The Octopus as a Model for Artificial Intelligence
A Multi-Agent Robotic Case Study

Alfonso Íñiguez
Swarm Technology, LLC Mesa, Mesa, AZ, U.S.A.

Keywords: Autonomous, Robotics, Artificial Intelligence, Multi-Agent, Swarm Intelligence, Parallel Processing.

Abstract: The aim of this paper is to investigate the curious cognition process exhibited by the octopus, and its practical applicability to multi-agent systems. The paper begins by explaining the limitations of using the human brain as a model to achieve artificial cognition and proposes an alternative model inspired by the octopus' distributed approach to solving problems. As a case study, a laboratory prototype demonstrates awareness, autonomy, solidarity, expandability, and resiliency in a multi-robotic system. The cognition model described in this paper is primarily algorithmic and does not explore the model creation process nor semantics; rather, it lays the foundation and inspiration for a future realization as a Process for Agent Societies Specification and Implementation (PASSI).

1 INTRODUCTION

The conventional approach in the attempt to achieve artificial intelligence is to use the human brain as a model of operation; in fact, there are operational similarities between the computer processing unit (CPU) and the human brain – both make decisions by fetching and processing data from memory, and both store the processed data in memory.

The downside of using the human brain as a model for artificial intelligence is *scaling*; as the complexity of the tasks increase, the performance demand on the CPU proportionally increases.

Limitations remain even with the advent of coprocessors — intended to offload tasks from the CPU. The traditional multiprocessing framework (see Figure 1) suffers from two major drawbacks, both caused by the architectural requirement that the CPU must divide and distribute the threads.

First, a significant amount of the CPU's processing time is consumed in managing the coprocessing tasks. The management may include: distributing tasks to the coprocessors according to their capabilities, waiting for those tasks to be completed before reassigning new tasks; and responding to interrupts from coprocessors every time a task is completed.

Second, a coprocessor will remain idle as it awaits for a thread to be assigned to it by the CPU. A multiprocessor system that alleviates the management workload on the CPU while keeping the co-processors busy is needed.

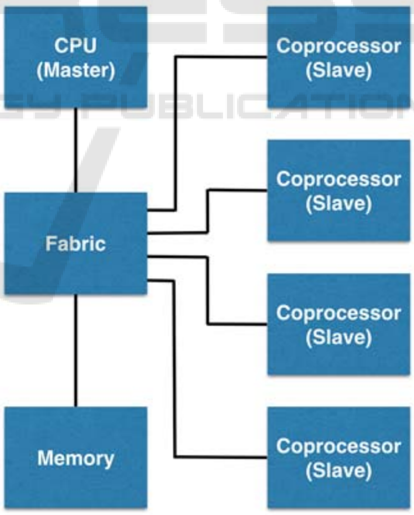

Figure 1: Traditional multiprocessing framework.

Given that robotic movement is ultimately enabled by its processing capability, the same two drawbacks that affect CPU/Coprocessor performance limit robotic autonomy. Hence, to enable robotic autonomy, it makes sense to begin by solving the two computer processing drawbacks.

ÃĂśiguez A.
The Octopus as a Model for Artificial Intelligence - A Multi-Agent Robotic Case Study.
DOI: 10.5220/0006125404390444
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 439-444
ISBN: 978-989-758-220-2

2 OCTOPUS COGNITION

Written records of octopuses leaving the water have existed for over 2,000 years (Balme, 1991). The *Octopus Alpheus* is known to leave the water to crawl between tide pools (Norman, 2000). More recently, Boyle (1991) wrote, “Octopuses are particularly prone to escape from aquarium tanks. Loose lids are of little value because the octopuses will easily lift them and push their way out of the tank” (Boyle 32).

2.1 Cognition Evidence

It is not surprising to learn that the octopus is considered to be the most intelligent of all invertebrates (Linden, 2002); it learns simple mazes (Boal, 1996), uses landmark navigation while foraging (Mather, 1991), and uses tools (Mather, 1994).

