Biomimetic Behaviour Based Underwater Control

Savas Piperidis[1] and Nikos C. Tsourveloudis[2]

Abstract— **The proposed controller's main task is to accomplish a safe diving autonomous operation under certain constraints, typically met in Autonomous Underwater Vehicles' (AUV) missions. AUV's behaviour follows a biomimetic approach inspired by the everyday routine of the majority of underwater living creatures. Several phases of this routine seem to play an important role in sustaining creatures' vitality and, as a rule, are present among their distinct physical behaviours. These phases of underwater life routine was the basis for the proposed controller's design. The developed controller succeeds to maintain vessel's energy level, assuring safety and self preservation. In the same time it explores the surrounding area in search for possible targets- preys and is alerted for the detection of potential threats. This control method was extensively tested inside an indoors laboratory underwater experimentation area of 1m^3 using Ale III, a compact custom made prototype AUV.**

I. INTRODUCTION

The outgrowing number of Autonomous Underwater Vehicles (AUV) models, available either commercially or as academia research prototypes, resulted to the development of several underwater control strategies. Last fifteen years' underwater technology evolution triggered the development of 231 unique AUV configurations of 133 vehicle platforms [1]. Although most of contemporary AUVs are the result of ongoing research and development in academia [2], [3] and government funded, military [4] or civil [5], [6], [7], [8] services, underwater technology is nowadays mature enough to present a range of fully functional commercial products, ready to undertake commission 'out of the box' [9], [10], [11], [12], [13], [14], [15] and service the following tasks:

- coastal and sea bed mapping, beach survey, rapid environmental assessment and monitoring, [16],
- oil and gas fuel industry [17],
- cable deployment and route survey [9],
- military uses including mine detection and countermeasures, explosive ordnance disposal, anti- submarine warfare, covert intelligence, surveillance and reconnaissance [9], [10],
- geological, geographical and hydro- graphical surveys [18], [19], [20],
- hull inspection [21],

As a result of AUVs operational potentials expansion during last decade, several control schemes have been developed and tested in simulation, in harbours, across coastal sea line or offshore in the vast oceanic environment. The extremely noisy and unstructured undersea environment along with the highly non linear and coupled underwater vessels' dynamics, pose a great challenge for the design and implementation of efficient and robust robotic controllers [22]. Distinct controller modules are differentiated according to vessel's specifications and payload, as in [23], where a navigational, a vision, a planner and a user interface module compose AUV's controller. PID deliberative control, along with sliding mode control is a robust and reliable solution used in several other AUV projects [24], [25], where the precise or a close estimation of vessel's dynamics model is necessary. In [26] a dynamic diving controller, based on Lyapunov theory and back stepping techniques was used, assuming a perfect knowledge of AUV's dynamic parameters. Then, an adaptive scheme was designed to obtain the desired robustness. The same technique was used for the problem of trajectory tracking of an under actuated, glider AUV moving in the vertical plane with the aid of an internal moving mass [27]. A different approach is chosen in [28] for the implementation of a general modular fuzzy logic control architecture for sonar sensor based AUV navigation in a three dimensional unknown undersea environments. Its advantage is that no assumption is made on the AUV type, on the amount of a priori knowledge of the three dimensional undersea environment, or on static and dynamic obstacle size and velocity. In the specific case that the AUV navigational task is undertaken inside a structured, or known undersea area the path planning strategy may be accomplished via genetic algorithms and B-Splines [29].

II. BEHAVIOUR BASED CONTROL

Contemporary research efforts have developed a large number and variety of distinct robot control schemes trying to confront with underwater robotic autonomy issues. These schemes, fundamentally, may be divided in four classes [30]:

- *Deliberative – Think, Then Act* methods, based on a sequence of sense-plan-act steps.
- *Reactive – Don't Think, (Re)Act* implements the extremely common physical characteristic of *stimulus-response*.
- *Hybrid – Think and Act Concurrently* aiming to combine the best aspects of the two previous methods.
- *Behaviour Based Control – Think the Way You Act* method is structured with interacting modules, called *behaviours*, that collectively achieve the desired robot behaviour [31]. Behaviours are programmed as control modules that cluster sets of constraints in order to achieve and maintain a goal. Each robot behaviour

[1]Savas Piperidis is with the Intelligent Systems and Robotics Laboratory, School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece. spiperidis@isc.tuc.gr

[2]Nikos C. Tsourveloudis is with the Faculty of School of Production Engineering and Management, Technical University of Crete, 73100 Chania, Greece. nikost@dpem.tuc.gr

receives inputs from its sensors and other behaviours and provides outputs to its actuators or to some other behaviours [32]. There is no centralized reasoning or world representation used by the deliberative methods. Also, behaviours do have state and can be used to construct representations, thereby enabling reasoning, planning and learning instead of reactive control's immediate and myopic responses, without representation, to a fast changing environment.

The choice of the right AUV control methodology should be based on its particular characteristics, its task specifications and the overall environment conditions. At the present work, a behaviour based approach was selected for the implementation of an underwater robotic controller. The controller was implemented and tested using Ale III AUV, as a development platform [33]. Ale III is a custom made prototype AUV, designed and constructed at the Intelligent Systems and Robotics Laboratory of the Production Engineering and Management School of the Technical University of Crete. It is an agile and compact vessel with dimensions 17×21×25cm (L×W×H), equipped with an an Ångström Linux powered ARM Cortex A8 architecture *Gumstix Iron Storm* Computer-On- Module (COM), a Vision Sensor Module (VSM) and an Inertial Measurement Unit (IMU). Ale III uses two lateral thrusters for yawing and surging along with a bottom thruster for heaving. More than a year of experimentation and testing with Ale III and the proposed controller scheme took place inside a black opaque cylindrical experimentation tank, with dimensions of 1.3×0.8m (diameter×height), located at a specially designed academic laboratory area, Figure 1.

