Proceedings of the World Congress on Engineering and Computer Science 2010 Vol I
WCECS 2010, October 20-22, 2010, San Francisco, USA

An Autonomous Robot Prototype using Concept Learning Model

Gabriel M. Barrera, Juan J. Iglesias, and Daniela López de Luise

Abstract— **Autonomous mobile robots require an efficient movement controller. Several very well-known techniques are used. This paper focuses on the implementation of a second controller that complements the usual close loop controller, providing further information about current environment and suggesting actions to the main controller. This advisory controller implements an adaptation of the Conscious modeling designed by Dr. Stan Franklin at Memphis University and its implementation as IDA and LIDA models.**

Index Terms— **Concept Learning, Robotics, Mobile robots.**

I. Introduction

Mobile robots could be studied in an outdoor or indoor environment. This paper is focused on the processing of main indoor movement parameters (wheel speed, world map characteristics, delays, etc.) in order to use them in a newer approach based on concept learning. The main goal is to describe a proposal of a feedback system able to advise a short loop robot controller. FIC prototype is described here, an autonomous indoor mobile robot controller adviser provided with a dynamic environment and obstacle conceptualization.

An autonomous mobile robot must dynamically navigate through its changing environments. But, this could require a large collection of environment information, storage and processing capabilities. All this information handling could require too much computation for a real-time implementation like this one. To overcome that problem, we propose a decentralized architecture with a dual-loop feedback system, one for immediate displacement decisions and other for middle to long-term strategies.

An autonomous mobile robot must dynamically navigate through its changing environments. But, this could require a large collection of information, storage and processing capabilities. All this data handling could require too much computation for a real-time implementation like this one. To overcome that problem, we propose a decentralized architecture with a dual-loop feedback system, one for immediate displacement decisions and other for middle to long-term approaches.

Traditional localization in places such as factory buildings, halls, etc. usually involves processing a set of events and facts, such as natural landmarks, walls, dynamic obstacles, etc. To solve such localization problems there are several alternatives, using ceiling features [20] [21] with a perceptual model for Monte Carlo Localization (MCL)[11], to match well-known ceiling structures, like beams, columns, lighting installation, etc, against a predefined world model. This general approach will be described in section II in more detail.

The authors are with AIGroup, Universidad de Palermo.
Gabriel Barrera is a systems engineer and coordinator for research lines at University of Palermo (e-mail: gmbarrera@gmail.com).

This project contributes to existing knowledge from two points of view:

a) Autonomous mobile technology is a new approach that improves static world descriptions from actual alternatives [21] [11] and dynamic conceptualizations [22] or [23], by using a more flexible environment model based on dynamic world construction. In this model, the obstacles and external references are not necessarily of the same type (i. e. doors, edges or roof characteristics), nor have they been previously physically sensed and mapped. With this model the robot can recognize new obstacles not represented in its previous world map. FIC could also be targeted dynamically to discriminate and recognize obstacles within the same concept, such as dangerous or preferred obstacles.

b) Concept technology: LIDA (Learning Intelligent Distribution Agent) concept model [24] first presented by S. Franklin, is implemented into the AlgOC framework, which is based on Copycat Architecture [15] and Flexo [25] framework, thus opening the possibility of a more cognitive robotic system with broader perceptual capabilities as well as the ability to learn.

In the following this paper presents a description about the basic features of the Concept Model (section II), present a general overview of the prototype architecture for the FIC project (section III), and actual state of the project (section IV).

II. Concept Model

LIDA is an IDA (Intelligent Distribution Agent) spring-off. IDA was conceived as an autonomous software agent that automates tasks performed by human resource personnel called detailers [1] [2]. In contrast to IDA, LIDA (Learning IDA) is capable of several forms of learning.

LIDA model intends to covering a large portion of human cognition. It implements and fleshes out a number of psychological and neuropsychological theories including situated cognition [4], perceptual symbol systems [5], working memory [6], memory by affordances [7], long-term working memory [8]. Sloman's cognitive architecture [9] and transient episodic memory [10].

The LIDA cognitive cycle states that every autonomous agent [12], be it human, animal, or artificial, must frequently sample (sense) its environment and select an appropriate response (action). Every cycle has three components: a unit of sensing, attending and acting. A higher-level cognitive process is just an aggregation of many cognitive cycles (or cognitive "atoms.")

Main core of LIDA model will now be described. Several very precise descriptions are available elsewhere [13] [14].

