Developing a Consciousness-based Mind for an Artificial Creature

Ricardo Capitanio Martins da Silva[1] and Ricardo Ribeiro Gudwin[1]

DCA-FEEC-UNICAMP
Av. Albert Einstein 400, 13.083-852 Campinas, SP,
{martins,gudwin}@dca.fee.unicamp.br

Abstract. This work describes the application of the Baars-Franklin Architecture (BFA), an artificial consciousness approach, to synthesize a mind (a control system) for an artificial creature. Firstly we introduce the theoretical foundations of this approach for the development of a conscious agent. Then we explain the architecture of our agent and at the end we discuss the results and first impressions of this approach.

1 Introduction

The scientific study of consciousness has improved dramatically in the last ten years [1]. A technological offspring of these studies is the field of artificial consciousness [2,3,4]. In this work we concentrate in what we call here the Baars-Franklin architecture (BFA). The BFA is a computational architecture being developed by the group of Stan Franklin, at the University of Memphis [5,2,6], based on the model of consciousness given by Bernard Baars, called Global Workspace Theory [7].

The BFA has already been applied to many different kinds of software agents. The first application of BFA was CMattie [5,2], an agent developed by the Cognitive Computing Research Group (CCRG) at the University of Memphis, whose main activities were to gather seminar information via email from humans, compose an announcement of the next week's seminars, and mail it to members of a mailing list. Through the interaction with human seminar organizers, CMattie could realize that there was missing information and ask it via email.

The overall BFA received major improvements with subsequent developments. One remarkable implementation of it was IDA (Intelligent Distribution Agent) [6], an application developed for the US Navy to automate an entire set of tasks of human personnel agent who assigns sailors to new tours of duty. IDA is supposed to communicate with sailors via email and, in natural language, understand the content and produce life-like messages.

The BFA was also used outside of Franklin's group. Daniel Dubois from University of Quebec developed CTS (Conscious Tutoring System) [8], a BFA-based autonomous agent to support the training on the manipulation of the International Space Station robotic control system called Canadarm2.

Nevertheless, up to our knowledge, BFA was never used to implement a mind (a control system) for an artificial virtual creature. Artificial Creatures are

a special kind of agents, embodied autonomous agents which exists in a certain environment, moving itself in this environment and acting on it [9]. Artificial creatures may be real or virtual. Examples of real artificial creatures are robots acting in the real environment. Virtual Artificial Creatures are software agents living in a virtual world, where they are able to sense and actuate by means of an avatar (a virtual body). One example of a virtual artificial creature is an intelligent opponent in a computer game, where an intelligent control system must decide the actions to be performed by the agent in order to foster a good entertainment to the system user, simulating with realism the behavior of a human opponent. Other examples of virtual artificial creatures include ethological simulation studies, in artificial life, where tasks such as foraging and sheltering are very common.

Virtual artificial creatures pose some interesting research problems when compared to other kinds of software agents where BFA has already been tested. In the original applications where BFA was tested, the perception system is based on the exchange of e-mail messages (the case of CMattie and IDA), and interactions in a HCI (human-computer interface), in the case of CTS. In a virtual artificial creature, perception must rely on remote (e.g. visual, sonar, etc) and/or local (e.g. contact) sensors, capturing properties of the scenario and interpreting them in order to create a world model. The behavior generation module is also different, as the agent must act on itself (its body) and over things on the environment. The main motivation for the research reported in this work is though to investigate how the use of BFA may impact the control of a virtual artificial creature, and what are the benefits which can be expected.

In the next section, we introduce briefly Baars' theory of consciousness, Global Workspace Theory, and then we describe how we customized BFA in order to deal with virtual artificial agents. After that, we introduce CAV (Conscious Autonomous Vehicle), the artificial creature we used in our study and its environment, and a brief analysis of the results of our simulations using CAV.

2 Global Workspace Theory and BFA

Bernard Baars has developed the Global Workspace Theory (GWT) [7] inspired by psychology and based on empirical tests from cognitive and neural sciences. GWT is an unifying theory that puts together many previous hypothesis about the human mind and human consciousness.

