Comprehensive Framework for the Development of Control and Navigation Systems of Autonomous Underwater Vehicles: the MISSION-SICUVA Project

F.J. Ortiz[1], A. Guerrero[2], J.Gilabert[3], D. Alonso[1], F.Garcia-Cordova[2], F. Rosique[1] F. Sánchez-Ledesma[1]

1) División de Sistemas e Ingeniería Electrónica (DSIE), Campus Muralla del Mar, Universidad Politécnica de Cartagena, 30202 Cartagena, Spain
2) Laboratorio de Vehículos Submarinos (LVS), Campus Muralla del Mar, Universidad Politécnica de Cartagena, 30202 Cartagena, Spain
3) Departamento de Ingeniería Química y Ambiental, Campus Alfonso XIII, Universidad Politécnica de Cartagena, 30203 Cartagena, Spain

Abstract- *This paper presents an overview of coordinated project MISSION-SICUVA, and the results achieved at its recent completion. A prototype of UUV has been built with an orientation to oceanographic research and test of new control algorithms. It consist of an underwater vehicle towing a surface buoy, with applications such as monitoring water quality, high resolution bathymetry of the seabed and its map projection. New biological inspired navigation algorithms have been implemented using a comprehensive component based development framework.*

Keywords- *Unmanned Underwater Vehicles, Biological Inspired Controller, Oceanographic Monitoring, Component Based Software Development, Software Framework.*

1. INTRODUCTION

The SICUVA project (Control and Navigation Systems for Autonomous Underwater Vehicles in missions of Oceanographic Monitoring), began in 2010 and ended in December 2014 aiming for the construction of a prototype of underwater vehicle towing a surface buoy with scalable and reusable software oriented oceanographic research, with applications such as monitoring water quality, high resolution bathymetry of the seabed and its map projection. To do this, the researchers signing this article have developed innovative sensing and control structures neurobiologically inspired to provide autonomy to underwater vehicles.

This project was funded in coordination with the MISSION Project (Comprehensive Framework for Software Development of Autonomous Underwater Vehicles based on Models, Components and Frameworks) in order to provide an enhanced development environment for software control of such vehicles so that (1) the integration and reuse of existing code is promoted, (2) the most advanced principles and techniques of Software Engineering in the domain of AUVs are systematically applied and (3) specific requirements of underwater robotics are taken into account: efficiency, reliability, lack of computational resources and energy constraints.

In the MISSION project a set of software tools have been built following the CBSE (Component Based Software Engineering) and MDSD (Model Driven Software Development) paradigms that facilitate the reuse of proven designs and software components and permit to raise the level of abstraction in software developments. To that end, the FraCC component-based framework has been defined to automatically interpret high-level design (graphical models and components) generating an executable. The model-driven toolchain C-Forge [1] provides support to define new components encapsulating algorithms and existing drivers. Thus, new applications are built simply selecting and assembling the right components in each case. The basis of design of these tools have been recently exposed in prestigious specialized international conferences [2] [3]. Because it is a coordinated project, both the software tools developed and new bio-inspired control algorithms have been validated through its application to real case studies.

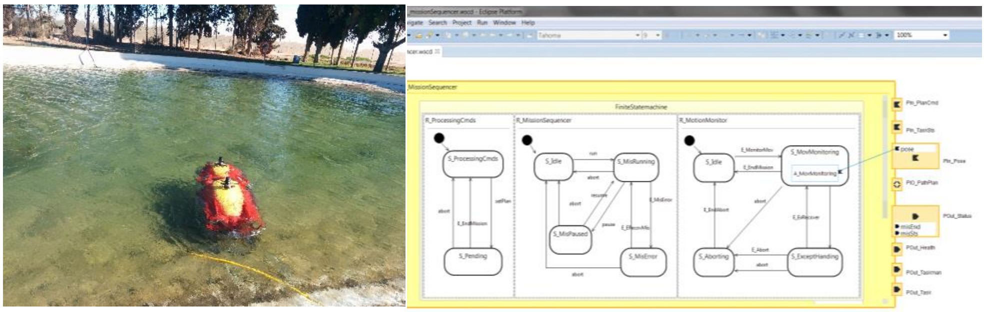

Fig 1. The underwater vehicle (left) and a graphical model of a software component (right)

2. THE BIOLOGICAL INSPIRED CONTROLLER. IMPLEMENTATION USING C-FORGE.

The UUV controller integrates a Self-Organization Direction Mapping Network (SODMN) and a Neural Network for the Avoidance Behaviour (NNAB) both biological inspired [4]. The SODMN is a kinematic adaptive neuro-controller and a real-time, unsupervised neural network that learns to control the underwater vehicle in a nonstationary environment (Fig.2). The NNB is an obstacle avoidance adaptive neuro-controller based on animal behaviour that learns avoidance behaviours based on a form of animal learning known as operant conditioning. This algorithm has been implemented as a module of the framework. This control architecture is based on the Grossberg's conditioning circuit Grossberg & Levine [5, 6] and Chang & Gaudiano [7].

