A model for embodied cognition in autonomous agents

CONFERENCE PAPER · SEPTEMBER 2012

DOI: 10.1007/978-3-642-33197-8_60

3 AUTHORS:

Marco Vala
Inesc-ID
37 PUBLICATIONS **393** CITATIONS
SEE PROFILE

Tiago Ribeiro
Inesc-ID
12 PUBLICATIONS **30** CITATIONS
SEE PROFILE

Ana Paiva
Instituto Superior Técnico, Inesc-ID
308 PUBLICATIONS **2,977** CITATIONS
SEE PROFILE

Censys: A Model for Distributed Embodied Cognition

Tiago Ribeiro, Marco Vala, and Ana Paiva

INESC-ID & Instituto Superior Técnico, Technical University of Lisbon
`{tiago.ribeiro,marco.vala,ana.paiva}@inesc-id.pt`

Abstract. The role of the body in the generation of behavior is a topic that has sparked the attention of many fields from philosophy to science and more recently robotics. We address the question of how an embodied agent should be modeled in order to change the traditional dualist approach of creating embodied agents. By looking at behavior generation as a shared process between mind and body, we are able to create modules that generate and manage behavior, which are neither part of the body nor the mind, thus allowing a more flexible and natural control. A case study is presented to demonstrate and discuss our model.

Keywords: Embodied Agents; Embodied Cognition; Artificial Intelligence; Robotics

1 Introduction

Embodiment as something of or related to the human body is an important field of research. Our physical bodies define how we stand in space and time, and our awareness is deeply influenced by the fact that we have a body. Over the years, philosophers, psychologists, cognitive scientists, and more recently computer scientists have looked at embodiment from different perspectives.

In Computer Science, there is the concept of embodied agent, which is a software agent that interacts with the surrounding environment through a body. Embodied agents can have actual physical bodies, like Robots [1], or they can have a graphical representation of their body, like Embodied Conversational Agents [2][3]. In both cases, a form of embodiment (physical or virtual) is a necessary condition to interact with the environment and with human beings [4].

One particular topic related with embodiment is the relationship between mind and body. The mind-body problem was famously addressed by Descartes in the 17th century when he proposed his dualist perspective. Cartesian Dualism assumes that the mental phenomena are essentially non-physical, and that mind and body are two separate things.

The traditional computational models to create embodied agents follow a dualist perspective. There is a clear separation between the "mind" of the agent and the "body" of the agent. The body is an interface to the environment through a set of sensors and effectors. The mind receives sensory information from the body, analyzes that information and activates the effectors. There is a continuous sense-reason-act loop in which the mind has full control over the body.

However, such approach has some implications. The mind, as a centralized decision-making system, has to cope with different levels of control at the same time, ranging from lower-level control of sensors and effectors to higher-level cognitive tasks that involve reasoning and deciding what the virtual agent or the robot should do next. Moreover, the level of abstraction provided by both sensors and effectors, and their ability to map sensory input into symbolic representations or turn symbolic representations into effector output, has a direct impact in what the agent can do. As a consequence, the mind usually ends up tightly coupled to a particular form of embodiment.

Human beings, on the other hand, have intermediate layers of control at different levels. Our bodies have regulation mechanisms that perform subconscious tasks in parallel with our higher-level cognitive tasks. Damásio [5] presents recent findings in neuroscience of how our bodies are important in shaping the conscious mind, and their role in key processes like the emotional phenomena. Pfeifer [6] also points out that, despite the breadth of the concept, whatever we view as intelligent is always compliant with the physical and social rules of the environment, and exploits these rules to create diverse behavior. Since our bodies define how we interact with the environment, we cannot dissociate intelligence from our body as a whole.

Therefore, we need a computational model that looks at mind and body as a continuum. In the Society of Mind [7], Minsky looks at the mind as a collection of cognitive processes each specialized to perform some type of function. A cognitive process is represented by a component and the internal composition of these components creates a network of complex behavior.