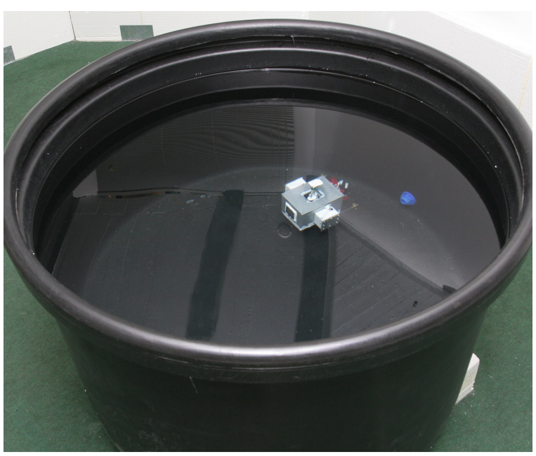

Fig. 1. *Ale III* inside the experimentation area used for development and testing.

The proposed controller scheme mimics sea creatures behaviour. Even the most simple underwater creatures take part in a persistent survival race where sensing of the surrounding environment conditions, quick reasoning, and acting are issues of great importance in their everyday living. Natural evolution has endowed underwater creatures with brain abilities like decision making based on incomplete and noisy perception and physical capabilities like swimming, feeding and reproducing, that no robot constructed until now can macroscopically compete. Autonomy is a given characteristic found in all the ranks of underwater life biological classification. The basic idea behind Ale III autonomy was the self sustaining ability to *survive* for long periods without human assistance and intervention, inside a dynamic complex environment. Behaviour based systems are best suited for this task where the environment presents continuous dynamic changes, fast response along with adaptivity are crucial and the ability to schedule and planning is also desirable [34]. Vessel's task during experimentation was to present a biomimetic behaviour that comprises typical activities among sea creatures that sustain autonomy: *wander*, *prey*, *predator avoidance* and *nesting*. To accomplish this, vehicle's Autonomous Controller Software (ACoS) was designed and implemented in a C++ object oriented architecture following the behaviour based methodology's basic principles [30]:

1) Behaviours are implemented as modules.
2) Each behaviour module may receive data from:
   - VSM and IMU as sensory inputs,
   - the rest of ACoS behaviour modules as triggering commands, suppressors and inhibitors [34]

   and forward their outputs to other behaviour modules or to AUV's effectors, that is its three thruster motors.
3) Many different behaviour modules may independently receive input from the same sensors and output action commands to the same actuators.
4) Behaviours are encoded to be relatively simple, and are added to the system incrementally.
5) Behaviour modules are concurrently active, in order to exploit parallelism and speed of computation, as well as the interaction dynamics among behaviours and between behaviours and the environment.

A. *ACoS showcase*

ACoS follows a biomimetic approach inspired by the everyday routine of the majority of underwater living creatures [35]. It could only be tested inside an indoors underwater experimentation area of 1m^3, showed in Figure 1, and proved its reliability not only in coordinating tasks like target following and object avoidance but also its efficiency in self preservation, showing its potentials for undertaking certain kind of underwater missions. Although at the current configuration Ale III may undertake vision and orientation related tasks, ACoS behaviour based architecture, due to its modular and expandable design, enables a posteri addition of alternative sensors' inputs, command outputs and also the design, implementation and encapsulation of novel behaviour modules.

ACoS integrates open source libraries and handles, apart from the control process, the input-output communications, Figure 2. ACoS programming frame comprises of two main parts:

1) The Robot *class:* uses operating system modules to communicate with the system's Input and Output (I/O) devices, i.e. the sensors and the thruster motors' drivers respectively. It accesses and manipulates the sensors readings

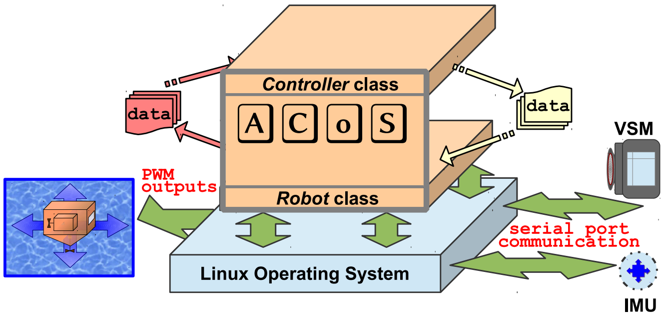

Fig. 2. ACoS architecture.

and forwards the control signals to the motor drivers. *Robot* class is modular in the sense that for every new device connected to the system a new class member has to be implemented. Open source libraries are used in *Robot* class, customized for ASoC's specialized needs

2) The Controller *class:* encapsulates vehicle's behaviour based control. Contains ACoS several behaviour modules as distinct class members. At each controller step it receives an input from the *Robot* class containing the sensor readings. It processes these data according to the ACoS algorithm and produces an output that, in turn, is fed back to the *Robot* class. As soon as *Robot* class receives controller's output, it produces the respective signals to command the thrusters' motor drivers.