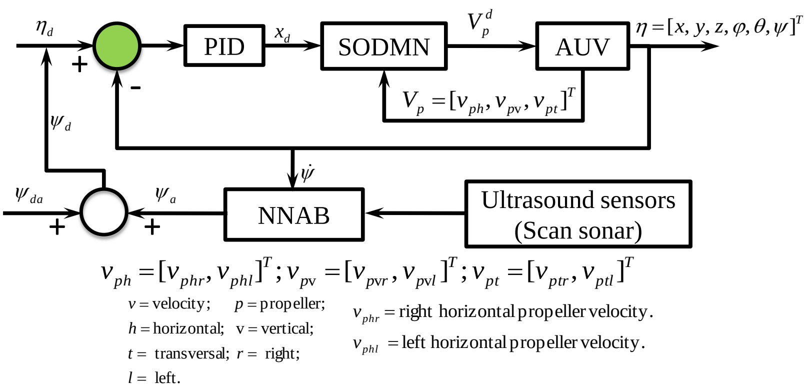

Fig. 2 Neural architecture for reactive and adaptive navigation of the UUV.

To implement the SODMN and NNAB algorithms, the designer simply fill a number of predefined methods in a software component. It is remarkable that software components in C-Forge are modelled by Finite-State Machines (FSM) with hierarchical and orthogonal regions, that is, users are able to model concurrent parts of the whole component behaviour (which can be sequentially or concurrently executed by FraCC). Transitions are triggered by events, which can be produced by both message reception and internal computation. States can contain one activity shell, which is a wrapper of the concrete algorithm that will be executed when the state becomes the region's active state (see inner box on the left, A_ProcessingCmds in Fig.3) These shells only define the messages the activity receives and sends to other activities or ports, represented as black arrows in A_ProcessingCmds in Fig.3. The algorithmic code will be implemented by the user in further steps, filling the methods on the right of Fig.3. Messages serve not only as the mechanism to share information among components, but also as links between the structure and behaviour parts of a component.

The interaction of this component with the other components conforming the application architecture is automatically managed by the FraCC framework. The designer is only concerned to build each component and link them graphically through their ports.

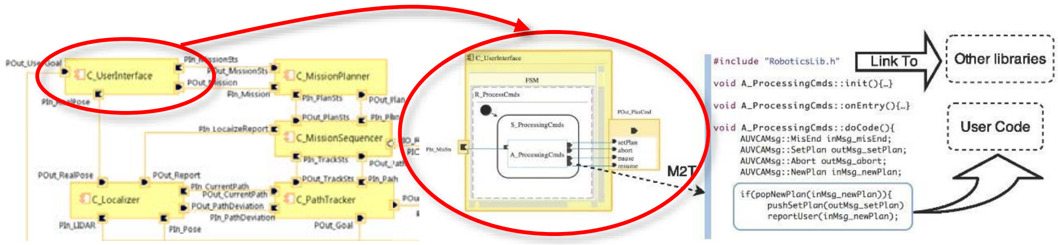

Fig. 3 Excerpt of code for activity A ProcessingCmds (right) in component C_UserInterface (center), as part of the software architecture (left). The skeleton of the class to which the user adds code is generated by an automatic model transformation (M2T).

The control software deployment on different hardware nodes and software processes and threads is performed also using a graphical tool which in turn validates the real-time behavior of the system. The deployment can be easily modified it if necessary depending on the schedulability tests. This is done graphically just moving one of the Region Activities in Fig.4 to other thread. The application of the same approach to develop a different AUV in collaboration with OSL of HWU is presented in [8]

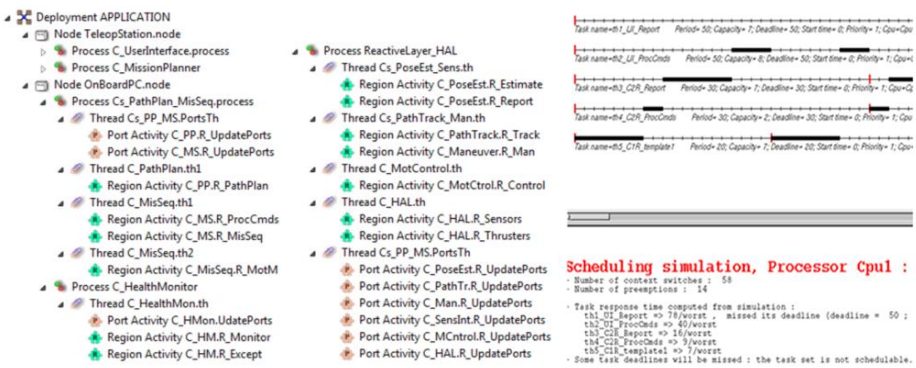

Fig. 4. Deployment model for MISSION CA and schedulability analysis with Cheddar.