2.1.1 Cognition Efficiency

Experimentation results do not imply that octopuses are smarter than human children; however, the octopus is a model for efficient cognition given the limited amount of available neurons in its brain — 500 million in the octopus as opposed to almost 100 billion in Homo sapiens.

Biologists at the Seattle Aquarium challenged a female *Enteroctopus dofleini* — a giant Pacific octopus — with a childproof bottle, the kind that can puzzle Homo sapiens. The results were staggering, “To open the lid it was necessary to push down on the lid at the same time as turning it … the octopus, accomplished this task in 55 minutes … Further presentations resulted in a decrease of the average opening time to 5 minutes” (Anderson, 2006).

2.1.2 Distributed Cognition

Distributed neurons allow the octopus’ arms to problem-solve autonomously; the “arms are not entirely under the control of the octopus' brain . . . two thirds of its neurons reside not in its central brain but out in its flexible, stretchable arms” (Harmon, 2013).

The “Octopus’ arms have a mind of their own … as a result, the arms can problem-solve how to open a shellfish while [the octopus] is busy doing something else, like checking out a cave for more edible goodies” (Nuwer, 2013).

2.1.3 Arms React after Detachment

Researchers, working at St. George's University of London and the Anton Dohrn Zoological Station in Naples, Italy, demonstrated that, "the arms are capable of reflex withdrawal to a 'noxious' stimulus without reference to the brain." (Harmon 2013a) Other experiments show an active nervous system after detachment, “the arms can react after they’ve been completely severed. In one experiment, severed arms jerked away in pain when researchers pinched them" (Nuwer, 2013).

2.1.4 Arm Ambidexterity

A series of interactions were performed to determine if the octopus (*Enteroctopus dofleini*) had arm preference when reaching for objects; the results supported the hypothesis of ambidexterity of the arms. All arms are equally willing to work; arm selection is based on availability and relative proximity (Wülker, 1910).

2.2 Summary of Principles of Cognition

After investigating the behavior of the octopus and the embedded cognition of its arms, we can clearly see that the octopus — when viewed as a processing system — is a superb model for efficient cognition.

Let’s now generalize the cognition principles governing the octopus’ system. As a way of keeping the principles as generic as possible, the arms will be referred to as “members” and the octopus will be called “system.”

2.2.1 Principle 1: Member Awareness

Each member must be aware of its surroundings and abilities. This principle is derived from the fact that each arm can react to its environment even when detached from the head.

2.2.2 Principle 2: Member Autonomy

Each member must operate as an autonomous master (not as a slave); this is essential to self-coordinate allocation of labor. This principle is derived from the fact that the arms are not entirely under control of the octopus’ head.

2.2.3 Principle 3: Member Solidarity

Each member must cooperate in solidarity; when a task is completed each member should *autonomously look for a new task* (leveraging its current position). This principle is derived from the observed ambidexterity of the arms.

2.2.4 Principle 4: Member Expandability

The system must permit expansion where members are dynamically aggregated. This principle is derived from the fact that octopuses can regenerate lost arms with ease (Harmon, 2013b).

2.2.5 Principle 5: Member Resiliency

The system must be self-healing; when members are removed, the remaining members should undertake the unfinished tasks. This principle is derived from the fact that losing an arm is not considered traumatic; octopuses occasionally lose an arm in nature and function normally while the limb regenerates (Levy, 2014).

The attributes described above may also be referred to as the Five Principles of Swarm Intelligence (Íñiguez, 2016).

2.3 Octopus' Cognition Model

After defining the principles of the octopus' arm behavior, the next step in abstracting the octopus' cognition model is to define an architectural representation (see Figure 2).

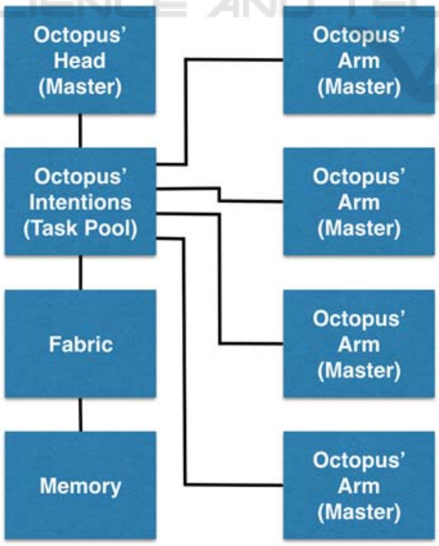

Figure 2: Octopus' Cognition Model.