To start a cognitive cycle the LIDA agent first makes sense of its current situation. To do that it updates its

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

external and internal world representation. After a competitive process it decides what portion of the represented situation requires prior attention. This portion, the current contents of consciousness, takes influence in the coice of an appropriate action to be executed.

After sensing external and internal stimuli, these low-level features are passed on to perceptual memory (depicted in the figure, and implemented, as a slip-net[15]). Its main function it to recognize higher-level features (such as objects, categories, relations, situations, etc.) These entities make up the percept that is passed to the workspace. At this time, it builds a model of the agent's current situation. Percepts could fire two forms of episodic memory: transient and declarative. Upon activation, they are able to start distinct sets of events. In addition to the current percept, the workspace contains recent percepts. It also holds certain contemporary models assembled from them (those that still represent a valid current situation. In other words, those that haven't yet decayed away).

The model of the agent's current situation is updated using percepts, their associations, and undecayed parts of the previous model. This activity is responsibility of a special subset of codelets (named structure-building codelets [15].) They use perceptual memory and sensory memory, to detect relations and situations. The updated model constitutes the agent's understanding of its current situation within its world.

To continue with the cognitive cycle, different portions of the model compete. Such coalitions are formed by other set of codelets (attention codelets). When a coalition wins the competition, it is said that the agent has decided on what to attend. Procedural memory, receives contents of the winner and stores templates of possible actions including their contexts, possible results and its activation value.

Then, an action is taken. To select which one to perform, it copares templates' context with winner's. The selected is the one that best matches. Templates remaining from previous cycles may also continue to be available. The action selection mechanism chooses a single action from one of these instantiations. The chosen action then goes to sensory-motor memory, where it picks up the appropriate algorithm by which it is then executed. The action taken affects the environment, and the cycle is complete.

III. FIC PROTOTYPE

This section will present a general overview of the prototype architecture, starting from the lightweight adaptation of LIDA and FLEXO models, following with aim structure description, and specific information about its implementation.

A. AlgOC

Based on LIDA model, a framework to simplify the task to work with concepts has been developed, that's the mail goal of AlgOC (Algoritmo Orientado a Conceptos, Concept Oriented Algorithm).

AlgOC basically has three main components: the Slipnet, the Workspace, the Coderack (which contains a collections of Codelets), providing the necessary means to enter the domain structure, through related objects. The relations between these components are show in figure 1.

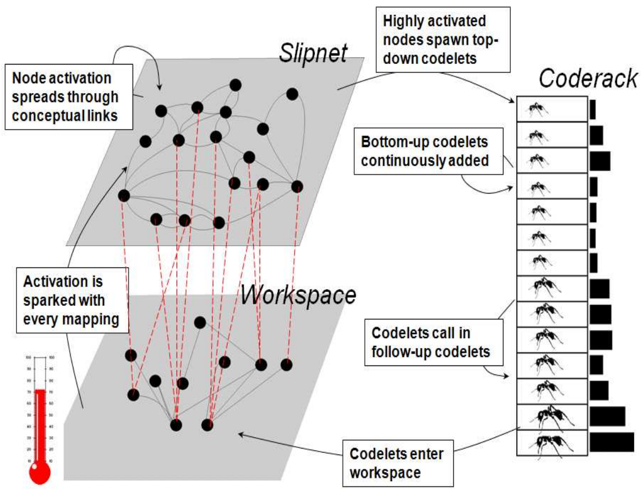

Fig. 1: AlgOC main components

The Slipnet makes the user can get an idea of nodes, links and all their properties, since it is a network of nodes connected by concepts or conceptual relations. The nodes have a degree of activation, which may rise or fall, reflecting the importance of the concept in the problem. When activation reaches a ceiling, the concept is triggered. Over time, the activation decreases gradually, if the concept is not invoked again. Relationships can change the conceptual distance, marking the dependence between two concepts.

The Workspace allows you to define the structures that can also be constructed by Codelets, the joint structures have to be reused and modified. At first, there are a large number of equipment manufacturers who all have the same set of blocks, ie the basic concept of the problem. Then they change to build more refined structures, starting at different times but working in parallel. The structures vary to be sufficiently convincing, or concepts are similar to those found in the Slipnet. It should be noted that these new structures may result from the union or division of larger structures.

The Coderack module is where the set of control algorithms is, it must also accept the burden of Codelets defined by the user, and all its parameters should be easily modified. The Codelets perform small actions in the workspace, while maintaining its hierarchy. There are some basic kinds of Codelets where each is most appropriate for a particular structure, but each user can create their own types of Codelets to the specific problem being treated. Self-control modules should be easy to add and use, as they are responsible for monitoring the behavior of the system.

