

THE MODULAR INTELLIGENT BEING: REVIEW AND A PROPOSAL

Sandor M. Veres *

** School of Engineering Sciences, University of
Southampton, UK
Highfield, SO17 1BJ, Email:s.m.veres@soton.ac.uk*

Abstract: The paper provides a review and addresses the problem of creating a modular hardware system that can be effortlessly reconfigured by developers of autonomous intelligent control systems for mission type task execution in realtime physical environments. The idea is to allow for a common hardware platform which enables algorithmic results to be compared, accumulated and properly assessed by the research community. Such a single platform can then be used by researchers of autonomous systems in their research to speed up progress. *Copyright ©2005 IFAC*

Keywords: Autonomous control, learning systems, adaptive systems, hardware architecture, software architecture.

1. INTRODUCTION

PC architectures are in universal use, even if you are running Linux on yours. It was such a long time ago that we already tend to forget about the 1950's dilemma, when it was debated whether analogue or digital computers are for the future (see the book of John von Neumann: "The Computer and the Brain"). Digital computers have been very successful, but not by mimicking human intelligence. They developed into different direction: they have become useful tools. The computers of today are reliable task executors first for replacing simple paper processing (banks, company admin), then controlling industrial processes (vehicle and manufacturing control), then they became platforms of communications systems (the Internet and digital phones) and now they start doing more intelligent jobs with the fast emergence of parallel analogue devices linked up.

The universal working platforms of the digital computers, that are used today on mass, are the operating systems. These now integrate more

and more intelligent functions of communication (including TV, video, wireless and Internet) and basic data processing and data organization. The operating system can run any software which can transform the computer into a more specialized machine to perform various tasks in the home, office, manufacturing, vehicles on the ground, in the air and in space.

All this is so obvious that why to recall this at all? Perhaps this could bring in front of us the whole perspective of development and can help to identify new avenues to follow in computer applications development. One such direction is to make computers easier to use in applications where interaction with the physical world using sensors, temperature, pressure, voltage, distance, still and video cameras, directional microphone arrays, etc. is important.

It is not that you could not buy a software today that drives a particular device and serves some application area. A lot of software is available. The problem is that it is rare to have a software

that enables the computer to execute a mission autonomously. For instance

(1) While in your office: why should you not plug your (powerful) computer into your car and ask it to drive to your home, ask your wife for a book and bring it to your office? Or send your computer down for shopping to to local store?

(2) While at home: why could you not ask your computer to mow the lawn or collect the leaves from the garden path?

(3) Why could you not ask your computer to entertain your guests with jokes, a quiz or some performance , depending on their interest?

(4) Why could you not ask your computer to monitor your health and advise you and save you a fortune on future treatment bills when you break down because you neglected your health?

(5) Why could you not send your computer to fix something on your factory floor?

Do you think these are far fetched examples as these tasks are too complex? Or do you think some academic or a leading electronic products manufacturer will once come up with a surprise solution to these very difficult problems? Then think again. These things could happen much more quickly if more researchers could be involved in developing intelligent systems in a more concerted way. Or do you think that you will have an intelligent car for driving, an intelligent lawn mower and a "playmate" for playing with the children in the garden and these will not be called "computers"? I agree that they may have special names depending on the application, but what will be at the core of these systems? *In time they will be the same computer architectures with different devices hooked on!*

This paper addresses the hardware and software structural issues of "more" intelligent machines that have the potential for complex task execution without human supervision.

2. A DIVERGING SET OF RESULTS

This section reviews some of the recent achievements in the area of autonomous intelligent control systems (AICS) and then some conclusions are drawn. The frequently mentioned "agent" is an autonomous, computational entity that can be viewed as perceiving its environment and acting upon it by way of event-driven objects that can be integrated in automated environments to control certain tasks.

2.1 A brief review of some results

From methodological point of view AICS control architectures are usually classified as (1) hierarchical (2) behavioral or (3) hybrid architectures.

- (1) *Hierarchical architecture* is also known as deliberative architecture. Data from sensors are sent to the world model. The world model is updated and decisions are made based on the sensory data, the given mission, and information stored in the database. The new action commands are then sent to the actuators (Kim and Yuh (1992)).
- (2) The *behavioral architecture* Kortenkamp et al. (1998); Brooks (1999); Pfeifer and Scheir (1999) are sometimes called heterarchical or reactive architectures. The mission is usually described as a sequence of phases with a set of predefined behaviors. These behaviors are based on the sense-react principle.
- (3) The *hybrid architecture* is trying to minimize the limitations of the previous two architectures (Gat (1998), Bonasso (1992)). An architecture developed by Healey et al. (1998) is organized in three levels: execution level, tactic level and strategic level, this is an approach that is often taken by designers.