3 APPLICABILITY OF THE COGNITION MODEL

Notice the differences between the traditional multiprocessing framework of Figure 1 and the octopus' cognition model of Figure 2. There are two main differences:

First, the coprocessors of the traditional model became masters instead of slaves.

Second, the CPU, which is equivalent to the octopus' head, does not directly communicate with the coprocessors; instead, the coprocessors autonomously read the octopus' intentions, i.e. seek tasks from the task pool.

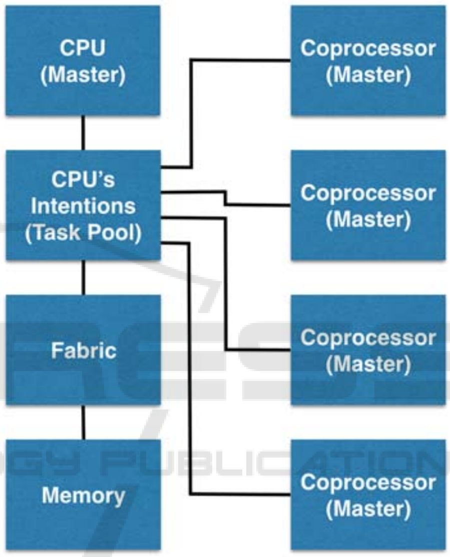

Figure 3: Solidarity Cell Architecture.

3.1 Transposing the Cognition Model

The transposing of Figure 1 into Figure 2 resulted in Figure 3. A fundamental principle of operation of the proposed model is the cooperation in solidarity; since each member is a processing cell in the system, we will refer to this cognition model by the name of Solidarity Cell Architecture (SCA).

The SCA model solves the two limitations of the traditional model previously described in the introduction. In the SCA model the CPU does not spend a significant amount of time micromanaging coprocessors — just as the octopus' head does not spend time micromanaging the arms — and the coprocessors do not remain idle waiting for tasks to be assigned.

3.2 Further Description of the Solidarity Cell Architecture

In general terms, the SCA model is to be described as a method for processing information in parallel; the system uses autonomous computer processing cells to perform tasks needed by a central processing unit. Each cell in the system is connected through a switching fabric, which facilitates connections for data transfer and arbitration between all system resources. A cell has an agent, which is a software module that may be transferred through the switching fabric to a task pool containing the tasks. The agent searches within the task pool for available tasks that match the cell's instruction type. A task may be broken into threads that are to be executed sequentially or independently depending on recipes constructed by the central processing unit. Interdependent tasks within the task pool may be logically combined as needed by the recipe. A notification is sent from the task pool to the central processing unit when a task or task thread is completed.

Therefore, it is an object of this new architecture to provide a method for parallel processing in a multiprocessor system using coprocessors — or autonomous robots — that proactively seek threads to process (Íñiguez, 2013).

3.3 Applicability into Robotics

A recurring challenge in robotics is to build a biped robot that has the balancing ability of humans. A mechanism to account for *continuous balancing* is needed; as the robot walks, climbs, or bends, it needs to swing its arms autonomously to keep its balance.

Figure 4 shows the conceptual balancing mechanism; the shoulders and elbows — *a*, *b*, *c*, and *d* —are equipped with actuators that continuously and autonomously send wireless software agents to seek tasks from the *intention's* task pool – the intention's task pool is a module in which the system's central brain deposits its desire to maintain balance. In this example, the intention's task pool is analogous to a cerebellum in charge of coordinating and maintaining balance. However, as opposed to a traditional cerebellum that sends commands to the body, the biped robot follows the octopus' model, in which the arms autonomously send inquiring agents to the task pool.