The Codelets are non-deterministic computational agents that carry small pieces of code.

A model of the same was implemented in the Copycat project Hoftadter Douglas and Melanie Mitchell [16] which was based James B. Marshall to develop Metacat [17][18] as an improved version of the above.

Also Stan Franklin and his group of researchers developed a more complex model based on this scheme. One of the projects where this technique is used IDA (Intelligent Distribution Agent) [19] that assigns staff to the United States Navy according to their preferences and the policies

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

of this security force.

Each Codelet has a value assigned emergency or activation, which varies according to the frequency with which they are called, and are selected to run according to this value. Which has a higher chance of being selected form the Coderack, where entering and exiting in a pseudo-random and remain there until their execution is triggered [17].

The whole process occurs through the collective actions of many Codelets working in parallel, at different speeds in different aspects of the problem, without any centralized control over the course of events.

The Codelets may be called by another agent or any entity you feel the need to execute them, since they are the responsible for proposing new structures or to evaluate the proposed structures to generate new agents, which will continue the process in the next stage

With all this Slipnet can be seen as a long-term memory and the Workspace, then, is a short-term memory where it is the current problem, so there must be a back and forth between all the components to distinguish is really useful and continually test their quality, especially perceptual activity (Workspace) and conceptual activity (Slipnet).

B. *The FIC Architecture*

The overall architecture of FIC prototype can be seen in figure 2. Here, the robot is connected to the Real Time Controller (RTC) which is interacts with FIC. As FIC was designed to be a framework, the actual RTC communicates to FIC just as another device.

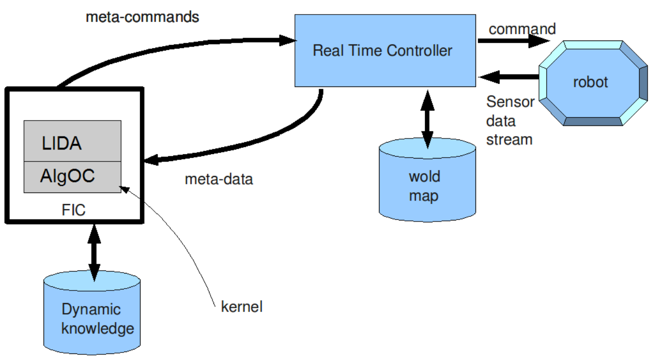

Fig. 2: FIC Architecture

It is responsibility of RTC to provide a correct World Map (WM) initial information and to handle it. FIC will manage to build an internal copy of this map within its Dynamic Knowledge local Database (DBD), and to perform the proper updates to it using data inputted from RTC.

FIC acts as an interface between Real Time Controller (RTC) that controls the robot, and the kernel. RTC is supposed to work using a static map of the world and taking decisions about robot activity. Eventually it will accept additional command information coming from the FIC.

While the main storage for RTC is the WM, the FIC's is DBD, to hold key information for life cycle management and specific data to improve performance.

C. *Implementation considerations*

The FIC architecture was defined in order to accomplish concept handling and meta-command interchange though the dual-loop feedback. As the computational requirements for AlgOC are relatively high, the hardware restrictions have been defined for robot, Controller and FIC respectively. Each one has an extra load of computation due the dual-loop. To guarantee a minimum performance and compatibility with real time answers, the following requirements must be filled:

1) FIC: It requires Java JRE v1.6.0, minimum hard disk storage of 50 Mb and a standard processor of 1.2 Ghz.
2) Restrictions for controller: It requires same as FIC, but with a property connection with the robot.
3) Restrictions for robot: Any robot that can understand meta-commands. Our test case is a Lego NXT armed robot with two independent wheels.

IV. Conclusion & Future Work

The overall architecture of the FIC prototype has been presented. FIC components perform specific and complementary tasks required to know actual robot localization and to proceed with certain movement. The movement is decided as a consequence of a combination of a short loop and long loop control.

There are new modules serving to implement part of LIDA complementing and extending FIC's abilities. These are still under a tuning process. It is expected to let the robot learn dynamically new obstacles and to adapt

We are in a stage of tuning the first prototype, including the AlgOC framework and the robot controller.

Next, is to extend the ability of the robot to learn new obstacles and eventually build a secure path for him. And as pending work remains to include LIDA's functionalities. This will make FIC able to set initial knowledge of the world as a automatic construction of a map after a previous go-around of the robot, and to re-evaluate constantly the need of updating original world map.