Hu et al. (1998) present a modular computing architecture for intelligent control of autonomous robots that takes the form of multiple sensing and control layers, based on Locally Intelligent Control Agents (LICAs) in which IBM PowerPC, SIEMENS 80C166, and INMOS Transputers are used.

Schetter et al. (2003) presents an architecture and multi-agent design and simulation environment that enables agent-based multi-satellite systems to fulfill their complex mission objectives for satellite constellations. Its application is shown for a SAR (synthetic aperture radar) mission.

Burgard et al. (1999) addresses the logical foundations of goal-regression planning in autonomous rational agents. It first recognizes that goals and subgoals will often be conjunctions, and to apply goal-regression planning to a conjunction we usually have to plan separately for the conjuncts and then combine the resulting sub-plans.

Chella et al. (1997) proposes a cognitive architecture for artificial vision. The architecture is cognitive in the sense that a conceptual representation level is placed between the sub-symbolic level, that processes sensory data, and the linguistic level, that describes scenes by means of a high level language. The conceptual level plays the role of the interpretation domain for the symbols at the linguistic levels.

As intelligent, autonomous systems are embedded in critical real-world environments, it becomes increasingly important to rigorously characterize how these systems will perform. Research in real-time computing and control has developed ways of proving that a given control system will meet the demands of an environment. Musliner et al. (1995) goes beyond that by including the dynamic planning of control actions for agents that can flexibly achieve their goals in changing environments.

Autonomous underwater vehicles (AUVs) often have to perform tasks requiring physical contacts with the underwater environment, such as plug-in/plug-out, construction and repair, cable streaming, mine hunting, munitions retrieval, and scientific sampling, etc. Kim and Yuh (2004) describe a semi-autonomous underwater vehicle that has multiple on-board CPUs, redundant sensors and actuators, on-board power source and a robotic manipulator.

There are a number of proposals (cf. Viana et al. (2004)) in the network community literature for technologies of ad hoc, and more recently, sensor networks to serve the deployment of distributed, autonomous, spontaneous, and self-organizing systems.

Farahvash and Boucher (2004) introduce a multi-agent system in an automated manufacturing environment. The architecture includes functions at the manufacturing cell level, materials handling and transport level, and factory scheduling level.

Barbier and Chantry (2004) presents an on-board architecture to enable an unmanned aerial vehicle (UAV) to carry out an observation mission autonomously. The management performed by the architecture relates to the objectives of the mission to observe several local areas in dynamic, partially known and unsafe environments.

Remagnino et al. (2004) applies latest advances in hardware technology and state of the art of computer vision and artificial intelligence research to develop autonomous and distributed monitoring systems. A multi-agent architecture for the understanding of scene dynamics is merging the information streamed by multiple cameras.

Arenaa et al. (2004) deals with the the design and realization of mobile platforms for Mars exploration, wheeled and legged ones, for autonomous deployment in unknown and hostile environments. Global Aerospace Corporation (GAC) Pankine et al. (2004) developed a system architecture for exploration of planetary atmospheres and surfaces from atmospheric altitudes.

All kinds of mobile robots can benefit from fast automated 3D modelling of their environment. Surmann et al. (2003) presents an automatic sys-

tem for gaging and digitalization of 3D indoor environments using a reliable 3D laser range finder and three elaborated software modules.

Yasuda (2003) describes the design and implementation of a modular distributed and hierarchical architecture for distributed autonomous control of modular robot systems using parallel programming in industrial robotic manufacturing applications.

Matsumotoa and Tsujinob (2003) considers the human brain as an organ especially differentiated to acquire algorithms in a self-organized fashion and these acquired algorithms allow the brain to respond to the ever changing environment that surrounds it. They hypothesize that two general principles give the brain its auto-designing capacity and, consequently, the potential to acquire algorithms. (1) Output-driven operation (2) Memory-based architecture.

Wang et al. (2003) proposes a communication and control architecture to improve the capability and the flexibility of multiple autonomous robot systems in performing a complicated task and coping with unpredictable situations.

The GLAIR (grounded layered architecture with integrated reasoning) by Shapiro and Ismail (2003) has been used for cognitive robots and intelligent autonomous agents in a series of projects in which Cassie, the SNePS cognitive agent, has been incorporated into hardware- or software-simulated cognitive robots.