This paper looks at embodiment following the same approach. We look at the body as a sum of components that perform specialized functions. Mind and body can share the same space, because we don't look at them as separate processes. Instead, both the notions of mind and body emerge from the components that support them.

In the next section we will look at related work. Then we present our model and a case study that discusses our approach. Finally, we draw some conclusions and outline future work.

2 Related Work

Researchers working in the field of virtual and robotic agents have been exploring richer models for behavior generation in autonomous agents. Recently, there have been developments towards new frameworks and tools to create agents capable of generating and exhibiting complex multimodal behavior. A popular framework that defines a pipeline for abstract behavior generation is the SAIBA framework [8], illustrated in Figure 1.

Our architecture follows on [9], which uses a pipeline of components that are reusable and migrate across different forms of embodiment. These components

Fig. 1. The SAIBA framework [8].

were used to create mixed scenarios where agents can migrate between virtual and robotic bodies.

Moreover, in order to deal with the other side of the loop, Scherer et al. propose a Perception Markup Language (PML) that should work just like BML [10]. As agents perceive the external world through their bodies, it enables embodiments to create an abstraction of perceptual data in order to bring it up to the cognitive level.

The interaction between perceptual data and BML has also been explored. We have integrated perceptions into BML and had two different robots interact with each other while running on the same system [11]. The interactive behavior is abstract enough to drive two completely different embodiments.

Other approaches to build embodied agents use physiological models to create behavior in autonomous agents. For example, Cañamero uses a multi-agent approach where each physiological function is modeled using an agent [12]. The work models hormones which are the foundation of motivational and emotional processes that guide behavior selection in Robots.

From a point of view of engineering, the component-based approach is very common in Robotics. ROS - Robot Operating System is a popular middleware developed by Willow Garage [13] that provides a common communication layer to enable different kinds of sensors, motors and other components to send data between each other. ROS is module-based, meaning that a ROS-based robot actually runs several different modules, being each one of them responsible for controlling one or several components of the robot. The main advantage of this is that all these modules can be shared and reused throughout the community.

3 The Censys Model

The model we propose follows on the concepts we previously introduced in the first section, and is inspired in the component decomposition proposed by Minsky.

Censys is modeled as a distributed network of Modules, which can have any non-zero number of connectors. A Module is conceptualized as being a *black-box* which can have or not an internal state, and can react to data received on its sensors, through the use of its effectors. As an abstraction, a Module can actually be seen as a sub-agent that composes both the mind and the mind-body interface.

The Censys model does not enforce any specific typology for the network. It can therefore be built just like a traditional agent, as show in Figure 2. Sensors are illustrated as triangles pointed at the module, and actuators as triangles pointed away from the module.

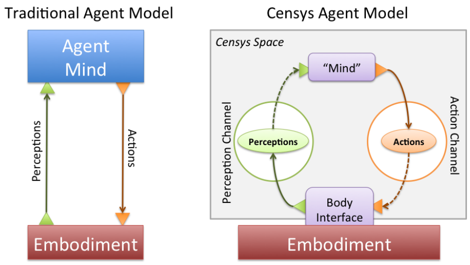

Fig. 2. How the Censys model fits into the Traditional model.

What is generally viewed as the Agent Mind, providing deliberative, reactive or dialogue behavior, is now just a module that fits into the architecture. The Body is decomposed into a one or more Censys modules that serve as interfaces to the Embodiment.

A Module can use four types of connectors to sense and act:
PerceptionSensor subscribes to and receives perceptions P_T of a type T;
PerceptionEffector generates perceptions P_T of a type T;
ActionSensor subscribes to and receives actions A_T of a type T;
ActionEffector generates actions A_T of a type T;

The ActionEffectors and PerceptionEffectors generate actions and perceptions and place them in the perception and action channel respectively. The channels have the information of which ActionSensors and PerceptionSensors are subscribed to which type of actions and perceptions, so it can then transmit those to all of the subscribed sensors.

An example of how these modules can be composed into a Censys agent with a Body, Decision-Making (DM) module, and three other Modules is shown in Figure 3. This agent is merely an out of context example and as such its Modules do not have any specific meaning.