Baars postulates that processes such as attention, action selection, automation, learning, meta-cognition, emotion, and most cognitive operations are carried out by a multitude of globally distributed unconscious specialized processors. Each processor is autonomous, efficient, and works in parallel and high speed. Nevertheless, in order to do its processing, each processor may need a set of resources (mostly information of a specific kind), and at the same time, will generate another set of resources after its processing. Specialized processors can cooperate to each other forming coalitions. This cooperation is by means of supplying to each other, the kinds of resources necessary for their processing. They exchange resources by writing in and reading from specific places in

working memory. Coalitions may form large complex networks, where processors are able to exchange information to each other. But processors within a coalition do have only local information. There may be situations, where the required information is not available within the coalition. To deal with these situations, and allow global communication among all the processors, there is a global workspace, where processors are able to broadcast their requirements to all other processors. Likewise, there may be situations where some processor would like to advertise the resource it generates, as there may be other processors interested in them. They will also be interested in accessing the global workspace and broadcasting to all other processors. In the broadcast dynamics, only one coalition is allowed to be within the global workspace in a given instance of time. In order to decide which coalition will go to the global workspace in a given instant of time, a whole competition process is triggered. Each processor has an activation level, which expresses its urgency in getting some information or the importance of the information it generates. A coalition will also have an activation level which is the average of activation levels of its participants. At each time instant, the coalition with the highest activation level will win the access to the global workspace. Once a coalition is within the global workspace, all its processors will broadcast their requests and the information they generate. The broadcast mechanism do allow the formation of new coalitions, and also some change in working coalitions.

For Baars, consciousness is related to the working of this global workspace. Processors are usually unconscious, having access only to local information, but in some cases they may require or provide global information, in which case they request access to consciousness, where they will be able to broadcast to all other processors. This is the case when they have unusual, urgent, or particularly relevant information or demands. This mechanism supports integration among many independent functions of the brain and unconscious collections of knowledge. In this way, consciousness plays an integrative and mobilizing role. Moreover, consciousness can be useful too when automatized (unconscious) tasks are not being able to deal with some particular situation (e.g. they are not working as expected), and so a special problem solving is required. Executive coalitions, specialized in problem solving will be recruited then in order to deal with these special situations, delegating trivial problems to other unconscious coalitions. In this way, consciousness works like a filter, receiving only emergencial or specially relevant information.

Inspired by Baars description of his theory of consciousness, and also by previous work in the computer science literature, Franklin proposed a framework for a software agent which realized Baars theory of consciousness, in terms of a computational architecture, constituting so what we are calling here the Baars-Franklin architecture. In specifying BFA, Franklin used the following theories as background, among others not detailed here: Selfridge's Pandemonium [10] and Jackson's extension to it [11], Hofstadter and Mitchell Copycat [12] and Maes' Behavior Network [13].

From Hofstadter's Copycat, Franklin borrowed the notion of a "Codelet" (and also the Slipnet, for perception). He noticed that these *codelets* were more or less the same thing as Selfridge's "demons" in Pandemonium theory and also a good computational version for Baars *processors*. Jackson's description of an arena of *demons* competing for selection will fit as well Baars description of processors competing in a *Playing Field* for access to consciousness. Using these similarities, Franklin set up the basis of BFA: cognitive functions are performed by coalitions of codelets working together unconsciously, reading and writing tagged information to a Working Memory. Each codelet has an activity level and a tagged information. A special mechanism, the *Coalition Manager* will manage coalitions and calculate the activity level of each coalition. Another special mechanism, the *Spotlight Controller*, will be evaluating each coalition activity level, and defining the winning coalition. Also, the *Spotlight Controller* will be responsible for performing the broadcast of the tagged information of each codelet in the winning coalition, to all codelets in the system. The agent behavior is decided using a Behavior Network, whose propositions are related to the tagged information in the Working Memory.

Unfortunately, a full description of BFA is beyond the space available in this text. We refer the interested reader to [2,6,8,14], where a more detailed description of BFA is available.

3 Our implementation of BFA

In our experiment, we developed an artificial mind (a control system), which we call CAV - *Conscious Autonomous Vehicle*, to control an artificial creature in a virtual environment (see figure 1). The creature and its environment were originally presented in [15] (where more details on its characteristics can be obtained) and were adapted for our current studies. In this environment, the creature is equipped with sensors and actuators, which enable it to navigate through an environment full of objects with different characteristics. An object can vary in its "color" and each color is linked to: a measure of "hardness" which is used in the dynamic model as a friction coefficient that can slow down the creature's movement (or completely block it), a "taste" which can be bad or good, and a feature related with "energy" which indicates that the object drains/supplies energy from/to the creature's internal rechargeable battery.

The creature connects to its mind through sockets. In this sense, the artificial mind is a completely separate process, which can be run even in a different machine. So, different minds can be attached to the creature and tested for the exact same situation.