Hagras et al. (2003) describes the use of intelligent autonomous systems in intelligent buildings (IB). IB agents are based on a hierarchical fuzzy genetic multi-embedded agent architecture comprising a low-level behaviour based reactive layer. The fuzzy rules related to the room resident comfort are learnt and adapted online using fuzzygenetic techniques. The learnt rule base is updated and adapted via an iterative machine-user dialogue.

An object oriented control architecture for autonomy (O2CA2) is developed by Ridao et al. (2002), As a hybrid system it is organized in three layers (deliberative, control execution and reactive) merging the deliberation mechanisms of planning with reactive behaviours in charge of controlling the robot in real-time.

Frezza-Buet and Alexandre (2002) examines the autonomous control problem starting from the biological perspective. They point out that the behavior of an autonomous robot is such a wide topic that a general framework paradigm of inspiration must be chosen in order to obtain a consistent model, for instance an artificial intelligence approach. They say that

... a general framework is not sufficient to determine a fully specified program to be implemented in a robot. Many choices, tuning and tests must be carried out before obtaining a robust system. [Frezza-Buet and Alexandre (2002)]

Broggi et al. (2000) discuss the main architectural issues of a challenging application of real-time image processing: the vision-based automatic guidance of road vehicles.

Sequeira et al. (1999) describe an integrated approach to the construction of textured 3D scene models of building interiors from laser range data and visual images, a system that can be very useful for mobile autonomous systems.

Fuzzy logic techniques have become popular to address various processes for multisensor data fusion. Examples include the following: (1) fuzzy membership functions for data association; (2) evaluation of alternative hypotheses in multiple hypothesis trackers; (3) fuzzy-logic-based pattern recognition (e.g., for feature-based object identification); and (4) fuzzy inference schemes for sensor resource allocation. (Stover et al. (1996))

There are many more fundamental contributions to the field by others, the above list can only be a small cross section due to limited space.

2.2 Comments on the state of the art

This section draws some conclusions from the achievements made.

(1) Probably it is fair to say that some of the AICS built have not been tested in an adequately formal experiment and their behavior reported in detail to appreciate how well they perform. Perhaps the reason is that a lot of engineering effort goes into the construction and little energy is left for formal analysis, also journal papers have limited space.

(2) Many of the autonomous systems reported operate in narrowly defined universes. Many of the methods do not easily generalize to other environments and the systems appear to be fragile to variable environmental conditions.

(3) Some computer vision techniques and 3D modelling have lately advanced a great deal and are not in full use by AICSs.

(4) A lot of progress has been made in various architectures for programming systems. Authors rarely follow the path started by others (which is not characteristics in other areas of control!) Publications are "advertisements" of the "excellent" work that individuals or teams obtained with their autonomous systems. This is also supported by the desire of publishing original material which is easier to achieve with a new architecture than by

developing on someone else's system further. This leads to an ever increasing diversity of hardware architectures and programming approaches since authors have difficulty to develop each other's results further, due to lack of detailed information. This slows down real progress.

(5) For some of the robot projects the deadlines of robot competitions encourage teams to do ad hoc fixes instead of doing a deeper analysis of the fundamental issues.

(6) Some of the results do a deep analysis starting from analyzing animal and human behaviour and in time they may turn out to have pointed to the right direction for a general architecture of AICS.

It appears that all the applications could be handled by a universal AICS if we knew how to build a universal one. Until it becomes clear what is the right direction in the a generic functional architecture and programming, *it would help progress if we could agree on an a hardware architecture that would allow researchers to build on each other's algorithmic results.*

3. THE FUNCTIONAL ARCHITECTURE OF MIBS

As many of the blocks and algorithms exist today, it is an organizational job to bring them together, it is clear that we need:

(1) a universal hardware component to receive signals from hundreds of sensors in parallel (SENSOR module); that should contain a component that is capable of handling a set of cameras or other vision devices (>2) and produce 3D models of the current physical environment in realtime (VISION module) and a hardware component that has the capacity to run software to model spatial distribution of sound sources and signals in realtime (HEARING module);

(2) a universal hardware component that is capable of generating signals for hundreds of actuators in parallel (ACTIONS module);

(3) a universal hardware component that is capable to run thousands of treads of algorithmic procedures in parallel (THINKING module);

(4) a network hardware component that connects all the modules and allows for very fast communication (SPINAL module).