The three new modules in the example agent are able to receive some kind of perceptions and actions, process them, and generate other kinds of perceptions and actions.

Taking as example Module A, it subscribes to actions of type A_3 and converts them into actions of type A_1. The purpose of this module is thus to act as a high-to-low-level converter, decomposing high-level actions from the DM into lower-level actions that are more appropriate for the Body to manage and use without having to know how to interpret the high-level information produced by the DM.

Another example, Module C, has a dual purpose. One purpose is to act as a perception converter, by receiving lower level P_1 perceptions and turning them into higher level P_3 symbolic perceptions that are more appropriate for the DM. At the same time, it can also generate low-level reactions by generating A_2

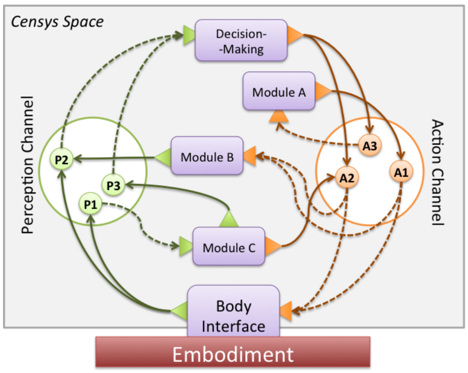

Fig. 3. An example (no context) Censys agent.

actions depending on the Module's internal state and the data collected from the perceived P_1s.

The main advantage of using this architecture as that the DM does not explicitly need to know how to communicate with the Body and vice-versa.

In our example agent, we can see that the DM can only receive P_2 and P_3 perceptions and generate A_2 and A_3 actions. The body, however, can only generate P_1 and P_2 perceptions, and receive A_1 and A_2 actions, so the DM and Body used can natively only communicate through P_2 perceptions and A_2 actions. This means that if we removed the three Modules (thus having a traditional agent instead of a Censys agent), we would need to adapt both the DM and Body to be able to deal with all types of perceptions and actions.

In a Censys agent we just use Modules that can handle these types of perceptions and actions, and still use the DM and Body as they are. A direct advantage of this is that it makes it much easier to swap the traditional *Agent Mind* or the *Embodiment* with other ones that may not have support for all the used actions and perceptions, and still function together.

4 Case Study - A Component-based Robot

In figure 4 we present a more complex case study in which a Censys agent is used to control a robot, based on the SAIBA framework. The Intention and Behavior Planning is done within the Decision-Making and Dialogue Manager modules (DMs), which generate only A_{BML} actions containing Behavior Markup Language (BML) blocks[8]. The DMs also receives only P_{PML} perceptions containing a Perception Markup Language (PML) blocks, which is a high-level representation of perceptual data [10].

This Embodiment is actually connected to the agent through three modules: the Speech, Audio and Body Interfaces. This case study thus also serves as an example of how a body is not necessarily one entity, but a coupling of several

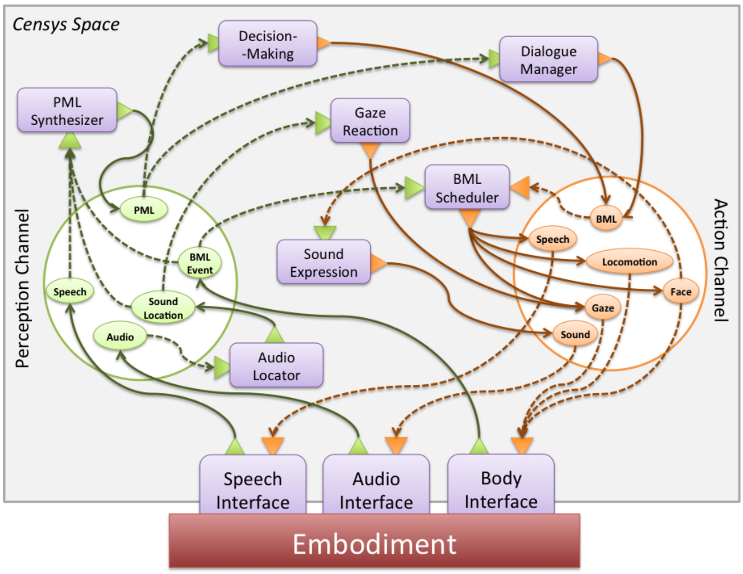

Fig. 4. Diagram of the Censys agent of the Cognitive/Reactive Robot case study.

entities. Moreover, if we wanted to use a different Robot we could just switch its Body Interface and still keep all our behavior related to speech and audio.