When the simulation is started, the creature builds an incremental map of the environment based on the sensory information. Our agent adds landmarks to this map and uses them to generate movement plans. It has two main motivations: it should navigate from an initial point up to a target point, avoiding collisions with objects; and it should keep its energetic balance, taking care of the energy level in the internal batteries.

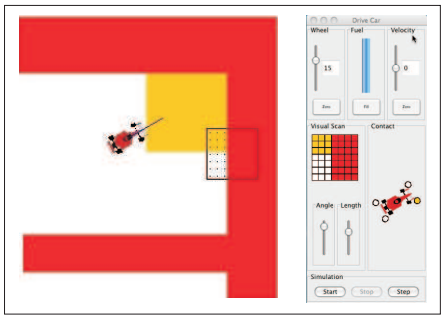


Fig. 1. Sensory-motor structure of the creature

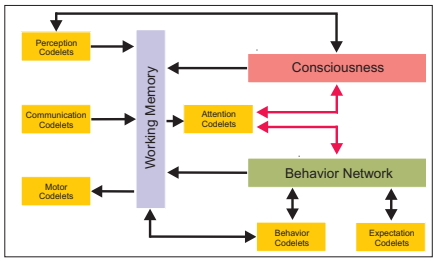


Fig. 2. *CAV*'s Architecture

Our architecture (see figure 2) is essentially rooted in the BFA implementation as in [2] (consciousness) and [6] (behavior network). *CAV* brings some modifications in the implementation related with the application domain, and the interaction among consciousness and behavior network. The following sections contain a brief description of CAV's modules.

3.1 Codelets

CAV is heavily dependent on small pieces of code running as separate threads called codelets (BFA borrows this name from Hofstadter's Copycat). Those codelets correspond pretty well to the specialized processors of global workspace theory or demons of Jackson and Selfridge.

BFA prescribes different kinds of codelets such as attention codelets, information codelets, perceptual codelets and behavior codelets. In addition to that, it is possible to create new types of codelets depending on the problem domain. CAV's domain does not require string processing as do most other BFA applications. Instead of that, the creature state is well divided in registers at the working memory. It is possible to have access to all variables anytime. Because of this, CAV does not use information codelets which in BFA are used to represent and transfer information. We have two kinds of behavioral codelets: the behavior codelets, linked with the nodes of the Behavior Network and responsible for "what to do", and motor codelets, which know "how to act" on the environment. With this in mind CAV has the taxonomy of codelets presented at Table 1.

3.2 Working Memory

The working memory consists of a set of registers which are responsible for keeping temporary information. The major part of the working memory is related to the creature status. The communication codelet constantly overwrites the registers like speed, wheel degree, sensory information and creature position. CAV's working memory works also as an interface among modules, for example, between consciousness and the behavior network. Some codelets, including attention codelets watch what is written in the working memory in order to find relevant, insistent or urgent situations. When they find something, they react in

Table 1. *CAV's Codelets Taxonomy*

Type	Role
Communication	Perform the communication with the simulator, bringing novel simulation information
Perception	Give an interpretation to what the agent senses from its environment
Attention	Monitor the working memory for relevant situations and bias information selection
Expectation	Check that expected results do happen
Behavior	Alter the parameter of the motor codelet
Motor	Act on the environment

order to compete for consciousness. Whenever one of then reaches consciousness, its information will influence the agent's actions.

3.3 Consciousness mechanism

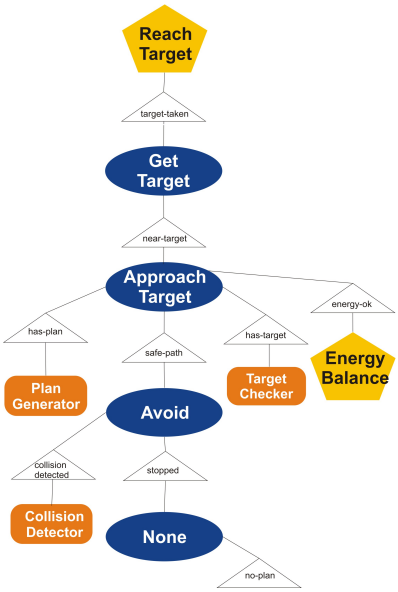
The consciousness mechanism consists of a *Coalition Manager*, a *Spotlight Controller*, a *Broadcast Manager* and attention codelets which are responsible for bringing appropriate contents to "consciousness" [2]. In most of the cases, codelets are observing the working memory, looking for some relevant external situation (e.g. a low level of energy). But some codelets keep a watchful eye on the state of the behavior network for some particular occurrence, like having no plan to reach a target. More than one attention codelet can be excited due to a certain situation, causing a competition for the spotlight of consciousness. If a codelet is the winner of this competition, its content is then broadcast to the registered codelets in the broadcast manager. We have three main differences between standard BFA and CAV, related to this module. The first one is that we don't use information codelets. The second is that not all of the codelets are notified like in BFA, just the registered ones. Finally, some codelets can be active outside of the playing field. In this case their contents will never reach consciousness.