These 6 components can form the hardware architecture of the shared MIB. Basic software support is also needed in the form of a realtime operating system that contains device drivers for a range of sensing and actuator devices that typically occur in robotics applications. Self-configuration is fundamentally important as an operating sys-

tem problem here, as we want each component to "work out" its role and connectivity with the other components automatically, so that no significant engineering effort would be needed for installation of components. Instead, building the hardware, engineers should focus on designing intelligent behaviour. In terms of software technology this needs a well defined agent behaviour from each of the modules and is technologically well within reach today.

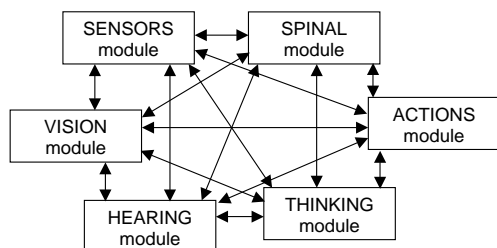


Fig. 1. Fast connectivity in every way to allow for any new idea to be implementable by researchers.

Technical standards are needed for

- (1) for communications protocol standards between the main modules and
- (2) for physical electronic connectivity, i.e. socket types, line allocations and power sources, etc.

4. FUNCTIONAL ANALYSIS OF MAIN COMPONENTS

The requirements listed here are intentionally non-restrictive. It is the communication protocols between modules that have to be fixed. What algorithms are run in the individual models can be changed by development research engineers at companies or universities, who want to experiment with the basic architecture.

Sensor module. A shared *library of data fusion algorithms* for divers sensor data needs to be provided. Researchers would be allowed to use any of the algorithms or develop their own and document and publish it by placing it into the shared library.

Vision module. An identical vision processing module with fixed communication protocols, but any number of cameras, would be used by all researchers involved. The vision based spatial modelling procedures would be collected in a shared library by those researchers who are interested to share their results. Agreed standards of object representations of spatial modelling objects are needed.

Hearing module. Spatial analysis of sound and separation of voice from hundreds of noise sources may be almost as complex as vision processing. The artificial hearing based spatial modelling procedures would be collected in a shared library by those researchers who are interested to share their results. Agreed standards of representations of spatial hearing model objects are needed.

Actions module. A motor control memory (action sequence memory) and a shared library of control procedures that can have inputs from any of the other modules. An agreed (1) communication protocol with other modules and (2) agreed representations of control action objects is also needed.

Thinking module. As at this stage we are uncertain about the most successful generic way of organizing an AICS, this module should enjoy the largest degree of freedom. Its only constraints are its communications protocol with each of the other modules and its ability to run a large number of threads in parallel under a realtime operating system with a user friendly programming system in an agreed language.

Spinal module. The spinal module is the most interesting and deviates from what its name suggests as it acts like a fast self-organizing communications network ("nervous system"). Recent results in the network area allow the realization of this software and hardware. The software pieces of the network are actually part of the operating system and cannot be changed by the developers/users of the MIB system.

An advanced but optional feature could be the requirement for each module that: when the module is re-programmed by a developer then a symbolic representation of the new software module's behavioural properties should be symbolically described (with some agreed logic formalism). This would allow for possible faultless self-configuration of the whole system (by the spinal module) and by modules asking each other about their functionality.

5. CONCLUSIONS

The field of autonomous intelligent control systems has been reviewed and a proposal was made for a shared architecture with some basic software platform also shared by the research community.

REFERENCES

- P. Arenaa, P. Di Giamberardinob, L. Fortunaa, F. La Galab, G. Muscatoa S. Monacob, A. Rizzoc, and R. Ronchinib. Toward a mobile autonomous robotic system for mars exploration. *Planetary and Space Science*, 52, 2004.