The **Body Interface** receives A_{Face}, A_{Gaze} and $A_{Locomotion}$ actions, and generates $P_{BMLEvent}$ perceptions. These perceptions contain feedback about the executing actions (when they physically started/finished/failed, etc.).

The **Audio Interface** receives A_{Sound} actions containing an audio signal to be output, and generates P_{Audio} containing audio signals whenever a sound is captured from the environment.

The **Speech Interface** acts as an interface to both a Text-to-Speech (TTS) engine, and to a Speech Recognition engine. In this case we assume that the Speech Recognition is already being fed with an audio signal, as such it produces P_{Speech} perceptions containing detected speech. This interface also receives A_{Speech} actions containing information that the TTS engine uses to generate speech.

We will now analyze each of the different modules that compose the behavior of this agent:

BML Scheduler deals with decomposing high-level behaviors, and scheduling and running the separate actions that compose such behavior. It subscribes to A_{BML} actions that correspond to the high-level behavior (a BML block) and generates actions of type A_{Face}, A_{Gaze}, $A_{Locomotion}$ and A_{Speech};

Sound Expression provides expressive redundancy to some of the agent's expressive behavior. In this case, every time a module produces an A_{Face} action, this module will generate an A_{Sound} action that contains an audio signal corresponding to the A_{Face} action;

Audio Locator serves as a converter module that takes as input P_{Audio} perceptions containing audio signals, and processes them in order to infer localization information. If the sound can be localized it generates a $P_{SoundLocation}$ perception with that information;
Gaze Reaction performs gaze reactions to sound events. Whenever it receives a $P_{SoundLocation}$ perception that is loud enough, it will generate an A_{Gaze} action that tells the body to gaze towards the direction of the sound;
PML Synthesizer acts as a low-to-high level converter, by receiving individual perceptions like $P_{BMLEvent}$, $P_{SoundLocation}$ or P_{Speech} and transforming them into high-level PML blocks.

4.1 Execution Example

Given the description of the case study, we now provide a description of how that scenario could actually run.

Let's start by assuming that we are using a NAO robot[1] for the embodiment, and that the DMs are very simple and currently only pursue the goal of travelling to a certain physical location that is located a couple of meters in front of the robot's current location. The Decision-Making module processes that goal and defines a simple plan consisting merely of walking forward.

That plan is transformed into a BML block containing a locomotion action and let's say, a speech action for *"I will be there in a minute!"*. The BML Scheduler receives this action and its scheduling of the two behaviors generates first an A_{Speech} action, and then an $A_{Locomotion}$ action. The A_{Speech} action is received by the Speech Interface, which unpacks it and sends it to the TTS, making the robot say *"I will be there in a minute!"*. The $A_{Locomotion}$ action is received by the Body Interface, which triggers the robot to start walking.

In the mean while, someone goes by NAO and speaks to him. The Audio Interface detects this sound and generates a P_{Audio} perception, which is sensed by the Audio Locator module which in turns calculates the offset angle at which the sound was detected. It then generates a $P_{SoundLocation}$ perception. This perception is received both by the PML Synthesizer which generates a P_{PML} perception based on its data, and also by the Gaze Reaction module. This one generates an A_{Gaze} action that tells the body to gaze at the angle of the detected sound. That A_{Gaze} action is then also received by the Body Interface, thus making NAO look at the direction of the person who spoke, while continuing to walk. The DMs also receive the P_{PML} perception containing the information about a sound perceived at certain angle, but as long as it does not interfere with its current state and goals, this perception does not trigger anything at this level.