Several behaviours may be programmed as different members of the *Controller* class. In fact, this architecture permits several behaviours to be programmed and initialized, so that the system user may choose which one of them to use, depending on the circumstances. All these different behaviour objects of the *Controller* class use the same communication protocol to implement the I/O operations with the *Robot* class.

B. ACoS roles

Ad hoc roles were developed as parts of the behaviour based architecture, to implement the basic biomimetic behaviours mentioned in the previous paragraphs [36], [37]:

- search-wander,
- act as prey – dynamic object avoidance,
- act as predator – dynamic object attraction,
- nesting as goal oriented behaviour – location attraction,

Controller's task is to manage these roles and their interaction with vessel's environment and sustain its autonomous operation via supporting the following distinct tasks simultaneously.

1) Vessel's energy reserves should never drop below a safety level: although Ale III operation is by default energy consuming, still there are two options for recharging. The first option is to act as predator and follow a moving target as if hunting for food. Vessel's energy level will increase for as long as its distance from the target is inside a predefined range. If no target is available in the nearby environment and the energy reserves are reaching the safety level, then the controller must activate the second option, that is the nesting behaviour module to navigate the vessel towards a base station recharger. Both of these options were chosen as Ale III ACoS behaviour modules because:

- Target following is a typical task in underwater robotics, covering fossil fuels' industry intervention missions, missions of cable deployment and route survey, explosive ordnance disposal, antisubmarine warfare, surveillance and reconnaissance.
- Navigating to a specific underwater or surface location, serving as base, is crucial for every AUV mission for recharging or due to safety and recovery reasons.

In ACoS experimentation area the recharging base is represented by a red coloured light sign, Figure 3. The potential target for hunting is represented by a white light following arbitrary three dimensional course inside the tank.

2) Selectively navigate the vessel inside a predefined and marked underwater diving area: ACoS experimentation tank was marked with colour light signs indicating the limits of the diving area where the vessel should bound its operation, Figure 3. The control scheme should guarantee this *bounded* navigation, because once the vehicle slips outside these limits it is concerned to be lost and the controller procedure fails. Three different behaviour modules, programmed as distinct ACoS *Controller* class members, take over the compliance with this constraint:

1) *Proximity adjuster* is ACoS behaviour module that equips Ale III with the ability to keep its distance from the diving area's limits light signs inside an acceptable range.
2) *Yaw control* is the behaviour module that implements Ale III yawing ability.
3) *Search-wander* subsumes the above two modules and navigates the vessel inside the bounded premises of the experimentation area. Apart from avoiding to overcome experimentation area's limits, *search-wander* navigates the vessel to explore the bounded area.

In AUV offshore or coastal operation, serving missions of oceanography, environmental monitoring, pollution detection, fishery, hull inspection, security and surveillance [16], [18], [19], [20], [21], the issue of staying inside a predefined, somehow bounded, area is of great importance not only for efficiency reasons but also for AUV's safety and recovery. Depending on vessel's underwater environment and sensory equipment, the light signs may be replaced by more effective and reliable acoustic signals transmitters, or by GPS signal receptions during successive emergences to the surface. Acoustic signals presume the deployment of a quite larger experimentation area than the one used during Ale III testing and GPS, furthermore, presumes outdoors experimentation. These were the reasons why the visual signs were chosen to implement the limited area controller constraint, instead of some other implementation.

3) Detect and avoid a specific moving object recognised as threat: as soon as a possible threat is recognised, the controller cancels all other ongoing tasks and navigates the vessel to a safe base station, mimicking prey's reaction when being hunted. This is another typical AUV controllers' requisite, contributing to vessels safety and recovery. At Ale III experimentation scenario this specific object-*threat* was implemented as a brown light sign, Figure 3.

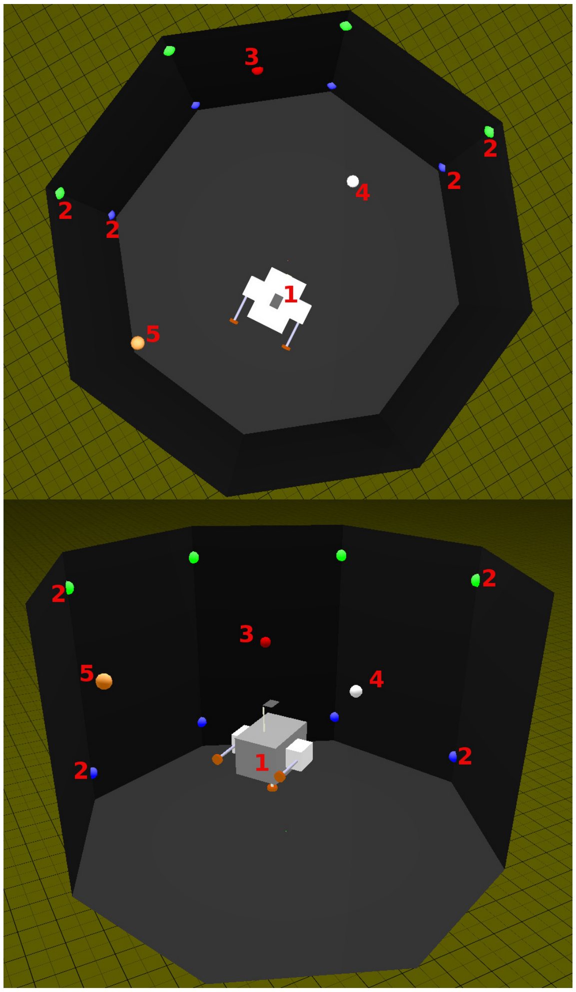

Fig. 3. Three dimensional representation top and side view of ACoS experimentation scenario with Ale III AUV, (1) Ale III, (2) green and blue light signs indicating diving area's upper and lower limits accordingly, (3) base- nest, (4) moving target- prey, (5) moving threat- predator.

