring so as to envisage the most appropriate action to take, according to the remaining vehicle energy. For making the decision upon Energy problems, the Intelligent Module (based on Bayesian Belief Networks) in charge of that task will be supplied with the following Basic Information.

– Initial SOC (percentage over full nominal charge)
– Planned mission and prediction of power requirements
– Speed profile

Diagnosis Problem: Actuator malfunction. If at any time during the mission some failure in the drive actuators arises (malfunction of motors, soft obstacles in the axes, sliperages, etc), mission execution and proper finalisation will not be possible any longer, or at least, vehicle safety and stability would be compromised. An appropriate diagnosis for correct detection of this type of faults is mandatory in order to start some recovery action to compensate for the vehicle failure, or to proceed to execute additional test manoeuvres so as to gather as much information as possible to get further diagnosis. Whenever the vehicle piloting module detects some actuator problems it sends an alarm to the decision module which is in charge of providing appropriate diagnosis and/or recovery actions. For making the decision upon actuator failures, the Intelligent Module (based on Neuro-Simbolic Systems) in charge of that task will be supplied with the following Basic Information.

- Vehicle Dynamic Model
- Commanded actuation
- Current vehicle state

Diagnosis Problem: Sensor related motion problems. Ultrasound sensorial information is the main input environment data used for navigation and thus it works as a safety system for collision avoiding in either the autonomous and the teleoperated modes. Nonetheless, there are some obstacles (short height obstacles, indeed) that can not be detected by ultrasound means. In case the AGV finds an obstacle of this type along its way, the vehicle will inevitably collide with the obstacle, which can even be dragged. In order to recover from these situations, the vehicle piloting module will carry out a first coarse detection of the problem intended to provide an alarm to the Decision Module. Upon the alarm, the Decision Module will require diagnosis on the problem to the appropriate Intelligent Module (based on Fuzzy Logic). If the diagnosis provided by the FL IM confirms the occurrence of the problem, a recovery action will be generated so as to get the vehicle rid of the obstacle and resume the mission afterwards. The Vehicle Piloting Module provides the folllowing information concerning real and measured data:

- Battery voltage and consumption
- Commanded and angular velocity of left wheels
- Commanded and measured angular velocity of right wheels
- Commanded and measured linear velocity
- Ultrasound range measures

3 Conclusions

The main objective of the ADVOCATE II project is to develop a software architecture to allow the implementation of intelligent control modules for underwater and ground robotic applications, as described in this paper, in order to increase their reliability. The interest of such a concept from the marketing point of view has been demonstrated by a market study. Additional ongoing information concerning the ADVOCATE II project can be found at the project web site: http://www.advocate-2.com .

Acknowledgments. This work is supported by the European Commission (IST-2001-34508).

References

1. Advocate Consortium: ADVOCATE: ADVanced On-board diagnosis and Control of Autonomous sysTEms. IPMU 2002, Information Processing and Management of Uncertainty, Annecy, France, July 1–5, 2002.
2. Advocate Consortium: ADVOCATE II Technical Annex: Description of Work. January 2002.
3. M. Hemmingsson: A Powerflow Control Strategy to Minimize Energy Losses in Hybrid Electric Vehicles. Department of Industrial Electrical Engineering and Automation. Lund Institute of Technology. Lund University. ISBN: 91-88934-11-X. Printed in Sweden. 1999.
4. E. Karden, P. Mauracher, and Friedhelm Schope. Electrochemical modelling of lead/acid batteries under operating conditions of electric vehicles. Journal of Power Sources. Elsevier, **64** (1997) 175–180.
5. P. Mauracher, and E. Karden. Dynamic modelling of lead/acid batteries using impedance spectroscopy for parameter identification. Journal of Power Sources. Elsevier, **67** (1997) 69–84.
6. S. Reehorst. Battery State-of-charge Calculations. Power and Propulsion Office. Appendix B1. NASA Glenn Research Center. 2002.