That does not imply that the robot is unable to react to anything else. When an intense sound is located, gazing at it can have several benefits. We think of a functional one - being able to use vision recognition to analyze what has generated the sound; and an expressive one - if it was a person who made

---

[1] `http://www.aldebaran-robotics.com`

the sound, then having the robot gaze towards that direction helps to transmit a more sentient impression of the robot.

Besides these two benefits, this behavior is not encoded neither in the DMs nor in the embodiment, so that means it could be reused even if we switch to another robot.

4.2 Discussion

While the diagram presented in Figure 4 may seem complex, it actually portrays a very simple scenario. What do we gain from having such cross-connections and interrelationships?

We find this kind of model to be especially appropriate for modeling high-level autonomous behavior in robots, because we can create some behaviors that are body-independent (like gaze-reacting to a sound) and use them with different embodiments. Of course, another question may arise that is HOW does the robot implement that gaze. That is a problem that we consider to be body-dependent, as a robot that has a head may be able to gaze while walking, whereas a robot built on a two-wheeled self-balancing base (like the Segway technology [2]) will have to stop moving in order to turn and face another direction.

Another vantage point we find is that each module offers its own independent control. An example of this is the Sound Expression module. The Dialogue Manager does not even need to be aware of this module while it supports the expression of the robot. If we know that our robot has limited expression capabilities, we can just include this module without having to change anything else.

The BML Scheduler is also a complex control module as it manages the scheduling and composition of behaviors that the DMs decided to execute. However, the DMs do not need to know what the Body is currently doing - resource management is distributed along the Mind-Body space.

There is still another situation related to resource management, that is when for example both the BML Scheduler and the Gaze Reaction modules produce an A_{Gaze} action. In this case, and having no other module to play that role, the resource management is expected to be done at the Embodiment layer, so we cannot predict or define in our model what will be the resulting action.

Summing up this case study, by using a Censys agent we steer towards the ability to reuse behaviors that manage proper and natural interaction, even when working with different embodiments (virtual or robotic). As some of the agent's behavior does not need to be re-programmed, the development of new behaviors and contexts for robot usage should therefore become more accessible.

5 Conclusion

There are several aspects that we intend to approach with our model. First of all, we were deeply inspired by the fact that most traditional architectures

---

[2] `http://www.segway.com`

overlook the role of the body in the cognition process. This leads to placing a high computational load on the agent's mind and expecting it to be able to cope with both low and high-level processes.

That is the issue that generally leads to dependence between body and mind using the traditional approach. By breaking that hard link between body and mind, we are able to transfer some common behaviors into a more abstract functional space between the cognitive mind and the body, thus making it feasible to share behaviors and part of the cognition process even if we use different bodies.

One important thing on embodiment switching is when we switch between virtual and robotic bodies. Virtual embodiments are considered to be perfect and immediate, meaning that they have direct access to real information about their body and the world around. Robots, however, have imperfect sensors and effectors, meaning that there can be noise in the information and also measurement errors or other deviations caused by gravity, inertia, friction, etc. By including a space where we can create filters for sensors and actuators, we can relieve the higher-level processes from handling all those issues.

In the same sense, there are some processes that may require a continuous feedback loop between a specific controller and the embodiment. Again, such a controller can deal with feedback and adjust the behavior in real-time without having to interrupt the higher-level processes.

The bottom line is that Censys creates a space where we can define internal processes that run in parallel with the main control loop of the agent. This distributed control is the base to create agents that are able to display a more natural behavior, for example by enriching their primary behavior with involuntary movements or by reacting faster when something happens.

Therefore, embodied agents are able to rely on their bodies as much as they rely on the artificial minds that reason and decide for them. Mind and body work in parallel to generate behavior and continuously adapt to each other. Like humans do.