III. Behaviours Subsumption Architecture

ACoS behaviour based model is comprised of a subsumption architecture of six different behaviour modules layered incrementally at three levels of competence, Table I, involving interconnections, inhibitors and suppressors [34], [30], Figure 4. Each behaviour is presented as a simple Finite State Machine (FSM) and programmed as a *Controller* class member. FSM representation is the standard technique used to describe behaviour based control methods [34]. It was chosen against other representation techniques, like data flow diagrams, flow charts or pseudo code for its simplicity and the ability for intuitive graphical representation. This technique manages to represent the distinct ACoS behaviour modules using simple FSMs containing, at most, four states

TABLE I
Behaviours Index

level	behaviour number	behaviour name	energy balance
0	01	proximity adjuster	negative
0	02	yaw control	negative
1	11	search- wander	negative
1	12	nesting	negative/positive
2	21	prey	negative/positive
2	22	predator	negative/positive

and nine transitions. Moreover, designing and representing ACoS behaviour based architecture with stand alone FSMs aided the compliance with the behaviour based principles of concurrent execution, modularity, simultaneous reception of sensory inputs and competence for the controller outputs [30]. FSM diagrams are included in the Appendix.

Depending on behaviour modules' interaction and the dynamically changing environmental conditions, the level of AUV's energy reserves increases or decreases after each controller time step. For example, *living* in an environment with redundancy of preys has definitely a positive influence to AUV's energy level. Energy consuming behaviours are the ones that have no means of recharging energy, instead they just consume vessel's energy to navigate it. They are listed in Table I as negatively balanced while those who feed energy to the AUV by hovering inside the recharging base premises or during a successful target following task, are listed as positively balanced. For some behaviours the energy balance may be alternatively positive or negative depending on their progress and outcome. Thus for example a successful target following task will feed energy to Ale III in contrast to an unsuccessful one, that will just consume some of its energy reserves. In the current ACoS experimentation scenario the energy term was virtual, implemented as an ACoS variable. So after a successful *hunt*, for example, the increase in energy level is virtual, by the means of ACoS, without some kind of energy transformation or transfer between the vessel and its surrounding underwater environment. ACoS performance depends on the behaviours' interaction parameters fine tuning, so as to obtain the lowest energy consumption during controller *life* time.

A. ACoS Zeroth Level

The zeroth level of the proposed behaviour based architecture, Figure 4, comprises of the following two basic-low level navigational behaviour modules, implemented using Proportional Derivative (PD) controllers:

- *01: proximity adjuster*, keeps vessel's distance from the signs inside a fixed range.
- *02: yaw control*, navigates the AUV to an orientation of a fixed target yaw value.

Zeroth level is the basis for the upper two levels. It implements ACoS potentiality for avoiding impediments, achieving and maintaining an orientation. Both zeroth level's behaviours consume energy as long as they take over the control procedure. At Figure 4 the wire with a circle beside it,

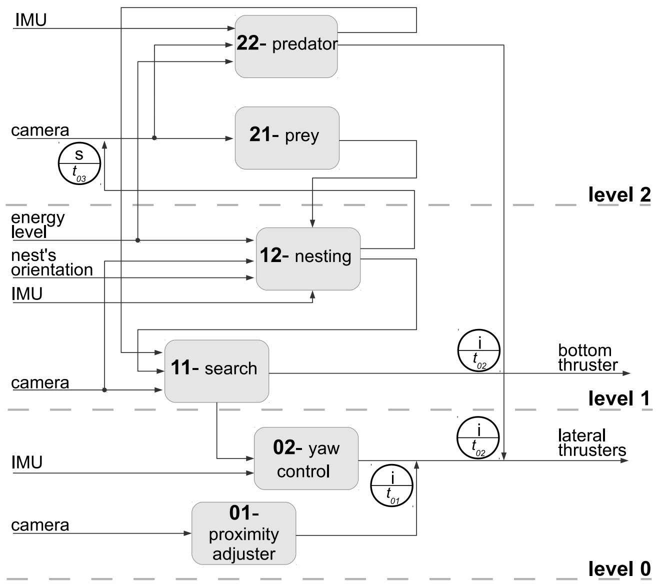

Fig. 4. ACoS three level behaviour based control subsumption architecture.

connecting the output of the *proximity adjuster* behaviour to the *yaw control* is an inhibitor [34]. When an output message from the *proximity adjuster* behaviour travels along this wire it inhibits *yaw control* outputs for a time period of t_{01}, where t_{01} is an ACoS parameter, and thus has the full control of thrusters' commands during this period.

B. ACoS Level 1

ACoS Level 1 introduces two new behaviour modules:

- *11: Search-wander* subsumes the lower level *02 yaw control* behaviour module for exploring the space between the tank's surface and bottom coloured light signs. There are green signs attached to the walls of the experimentation tank, near the surface and blue ones near the bottom, Figure 3. They indicate the safe diving area limits for the robot.
- *12: nesting* implements AUV's homing procedure for returning to the nest- base. It interacts with the *01 search-wander* behaviour and depending on vessel's energy reserves, may decide to navigate it to its nest-base. It presents a positive energy balance as long as AUV's position is inside nest's-base's range. While on the search of nest-base, the AUV consumes energy.