3.4 Behavior Network

CAV's behavior network is based on a version of Maes' architecture [13] modified by Negatu [6]. Negatu adapted Maes' behavior network so each behavior is performed by a collection of codelets. Negatu's implementation also divided the behavior network in *streams* of behavior nodes.

The behavior network works like a long-term procedural memory, a decision structure and a planning mechanism. It coordinates the behavior actions through an "unconscious" decision-making process. Even so it relies on conscious broadcasts to keep up-to-date about the current situation. This is called "consciously mediated action selection" [6].

CAV uses two main behavioral streams, the *Target* stream and the *Energy* stream, as in figures 3 and 4.



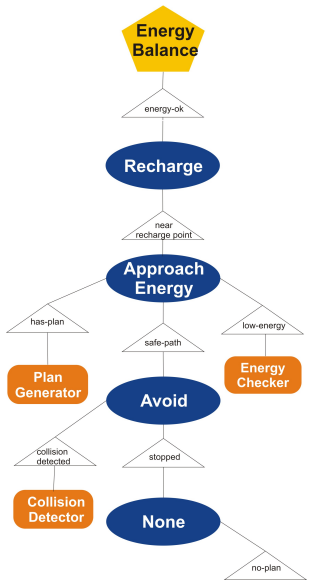


Fig. 3. Behavior Network - Target Stream **Fig. 4.** Behavior Network - Energy Stream

4 A Brief Analysis of CAV's implementation

A running simulation of CAV's performance is illustrated in figure 5. The main experiment worked as expected. The creature was able to pursue its main objectives: to avoid collision with obstacles while exploring the environment, and at the same time maintaining an energy balance. While exploring the environment, if the energy level decreased to a critic limit, CAV correctly postponed its exploratory behavior, looked for the closest source of energy and traced a route to it to feed itself. After refreshing its batteries, it returned to its exploratory behavior. As we said before, though, our main goal was not simply related to the achievement of these tasks (something which could be achieved by more traditional methods, as e.g. in [15]), but understanding how "consciousness" could be used in such an application.

By applying BFA to this application, we would like to evaluate the value of "consciousness" (as in BFA) to the construction of a new generation of cognitive architectures to control artificial creatures. Pragmatically, we would like to understand what exactly it is this "consciousness" technology, and what the benefits to expect while applying it as a mind to an artificial creature. This goal was also achieved while we had the experience of studying BFA and applying it to the current application. Our findings are summarized in the next subsections.

4.1 A Qualitative Analysis

Our implementation of BFA as a mind of our artificial creature allowed us to better understand what is the role of consciousness in BFA and what are its main benefits as a technology. First let us make it very clear what is consciousness (in the context of BFA). The philosopher Daniel Dennet has already stated that:

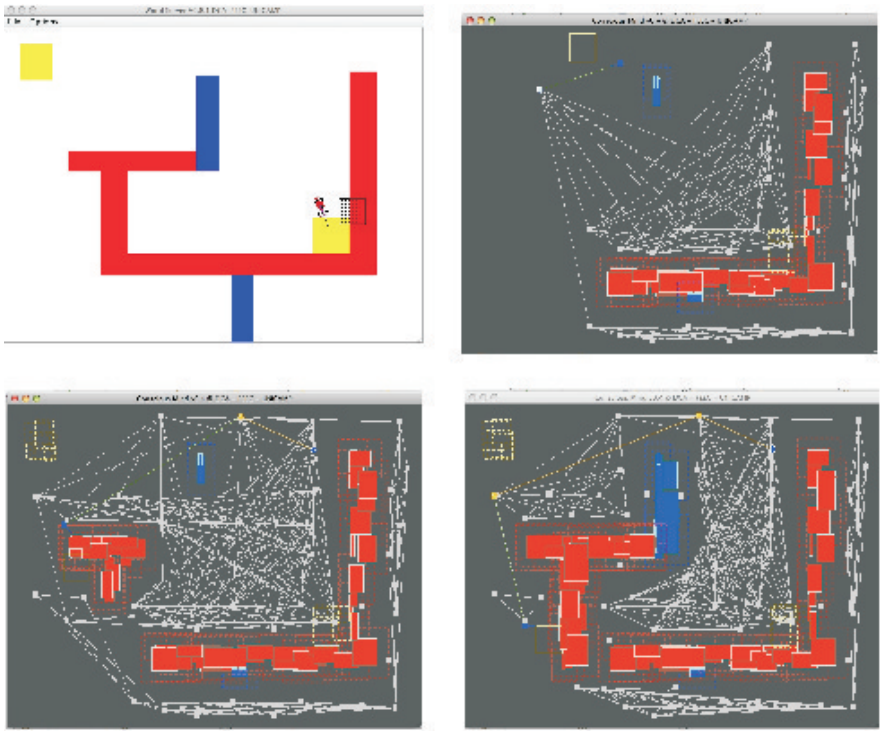


Fig. 5. Example of Simulation

"Human consciousness (...) can best be understood as the operation of a '*Von Neumannesque*' virtual machine *implemented* in the parallel architecture of a brain". This is what BFA provides. It implements a (virtual) serial machine on top of a parallel machine. The overall structure of codelets reading and writing on the *Working Memory* configures a fully parallel multi-agent system. The constraints of the *SpotlightController* and the broadcast mechanism implements on top of it the emergence of a serial stream which is the consciousness. But this serial stream is not just any serial stream. It focuses attention on the most important kind of information in each time step. It builds what Koch called an *executive summary* of information [16]. This is one of the main advantages of this technology: to focus attention on what is most important and spreading this to all agents in the multi-agent system. This simple understanding opens a large set of opportunities to future research. Now we are able to improve this main idea and check other uses for such a technology.

4.2 A Quantitative Analysis

Some data related to the experiment can be viewed in figures 6, 7 and 8.

Figure 6 shows the number of active threads at each instant of time. We can see that an average of 8 threads are working at the same time. Figure 7 shows the number of codelets running at the same time at the playing field. An average of 1 or 2 codelets were at the playing field at the same time. The maximum of

codelets at the playing field at the same time was 3. Finally, figure 8 shows the different types of codelets accessing the consciousness at each time. We can see that most of the time the codelet *ObstacleRecorder* was at consciousness. The second more frequent was *PlanGenerator*. The other three, *TargetCarrier*, *CollisionDetector* and *PathChecker* were less frequently at the consciousness.

These data refer to 1 minute of simulation. The subsequent instants of time show more or less the same behavior. Other codelets, like e.g. *LowEnergy*, also appear from time to time, but they didn't appear in the time-frame shown in the figure.