5.1 Future Work

The next step will be to implement some concrete scenarios using Censys. We want to compare the development of a Censys agent against the traditional approach. This comparison should yield results both about how easy and fast it is to develop a Censys agent, but also to what degree the Censys modules are actually interchangeable within different embodiments. It should also guide us towards defining the requirements on both the model and its modules in order for that interchangeability to remain valid.

Censys also enables the creation of an internal body model where we can explore concepts like physiological space, interoception, and proprioception. Usually these mechanisms are part of subconscious processes which are not that important for the actual behavior displayed by our agents. However, they might be important in the decision process that lead to those behaviors in the first place and we will be able to experiment with that.

There is also an open issue related to resource management (RM). Having distributed control implies having some concurrency and how to deal with shared resources, like the effectors. In this paper we have delegated that process to the actual embodiment. However, we have also hinted in our case study how the BML Scheduler also performs some RM. That make us to wonder if it may be possible to actually include better RM mechanisms in our model, and to what level can they perform to be shared amongst different types of embodiment.

Acknowledgments. This work was supported by national funds through FCT Fundação para a Ciência e a Tecnologia, under project PEst-OE/EEI/LA0021/2013 and by the EU 7th Framework Program project EMOTE (FP7/2007-2013) under grant agreements no. 215554 and 317923.

References

1. Niewiadomski, Radosaw, Obaid, M., Bevacqua, E., Looser, J., Le, Q. A., and Pelachaud, C. (2011) Cross-media agent platform. **1**, 11–20.
2. Cassell, J., Bickmore, T., Billinghurst, M., Campbell, L., Chang, K., Vilhjálmsson, H., and Yan, H. (1999) Embodiment in conversational interfaces: Rea. *SIGCHI '99*, New York, NY, USA, pp. 520–527, CHI '99, ACM.
3. Poggi, I., Pelachaud, C., De Rosis, F., Carofiglio, V., and De Carolis, B. (2005) *GRETA. A Believable Embodied Conversational Agent*, vol. 27, pp. 1–23. Kluwer Academic Publishers.
4. Dautenhahn, K. (1999) Computation for metaphors, analogy, and agents. chap. Embodiment and Interaction in Socially Intelligent Life-like Agents, pp. 102–141, Springer-Verlag.
5. Damasio, A. (2011) *Self Comes to Mind: Constructing the Conscious Brain*. Random House.
6. Pfeifer, R., Bongard, J., and Grand, S. (2007) *How the Body Shapes the Way We Think: A New View of Intelligence*. Bradford Books, Mit Press.
7. Minsky, M. (1988) *Society Of Mind*. A Touchstone Book, Simon & Schuster.
8. Kopp, S., Krenn, B., Marsella, S., Marshall, A. N., Pelachaud, C., Pirker, H., Thórisson, K. R., and Vilhjalmsson, H. (2006) Towards a common framework for multimodal generation: The behavior markup language. *Intelligent Virtual Agents (IVA)*.
9. Kriegel, M., Aylett, R., Cuba, P., Vala, M., and Paiva, A. (2011) Robots meet ivas: a mind-body interface for migrating artificial intelligent agents. *Intelligent Virtual Agents (IVA)*, Berlin, Heidelberg, pp. 282–295, IVA'11, Springer-Verlag.
10. Scherer, S., Marsella, S., Stratou, G., Xu, Y., Morbini, F., Egan, A., Rizzo, A. S., and Morency, L.-p. (2012) Perception Markup Language : Towards a Standardized Representation of Perceived Nonverbal Behaviors. pp. 455–463.
11. Ribeiro, T., Vala, M., and Paiva, A. (2012) Thalamus: Closing the mind-body loop in interactive embodied characters. *Intelligent Virtual Agents IVA'2012*, Eindhoven, The Netherlands, vol. 7502 of *Lecture Notes in Computer Science*, pp. 189–195, Springer Berlin / Heidelberg.
12. Canamero, D. (1997) A hormonal model of emotions for behavior control. *VUB AILab Memo*, **2006**.
13. Quigley, M. and Gerkey, B. (2009) ROS: an open-source Robot Operating System.