C. ACoS Level 2

The second and top level of competence adds to the proposed model two new behaviour modules:

- *21: prey* is responsible for detecting the presence of a brown light that denotes a possible predator-threat. As soon as the controller is alerted by the detection of a brown coloured light, *21 prey* and the subsumed *12 nesting* behaviours navigate the AUV straight to its nest-base.
- *22: predator* detecting the presence of a white light target that during the experimentation scenario reverses the roles and represents a prey the AUV has to hunt and catch. *Predator* subsumes the lower level *11 search* behaviour for implementing the hunting procedure inside the diving area limits of the experimentation tank, using a PD controller.

At Figure 4 the use of letter *s* at an upper hemicycle, beside a wire, depicts suppression of the corresponding VSM sensory data inputs to *21 prey* and *22 predator* behaviour modules [34]. Using this suppressor *12 nesting* cancels VSM sensory inputs to level 2 behaviours, so as to neutralize them and take full command of the vessel until it returns to the base, after a time period of t_{03} controller steps. The inhibitors of *22 predator* behaviour cancel the outputs of *11 search* and *02 yaw control* and take full command of ACoS outputs during a hunting procedure until a prey is caught or a maximum interval time of t_{03} controller time steps has expired.

IV. Experimentation Results

ACoS was extensively tested using Ale III AUV inside the indoors laboratory experimentation area of Figure 1. The experimentation tank was marked with light signs representing diving area's limits and AUV's base-nest. A brown and a white light suspended from a rod, appeared randomly inside the diving area and following arbitrary three dimensional courses, played the role of a potential threat-predator and target-prey accordingly. The experimentation scenario was chosen to trigger and use alternatively all the behaviours, Table II: The vessel begins its underwater commission or, using biomimetic terms, *life* filled up with energy and wanders between its area limits in the experimentation tank. After a while a brown light *predator* or a white light *prey* may appear to the underwater scenery and Ale III must respond accordingly: to hide at the safety of its nest-base or hunt the *prey*. Wandering or hunting decreases energy reserves and if a *prey* is not available then the vessel must visit its nest-base for recharging.

This kind of experimentation was aiming at highlighting controller's ability to sustain autonomous underwater operation, mimicking the commutation of the several behaviours typically found in the routine of undersea living creatures. The experimentation had several phases of *search- wander*, *acting as prey* featuring dynamic object avoidance, *acting as predator* featuring dynamic object attraction, *nesting* as goal oriented behaviour and location attraction, because these phases are common tasks among AUV missions in coastal line and offshore. ACoS was challenged to support Ale III autonomous operation and dynamically confront possible threats in its search for potential targets and in the same time monitor vessel's energy reserves, assuring reliability and safe recovery.

Figure 5 show typical results following previous paragraph's experimentation scenario. The test lasted several minutes but the plots show only the first three minutes for readability reasons. The first plot of each figure shows behaviours' interaction log throughout the experiment. The second plot shows AUV's *virtual* energy level and instantaneous consumption measured in virtual *energy units*. At each controller step a '+' mark is added for these ACoS behaviours that take part to the controller outputs' determination. According to the behaviour based design principles all of them have access to the controller outputs. Still, at each controller time step not all of them contribute to the control process' outputs due to the subsumption architecture inhibitors and successors or the lack of the proper stimulus. Figure's 5 results confirm the correctness of the behaviour modules' interaction and the compliance with scenario's constraints. For example, during the wander phase in search for a target, there is a continuous commutation between $search_G$, that is *search* behaviour detecting for diving area's upper limits green lights and $search_B$, that is *search* detecting for lower limits blue lights, as shown at the 7^{th}, 15^{th}, 36^{th}, 45^{th}, 50^{th} second etc. The wander phase is cancelled every time a *prey*, as at 17^{th} second, or a *predator*, as at 64^{th} second, is detected and *nesting* is actuated when the energy reserves are low, as at 131^{th} second, or a *predator* was detected, as at the 64^{th} second. The decrease and increase of the energy level between 26^{th} and 35^{th} second denotes a, partly, successful *hunt*.

TABLE II

ACoS REFLEXES IN ACCORDANCE WITH THE DETECTED LIGHT COLOUR SIGNS

Colour	*ACoS Behaviours*	*Next Action*	*Physical Meaning*
green	search- wander yaw control proximity adjuster	stop emerging and start diving	Ale III reached experimentation area's upper limit
blue	search- wander yaw control proximity adjuster	stop diving and start emerging	Ale III reached experimentation area's lower limit
brown	prey nesting search- wander yaw control proximity adjuster	navigate to base-nest	Ale III detected a threat- predator
red	prey nesting search- wander yaw control proximity adjuster	hover in front of the base- nest	Ale III is recharging or trying to protect from a threat- predator
white	predator	follow target	Ale III detected a target- prey

V. CONCLUSIONS AND FURTHER WORK

ACoS implements a control methodology suitable for autonomous AUV missions. Its behaviour based design facilitates the controller representation with remarkable simplicity. The easiness of programming the behaviour modules, based on their FSM, and implementing the whole control scheme at an object oriented programming environment, significantly reduced project's development time. During all design, development and testing phases this scheme demonstrated the advantage of modularity. Ale III, equipped with the