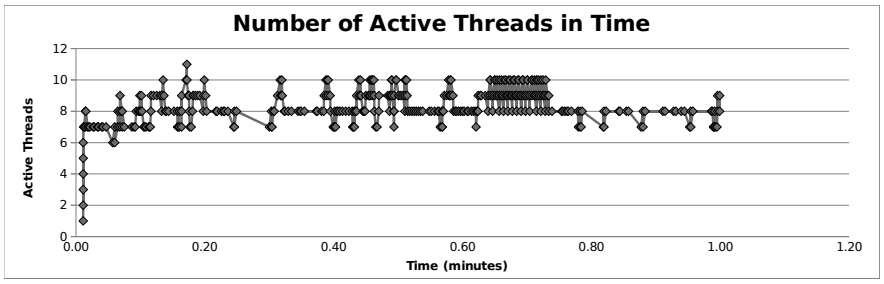


Fig. 6. Number of Active Threads in Time

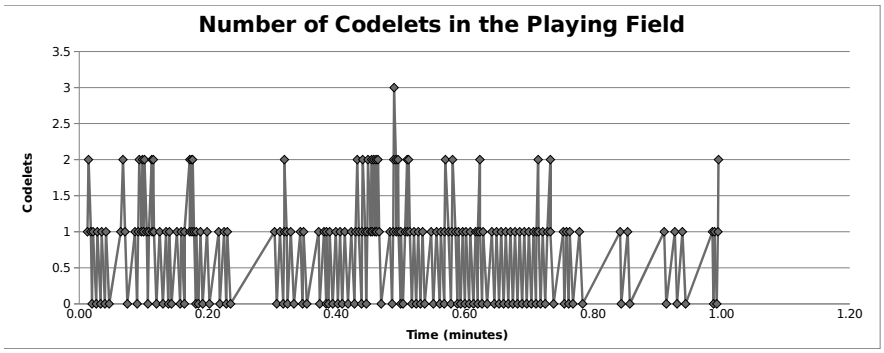
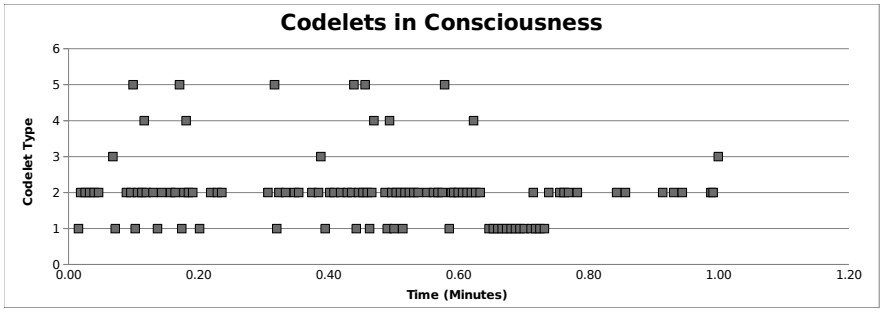


Fig. 7. Number of Codelets in the Playing Field



1 - PlanGenerator 2 - ObstacleRecorder 3 - TargetCarrier
4 - CollisionDetector 5 - PathChecker

Fig. 8. Types of Codelets in Consciousness

5 Conclusion

BFA is shown to be a very flexible and scalable architecture, due to its consciousness and behavior network mechanisms implemented through independent codelets. Newer features can be easily included by means of newer codelets performing new roles. Consciousness mechanism makes possible a deliberation process that enables the perception of most relevant information for the current situation, building what Koch called an executive summary of perception. Much work remains to be done, especially related to a better model formalization and a better understanding of the overall role of coalitions. However, seen as an embryo of a conscious artificial creature, the first results of this study show the feasibility of such techniques, motivating our group to continue on this line of investigation.

References

1. Blackmore, S.: Consciousness - A very short introduction. Oxford University Press (2005)
2. Bogner, M.B.: Realizing "Consciousness" in Software Agents. PhD thesis, The University of Memphis (December 1999)
3. Chella, A., Manzotti, R.: Artificial Consciousness. Imprint Academic (2007)
4. Gamez, D.: Progress in machine consciousness. Consciousness and Cognition **17** (2008) 887–910
5. Franklin, S., Graesser, A.: A software agent model of consciousness. Consciousness and Cognition **8** (September 1999) 285–301
6. Negatu, A.S.: Cognitively Inspired Decision Making for Software Agents: Integrated Mechanisms for Action Selection, Expectation, Automatization and Non-Routine Problem Solving. PhD thesis, The University of Memphis (August 2006)
7. Baars, B.J.: A cognitive theory of consciousness. Cambridge University Press (1988)
8. Dubois, D.: Constructing an Agent Equipped with an Artificial Consciousness: Application to an Intelligent Tutoring System. PhD thesis, Université du Québec à Montréal (August 2007)
9. Balkenius, C.: Natural Intelligence in Artificial Creatures. Lund Univ. Cognitive Studies 37 (1995)
10. Selfridge, O.G.: Pandemonium: a paradigm for learning. In: Mechanism of Thought Processes: Proceedings of a Symposium Held at the National Physical Laboratory, London: HMSO (November 1958) 513–526
11. Jackson, J.V.: Idea for a mind. ACM SIGART Bulletin **xx**(101) (July 1987) 23–26
12. Hofstadter, D.R., 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 **2** (1994) 31–112
13. Maes, P.: How to do the right thing. Connection Science Journal **1** (1989) 3
14. da Silva, R.C.M.: Análise da arquitetura baars-franklin de consciência artificial aplicada a uma criatura virtual. Master's thesis, DCA-FEEC-UNICAMP (July 2009)
15. Gudwin, R.R.: Contribuições ao Estudo Matemático de Sistemas Inteligentes. PhD thesis, DCA-FEEC-UNICAMP (1996)
16. Koch, C.: The Quest for Consciousness - A Neurobiological Approach. Roberts & Company Publishers (2004)