The breakthrough advantage of this implementation is that the central brain system can delegate the task of maintaining balance to an

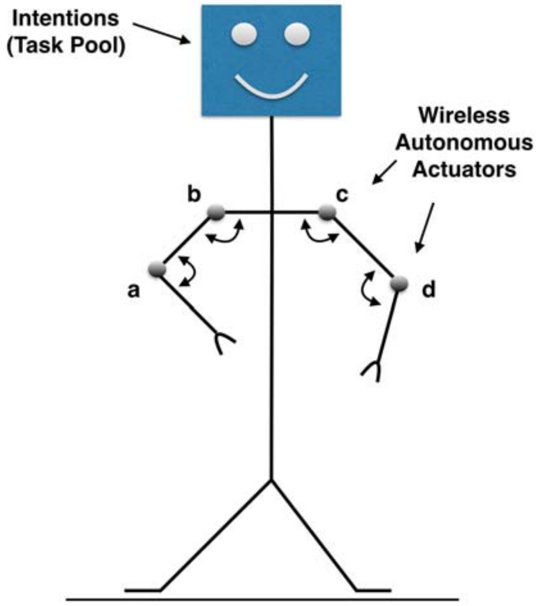

Figure 4: Mechanism to account for continuous balancing.

electronic gyroscope that constantly deposits the balancing requirements into the intention's task pool.

If the biped robot begins to lean to the left, then the gyroscope will deposit an intention in the task pool named "I need somebody to help me move my center of gravity to the right." The shoulders and elbows continuously send agents looking for tasks; when an agent finds a task in the task pool, it returns to its actuator to execute the requirement. The process is repeated continuously achieving human-like balancing without direct intervention from the central brain system.

4 ROBOTIC CASE STUDY

As a way to demonstrate the SCA model with a proof-of-concept prototype, we adapted the biped robot of Figure 4, into the streamlined laboratory representation shown in Figure 5. Free-moving wireless-connected tank robots represent the shoulders and elbows. Hence, the designations of *a*, *b*, *c*, and *d* originally used by the shoulders and elbows in Figure 4 are now given to the tank robots in Figure 5.

The *intention* is implemented by a gyroscope that places "move left" or "move right" tasks into the task pool.

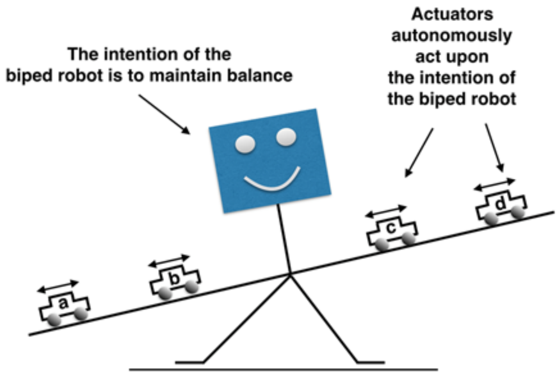

Figure 5: Proof-of-concept laboratory experiment.

The original octopus cognition model can be transposed into the robotic architecture without modification; the only difference between Figure 2 and Figure 6 is the terminology.

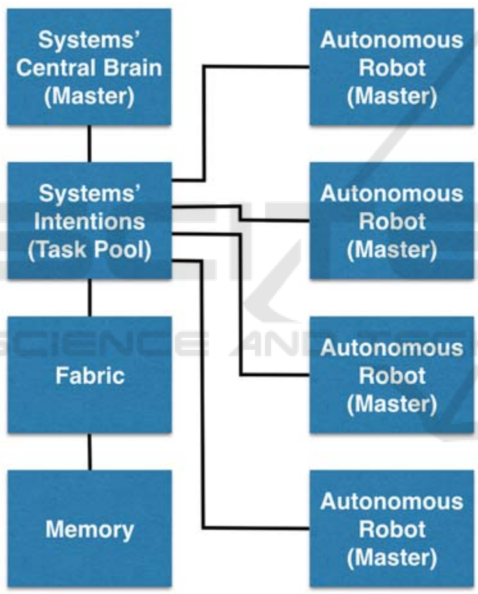

Figure 6: Block representation of the laboratory prototype.