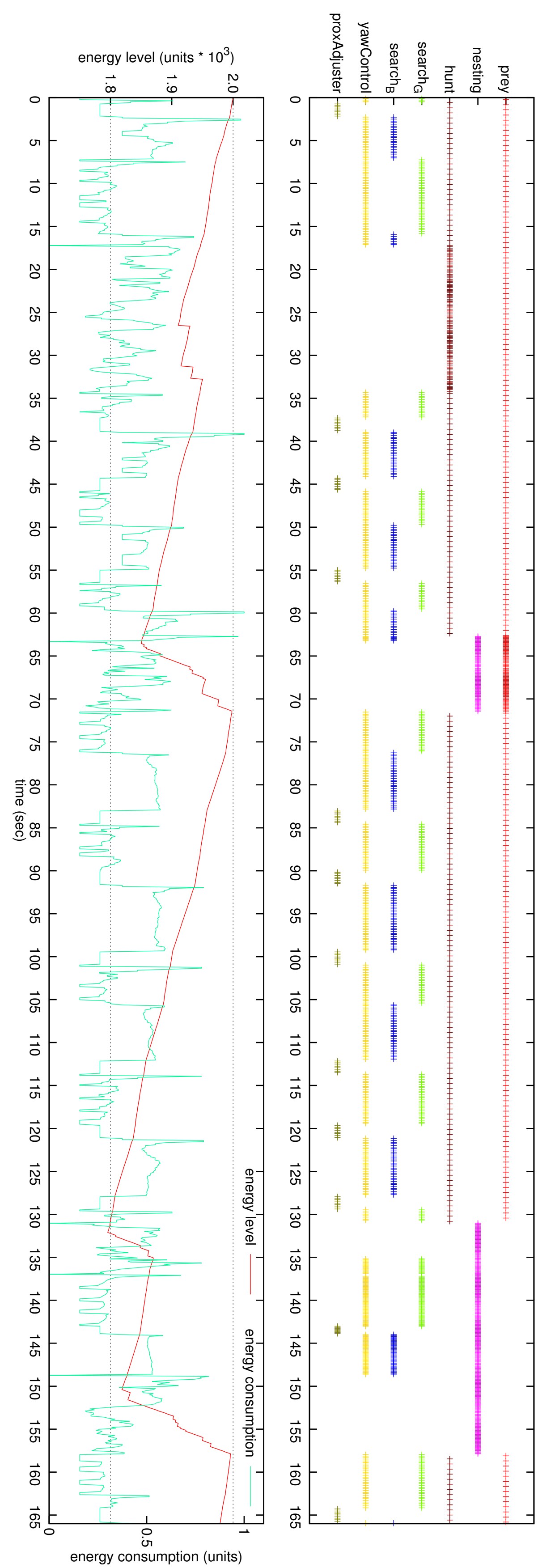

Fig. 5. Biomimetic behaviour based control experimentation results.

biomimetic behaviour based ACoS, was commissioned to test the proposed model inside the laboratory experimentation area. The experimentation results not only confirmed the behaviour based design correctness but also demonstrated ACoS ability to sustain underwater autonomous operation undertaking typical AUV tasks and satisfying certain constraints met at common underwater missions.

ACoS development was based on an interaction between behaviour modules organized in a multi level hierarchy. This interaction is programmed in a static way inside ACoS framework. ACoS at the current state of development does not contain any sense of *memory* or *learning*. This would be an interesting further research target: artificial neural network and genetic algorithms evolution of an intelligent biomimetic behaviour, based on and enriched with the expertise gained during the *life* time of each real or simulated experiment [38] [39]. The individual behaviour based modules would be borrowed from the already developed ACoS version and their interaction will be the result of an evolution process. Under this point of view, the evolved behaviour will encapsulate the concepts of memory and learning [40] [41]. This combination will be an *intelligent* behaviour based control scheme, based on the static approach, real tests or simulation results. Another extremely interesting issue is redesigning ACoS so as to sustain cooperative autonomous underwater behaviour, following the same principles of behaviour based control and encapsulating the experience gained from the current atomic behaviour work. The next experimentation step with Ale III will be at seashore. New scenarios will test ACoS potentials in this extremely noisy and unstructured environment. The lighting conditions undersea will be a challenge for ACoS VSM. Sea waving and currents will test ACoS *seaworthiness* in the hard way, far from the tranquil laboratory experimentation area.

Apart from the issue of testing in the *wild*, the domestic laboratory experimentation area needs expansion. Indoors experimentation presents overwhelming advantages in relevance with open sea in terms of availability, feasibility, money and time cost. To exploit the full range of them the acquisition of a tank with volume at least 3m^3 will be a great progress for the experimentation area. As realized after several months of tests in the existing laboratory facility, for every *Ale III class* AUV added to the experimentation scenario at least 1m^3 of additional underwater space volume is necessary. As soon as a bigger experimentation area is available, more than one vessels may be operated simultaneously inside the tank. This would be a great opportunity to develop a cooperative behaviour ACoS.

APPENDIX

A. Behaviour 01: proximity adjuster

1) States: (the initial state is denoted by its double lined circle) the vessel is not performing any movement and there is no thruster power manipulation, the vessel is performing a backwards surge movement for t_{01} controller steps, the vessel is performing a surge movement for t_{01} controller steps.

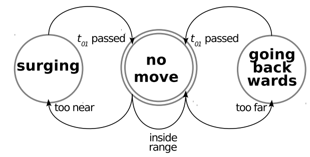

Fig. 6. Zeroth level *01: proximity adjuster* behaviour's FSM diagram.