3. SEA-TRIALS

To carry out an evaluation of the UUV with this control software architecture, several tests in Mar Menor coastal Lagoon were carried out (Fig.5). One set of test to evaluate the performance of SODM and other to evaluate the performance of the NNB algorithm. Conclusions of these trials are presented in next number or journal RIAI and in international conferences [9].

Fig. 5. Tests in Mar Menor.

The first tests of navigation on the surface and immersion was performed in a pool in the Technological Park of Fuente Álamo, Murcia-Spain. These tests confirmed the manoeuvre of the vehicle and the response of the control architecture through several experiments of directional stability, turns and immersions. The neuro-controllers learned the manoeuvres of avoidance behaviour and recovery of the path to unexpected situations.

4. CONCLUSIONS

One of the distinguishing characteristics of the MISSION-SICUVA coordinated project has been the integration of the most advanced principles and techniques of Software Engineering in the development of the architecture and software development of AUVs. This paper has showed briefly how a CBSE approach favours the rapid development and reuse of software components whilst a MDSD approach allows domain experts to express their knowledge independently from the implementation platform, these models can express graphically the internal of components integrating the architecture and then the code can be automatically generated from these models. Moreover, developers use FraCC to set application deployment, i.e. the number of processes and threads in which the application will be run, as well as their workload. This permits the realization of schedulability analysis during the early design stages. The separation between structural architecture and deployment enables application developers to generate, analyse and test various deployment scenarios without changing the component definitions and the structural view of the application architecture.

The proposal bio-inspired architecture has been successfully demonstrated in the SICUVA project, in the experimental results for trajectory tracking and reaching, as well as avoidance behaviours of an underwater robot. Test carried out confirm the capabilities of this neural architecture to be used in multitasking vehicles for oceanographic research and missions.

REFERENCES

[1] C-Forge toolchain [online]. http://www.dsie.upct.es/cforge/

[2] Ortiz FJ, Sánchez F., Alonso D., Rosique F, Insaurralde, C, "A Model-Driven Toolchain for Developing Component-Based Robotics Software", Proceedings of IEEE ICRA 2013 – Workshop Software Development and Integration in Robotics (SDIR VIII), Karlsruhe, Germany. May 2013.

[3] Ortiz FJ, Alonso D., Rosique, Sánchez F., Pastor J., "A Component-Based Meta-Model and Framework in the Model Driven Toolchain C-Forge", Proceedings of the International Conference on Simulation, Modeling and Programming for Autonomous Robots (SIMPAR IV), Bergamo, Italy, Oct. 2014.

[4] Garcia-Cordova, F., Guerrero-González, A., " Intelligent Navigation for a Solar Powered Unmanned Underwater Vehicle", International Journal of Advanced Robotic Systems", ISSN 1729-8806, DOI: 10.5772/56029. Published: April 2, 2013.

[5] C. Chang, and P. Gaudiano, Application of biological learning theories to mobile robot avoidance and approach behaviors. J. Complex Systems, vol. 1, 1998, pp. 79–114.

[6] S. Grossberg, and D. Levine, Neural dynamics of attentionally moduled Pavlovian conditioning: Blocking, interstimulus interval, and secondary reinforcement, Applied Optics, vol. 26, 1987, pp. 5015–5030.

[7] S. Grossberg, On the dynamics of operant conditioning, Journal of Theorical Biology, vol. 33, 1971, pp. 225—255.

[8] Ortiz, F.J., Insaurralde, C., Alonso, D., Sánchez-Ledesma, F., Petillot, Y. "Model-Driven Analysis and Design for Software Development of Autonomous Underwater Vehicles". ROBOTICA ISSN 0263-5747, Cambridge University Press Vol. 3. DOI: 10.1017/S0263574714001027. Published: April 29, 2014.

[9] Guerrero-González, F. García-Cordova, J. Gilabert-Cervera. "A biologically inspired neural network for navigation with obstacle avoidance in autonomous underwater and surface vehicles". DOI: 10.1109/Oceans-Spain.2011.6003432 Conference: OCEANS. IEEE – Spain. 2011.