References

[1] Franklin S. “An Autonomous Software Agent for Navy Personnel Work: a Case Study”. 2001.
[2] McCauley L., Franklin S. "A Large-Scale Multi-Agent System for Navy Personnel Distribution". Connection Science 14:371-385. 2002.
[3] Baars, B. J. “A Cognitive Theory of Consciousness.” Cambridge: Cambridge University Press. 1988.
[4] Varela A. H., Thompson E., Rosch E. "The Embodied Mind: Cognitive Science and Human Experience". Cambridge, MA: MIT Press. (1991).
[5] Barsalou L.W. "Perceptual symbol systems". Behavioral and Brain Sciences, 22, 577-609.(1999).
[6] Baddeley A.D., Hitch G."The psychology of learning and motivation: Advances in research and theory". "Working memory". In G.H. Bower (Ed.) (Vol. 8, pp. 47--89). New York: Academic Press. (1974).
[7] Glenberg A. M. "What memory is for". Behavioral and Brain Sciences, 20, 1-19. (1997).
[8] Ericsson K. A., Kintsch W. "Long-term working memory". Psychological Review, 102, 211-245.(1995).

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

[9] Posadas, J. L., Poza, J. L., Simó, J. E., Benet , G., & Blanes, F. (2008). Agent-based distributed architecture for mobile robot control. Engineering Applications of Artificial Intelligence, 21(6), 805-823.

[10] Conway M. A. "Sensory-perceptual episodic memory and its context: Autobiographical memory". In A. Baddeley, M. Conway, & J. Aggleton (Eds.), Episodic Memory. Oxford: Oxford University Press.(2001).

[11] Eberst C., Andersson M., Christensen H.I., Vision-based door-traversal for autonomous mobile robots". Inst. for Real-Time Comput.-Syst., Tech. Univ. Munchen; Proc. 2000 IEEE/ RSJ International Conference on Intelligent Robots and Systems.

[12] Franklin S, Graesser A. "Is it an Agent, or just a program?: A Taxonomy for Autonomous Agents". Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages, Springer-Verlag, 1996.

[13] Baars B. J., Franklin S. "How conscious experience and working memory interact". Trends in Cognitive Science 7:166–172. 2003

[14] Franklin S., Baars B. J., Ramamurthy U., Ventura M. "The Role of Consciousness in Memory". Brains, Minds and Media 1:1-38. 2005.

[15] Hofstadter, D. R. and Mitchell, M. "The Copycat Project: A model of mental fluidity and analogy-making." In Holyoak, K.J. & Barnden, J.A. (Eds.) Advances in connectionist and neural computation theory, Vol. 2: Analogical connections. Norwood, N.J.: Ablex. 1994.

[16] Hofstadter D. R., Mitchell M., "The Copycat Project: A Model of Mental Fluidity and Analogy-Making", in Fluid Concepts and Creative Analogies: Computer Models of the Fundamental Mechanisms of Thought, Ed. New York: Basic Books, 1995, pp. 31-112.

[17] Marshall J. B., "Metacat: A Self-Watching Cognitive Architecture for Analogy-Making and High-Level Perception", Indiana University, USA, 1999.

[18] Marshall J. B., Hofstadter D. R., "The Metacat Project: A Self-Watching Model of Analogy-Making", Indiana University, USA, 1997.

[19] Franklin S., Kelemen A., McCauley L., "IDA: A Cognitive Agent Architecture", Proc. 1998 IEEE Conf on Systems, Man and Cybernetics.

[20] Kolski S., ferguson D., Bellino M., Siegwart R., "Autonomous Driving in Structured and Unstructured Environments". IEEE Intelligent VehiclesSymposium. 2006.

[21] Lecking D., Wulf O., Wagner B., "Localization in a wide range of industrial environments using relative 3D ceiling features".

[22] Zendera H., Martínez Mozosb O., Jensfeltc P., Kruijffa G., Burgardb W. "Conceptual spatial representations for indoor mobile robots", German Research Center for Artificial Intelligence (DFKI GmbH). 2008 Elsevier.

[23] MacPherson D.L., "Automated Cartography by an Autonomous Mobile Robot Using Ultrasonic Range Finders". PhD Thesis. 1993 Naval Postgraduate School, Monterey CA.

[24] Franklin S., Patterson F G Jr, "The LIDA architecture: Adding new modes of learning to an intelligent, autonomous, software agent". In IDPT-2006 Proceedings (Integrated Design and Process Technology): Society for Design and Process Science. 2006.

[25] Vanschoren J., "Development of a framework for high-level perception", Master's thesis. 2005.

ISBN: 978-988-17012-0-6
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)