2) Transitions and sensor inputs: coloured signs traced by VSM show that AUV's distance from tank walls is further than the maximum permitted value – distance from tank walls is closer than the minimum permitted value – distance from tank walls is inside the permitted range.

B. Behaviour 02: yaw control

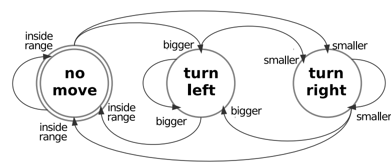

Fig. 7. Zeroth level *02: yaw control* behaviour's FSM diagram.

1) States: AUV is not performing any movement and there is no thruster power manipulation – AUV is turning left – AUV is turning right.

2) Transitions and sensor inputs: vessel's yaw heading value is inside range – yaw value is bigger than the target value – yaw value is smaller than the target value.

C. Behaviour 11: search-wander

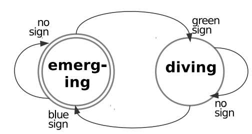

Fig. 8. Level 1 *11: search- wander* behaviour's FSM diagram.

1) States: emerging keeping a steady yaw heading – diving keeping a steady yaw heading.

2) Transitions and sensor inputs: no sign detected – green sign detected – blue sign detected.

D. Behaviour 12: nesting

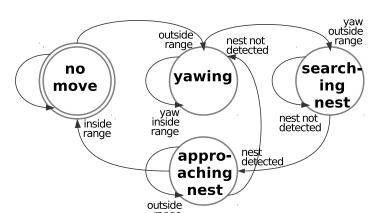

Fig. 9. Level 1 *12: nesting* behaviour's FSM diagram.

1) States: inside nest performing no movement – yaw turning – searching for red coloured nest-base light sign – nest detected and approaching to nest.

2) Transitions and sensor inputs: AUV's distance from nest is inside range – distance from nest is outside range – AUV's yaw heading value is inside range and heading towards nest – yaw heading value is not inside range – nest's colour sign detected – nest's colour sign not detected.

E. Behaviour 21: prey

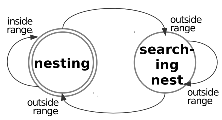

Fig. 10. Level 2 *21: prey* behaviour's FSM diagram.

1) States: nesting has been accomplished – searching for the nest-base.

2) Transitions and sensor inputs: distance from nest-base is inside range – distance from nest-base is not inside range.

F. Behaviour 22: predator

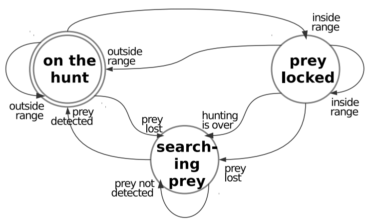

Fig. 11. Level 2 *22: predator* behaviour's FSM diagram.

1) States: on the hunt of detected prey-target – prey-target locked.

2) Transitions and sensor inputs: distance from prey-target is inside range – distance from prey-target is not inside range – prey-target is lost – prey-target is locked for a period long enough to fill up with energy or a maximum interval time of t_{03} controller time steps has expired and the hunting is successfully terminated – prey-target detected – prey-target is not detected.

REFERENCES

[1] Autonomous Undersea Development Center (AUVAC). (2013, December) AUV database. [Online]. Available: http://auvac.org

[2] University of Victoria. (2013, December) Ocean Technology Lab MAKO AUV. [Online]. Available: http://web.uvic.ca/∼lacir/ocean

[3] Virginia Tech Department of Electrical and Computer Engineering. (2013, December) Autonomous Systems and Control Laboratory. [Online]. Available: http://www.ascl.ece.vt.edu

[4] Naval Sea Systems Command US Navy. (2013, December) Naval Undersea Warfare center. [Online]. Available: http://www.navsea.navy.mil/nuwc

[5] National Oceanography Centre. (2013, December) Autosub6000 AUV. [Online]. Available: http://noc.ac.uk

[6] Monterey Bay Aquarium Research Institute. (2013, December) Dorado AUV. [Online]. Available: http://www.mbari.org/auv

[7] Korea Ocean Research and Development Institute. (2013, December) Maritime & Ocean Engineering Research Institute. [Online]. Available: http://www.moeri.re.kr/kordi_daeduck/eng

[8] Japan Agency for Marine-Earth Science and Technology. (2013, December). [Online]. Available: http://www.jamstec.go.jp/maritec/e

[9] Gavia Autonomous Underwater Vehicles. (2013, December). [Online]. Available: http://www.gavia.is

[10] Bluefin Robotics. (2013, December). [Online]. Available: http://www.bluefinrobotics.com

[11] Kongsberg Maritime. (2013, December). [Online]. Available: http://www.km.kongsberg.com/hydroid

[12] The Atlas Electronik Group. (2013, December) Unmanned Vehicles. [Online]. Available: http://www.atlas-elektronik.com

[13] Daewoo Shipbuilding & Marine Engineering Co. Ltd. (2013, December) DSME E&R. [Online]. Available: http://dsmeu.en.ec21.com

[14] International Submarine Engineering Limited. (2013, December). [Online]. Available: http://www.ise.bc.ca

[15] SAAB Ltd. (2013, December) Seaeye. [Online]. Available: http://www.seaeye.com

[16] M. Ludvigsen, G. Johnsen, P. a. Lagstad, A. J. Sorensen, and O. Odegard, "Scientific operations combining ROV and AUV in the Trondheim Fjord," in *2013 MTS/IEEE OCEANS - Bergen*. IEEE, June 2013, pp. 1–7.