- Magali Barbier and Elodie Chantry. Autonomous mission management for unmanned aerial vehicles. *Aerospace Science and Technology*, 8, 2004.
- R. P. Bonasso. Using parallel program specifications for reactive control of underwater vehicles. *Journal of Applied Intelligence*, 2, 1992.
- Alberto Broggi, Massimo Bertozzi, and Alessandra Fascioli. Architectural issues on vision-based automatic vehicle guidance: The experience of the argo project. *Real-Time Imaging*, 6, 2000.
- R. A. Brooks. *Cambrian intelligence, the early history of the new AI*. MIT Press, Cambridge, MA, 1999.
- Wolfram Burgard, Armin B. Cremers, Dieter Fox, Dirk Hhnel, Gerhard Lakemeyer, Dirk Schulz, Walter Steiner, and Sebastian Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114, 1999.
- A. Chella, M. Frixione, and S. Gaglio. A cognitive architecture for artificial vision. *Artificial Intelligence*, 89, 1997.
- Pooya Farahvash and Thomas O. Boucher. A multi-agent architecture for control of agv systems. *Robotics and Computer-Integrated Manufacturing*, to appear, 2004.
- H. Frezza-Buet and F. Alexandre. From a biological to a computational model for the autonomous behavior of an animal. *Information Sciences*, 144, 2002.
- E. Gat. Three layer architectures. *Artificial Intelligence and Mobile Robots, Kortenkamp, Bonaso and Murphy (Eds.)*, AAAI Press, Menlo Park, CA, 1998.
- Hani Hagras, Victor Callaghan, Martin Colley, and Graham Clarke. A hierarchical fuzzy-genetic multi-agent architecture for intelligent buildings online learning, adaptation and control. *Information Sciences*, 150, 2003.
- A.J. Healey, D.B. Marco, and R.B. McGhee. Autonomous underwater vehicle control coordination using a tri-level hybrid software architecture. *Proceedings of the IEEE international conference on robotics and automation*, Minneapolis, MN, 1998.
- Huosheng Hu, Dongbing Gu, and Michael Brady. A modular computing architecture for autonomous robots. *Microprocessors and Microsystems*, 21, 1998.
- T.W. Kim and J. Yuh. Autonomous vehicle taxonomy. *Proc. of the symposium on autonomous underwater vehicle technology*, 1992.
- T.W. Kim and J. Yuh. Development of a real-time control architecture for a semiautonomous underwater vehicle for intervention missions. *Control Engineering Practice*, 12, 2004.
- David Kortenkamp, R. Peter Bonasso, and Robin Murphy (Eds.). *Behaviour-based robotics*. MIT Press, Cambridge, MA, 1998.
- Gen Matsumoto and Hiroshi Tsujinob. Design of a brain computer using the novel principles of output-driven operation and memory-based architecture. *International Congress Series*, 1250, 2003.
- David J. Musliner, Edmund H. Durfee, and Kang G. Shin. World modeling for the dynamic construction of real-time control plans. *Artificial Intelligence*, 74, 1995.
- A.A. Pankine, K.M. Aaron, M.K. Heun, K.T. Nock, R.S. Schlaifer, C.J. Wyszowski, A.P. Ingersoll, and R.D. Lorenz. Directed aerial robot explorers for planetary exploration. *Advances in Space Research*, 33, 2004.
- R. Pfeifer and C. Scheir. *Understanding intelligence*. MIT Press, Cambridge, MA, 1999.
- P. Remagnino, A.I. Shihab, and G.A. Jones. Distributed intelligence for multi-camera visual surveillance. *Pattern Recognition*, 37, 2004.
- Pere Ridao, Joan Batlle, and Marc Carreras. O2ca2, a new object oriented control architecture for autonomy: the reactive layer. *Control Engineering Practice*, 145, 2002.
- Thomas Schetter, Mark Campbell, and Derek Surka. Multiple agent-based autonomy for satellite constellations. *Artificial Intelligence*, 145, 2003.
- V. Sequeira, K. Ng, E. Wolfart, J.G.M. Goncalves, and D. Hogg. Automated reconstruction of 3d models from real environments. *Journal of Photogrammetry & Remote Sensing*, 54, 1999.
- Stuart C. Shapiro and Haythem O. Ismail. Anchoring in a grounded layered architecture with integrated reasoning. *Robotics and Autonomous Systems*, 43, 2003.
- J.A. Stover, D.L. Hall, and R.E. Gibson. A fuzzy-logic architecture for autonomous multisensor data fusion. *IEEE Transactions on Industrial Electronics*, 43, 1996.
- Hartmut Surmann, Andreas Nchter, and Joachim Hertzberg. An autonomous mobile robot with a 3d laser range finder for 3d exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, 45, 2003.
- Aline Carneiro Viana, Marcelo Dias de Amorim, Serge Fdida, and Jose Ferreira de Rezende. Self-organization in spontaneous networks: the approach of dht-based routing protocols. *Ad Hoc Networks*, to appear, 2004.
- Z.D. Wang, T. Takahashi, T. Nitsuma, T. Ninjouji, and E. Nakanob. Logue: an architecture for task and behavior object transmission among multiple autonomous robots. *Robotics and Autonomous Systems*, 44, 2003.
- G. Yasuda. Distributed autonomous control of modular robot systems using parallel programming. *Journal of Materials Processing Technology*, 141, 2003.