[17] G. Antonelli, T. I. Fossen, and D. R. Yoerger, "Underwater robotics," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, Jan. 2008, vol. 15, no. 5, ch. 43, pp. 987–1008.

[18] C. Kaminski, T. Crees, J. Ferguson, A. Forrest, J. Williams, D. Hopkin, and G. Heard, "12 days under ice - An Historic AUV Deployment in the Canadian High Arctic," in *2010 IEEE/OES Autonomous Underwater Vehicles*, no. 1. IEEE, Sept. 2010, pp. 1–11.

[19] T. Hiller, A. Steingrimsson, and R. Melvin, "Expanding the Small AUV Mission Envelope; Longer, Deeper & More Accurate," in *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*. IEEE, Sept. 2012, pp. 1–4.

[20] A. Alvarez, A. Caffaz, A. Caiti, G. Casalino, L. Gualdesi, A. Turetta, and R. Viviani, "Fòlaga: A low-cost autonomous underwater vehicle combining glider and AUV capabilities," *Ocean Engineering*, vol. 36, no. 1, pp. 24–38, Jan. 2009.

[21] G. a. Hollinger, B. Englot, F. S. Hover, U. Mitra, and G. S. Sukhatme, "Active Planning for Underwater Inspection and the Benefit of Adaptivity," *The International Journal of Robotics Research*, vol. 32, no. 1, pp. 3–18, Nov. 2012.

[22] T. I. Fossen, *Guidance and Control of Ocean Vehicles*. Chichester, England: John Wiley & Sons, 1994.

[23] C. Barngrover, R. Kastner, T. Denewiler, and G. Mills, "The stingray AUV: A small and cost-effective solution for ecological monitoring," in *OCEANS 2011*. Waikoloa, HI: IEEE, 2011, pp. 1–8.

[24] J. Busquets, D. Tudela, F. Perez, J. Busquets-Carbonell, A. Barbera, C. Rodriguez, A. J. Garcia, and J. Gilabert, "Low-cost AUV based on Arduino open source microcontroller board for oceanographic research applications in a collaborative long term deployment missions and suitable for combining with an USV as autonomous automatic recharging platform," in *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, no. 1. IEEE, Sept. 2012, pp. 1–10.

[25] K. Kim and H. S. Choi, "Analysis on the controlled nonlinear motion of a test bed AUVSNUUV I," *Ocean Engineering*, vol. 34, no. 8-9, pp. 1138–1150, June 2007.

[26] L. Lapierre, "Robust diving control of an AUV," *Ocean Engineering*, vol. 36, no. 1, pp. 92–104, Jan. 2009.

[27] J.-w. Li, B.-w. Song, and C. Shao, "Tracking Control of Autonomous Underwater Vehicles with Internal Moving Mass," *Acta Automatica Sinica*, vol. 34, no. 10, pp. 1319–1323, 2008.

[28] V. Kanakakis, K. P. Valavanis, and N. C. Tsourveloudis, "Fuzzy-Logic Based Navigation of Underwater Vehicles," *Journal of Intelligent and Robotic Systems*, vol. 40, no. 1, pp. 45–88, May 2004.

[29] V. Kanakakis and N. Tsourveloudis, "Evolutionary Path Planning and Navigation of Autonomous Underwater Vehicles," in *Control Automation, 2007. MED '07. Mediterranean Conference on*, June 27-29 2007, pp. 1–6.

[30] Maja J. Mataric and F. Michaud, "Behavior-Based Systems," in *Springer Handbook of Robotics*. Springer, 2008, ch. 38, pp. 891–909.

[31] M. J. Mataric, "Designing and Understanding Adaptive Group Behavior," *Adaptive Behavior*, vol. 4, no. 1, pp. 51–80, Sept. 1995.

[32] R. C. Arkin, *Behavior-Based Robotics*. Cambridge, USA: MIT Press, 1998.

[33] S. Piperidis and N. C. Tsourveloudis, "Testing controllers on ALE III: A low cost mini Autonomous Underwater Vehicle," in *21st Mediterranean Conference on Control and Automation*. IEEE, June 2013, pp. 551–557.

[34] R. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal on Robotics and Automation*, vol. 2, no. 1, pp. 14–23, 1986.
[35] T. L. Anderson and M. Donath, "Animal behavior as a paradigm for developing robot autonomy," *Robotics and Autonomous Systems*, vol. 6, no. 1-2, pp. 145–168, June 1990.
[36] M. O. Franz and H. A. Mallot, "Biomimetic robot navigation," *Robotics and Autonomous Systems*, vol. 30, no. 1-2, pp. 133–153, Jan. 2000.
[37] O. Trullier, S. I. Wiener, A. Berthoz, and J. a. Meyer, "Biologically based artificial navigation systems: review and prospects." *Progress in neurobiology*, vol. 51, no. 5, pp. 483–544, Apr. 1997.
[38] A. P. Engelbrecht, *Computational intelligence: an introduction*. John Wiley & Sons Ltd, 2007.
[39] M. Lee, "Evolution of behaviors in autonomous robot using artificial neural network and genetic algorithm," *Information Sciences*, vol. 155, no. 1-2, pp. 43–60, Oct. 2003.
[40] C. Shin and S. Park, "Memory and neural network based expert system," *Expert Systems with Applications*, vol. 16, no. 2, pp. 145–155, Feb. 1999.
[41] C. G. Atkeson and S. Schaal, "Memory-based neural networks for robot learning," *Neurocomputing*, vol. 9, no. 3, pp. 243–269, Dec. 1995.