4.1 Prototype Implementation

To build the prototype, we selected to work with off-the-shelf components. The microcontrollers are Arduino™ boards, the robots are a customized version of Makeblock™ toy tanks, and the wireless communication is through Xbee™ (Zigbee) transceivers. We have a YouTube video illustrating the interaction between the tank robot and the gyroscope:
https://www.youtube.com/watch?v=jq1EfxkneJI

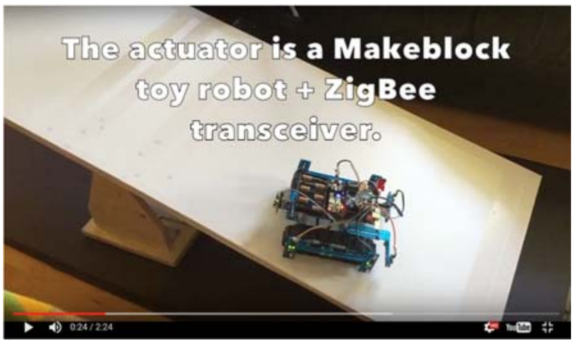

Figure 7: Toy robot interacting with the gyroscope.

4.2 Complete System Implementation

The demonstration of the proof-of-concept prototype exhibiting the five principles of cognition — also known as the five principles of swarm intelligence — is also available via YouTube:
https://www.youtube.com/watch?v=axxXz2BM0yw

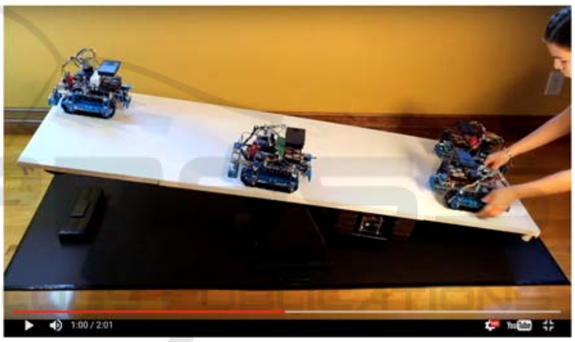

Figure 8: The Five Principles of Swarm Intelligence.

5 CONCLUSION

Various companies and academic institutions are actively researching the field of swarm intelligence; a search on the topic reveals two distinct approaches:

a) Each member is controlled through a central computer, e.g. Intel's 100 drones (Geiver, 2016).
b) Each member behaves autonomously without a central computer; e.g., Harvard University's 1024 Robot Swarm (Hotz, 2014).

Both approaches have merits and limitations (Íñiguez, 2016).

In the case of *a*, members are slaves in a system controlled by a central computer with sufficient channels of communication. The results can be visually spectacular — as illustrated by Intel's drones. However, since a central computer dictates the movement of each member, there is limited flexibility to adapt to changing environments, such as: x) members lost to unforeseen events, y) members added to speed up the mission, or z) members autonomously self-allocating labor.

Of course, the intricacy of the central software may be increased to account for *x*, *y*, and *z*, but that would make the central computer responsible for real time response, it would increase vulnerability due to single point of failure, and it would deviate from the concept of swarm intelligence which is defined as the collective behavior of decentralized, self-organized systems.

In the case of *b*, members have the autonomy to adapt without a central dictator. Considering that each member possesses modest processing power — as illustrated by Harvard University's swarm of robots — the results are truly impressive; nevertheless, this type of behavior falls into the realm of swarm flock. It does meet swarm intelligence's basic definition of collective behavior of decentralized, self-organized systems, but it still lacks the ability to autonomously distribute and undertake allocation of labor.

If neither *a* nor *b* meets the requirements of autonomous allocation of labor, then we need a different approach.

As demonstrated in the proof-of-concept protopype, the Solidarity Cell Architecture effectively achieves the principles of awareness, autonomy, solidarity, expandability, and resiliency; it also solves the two major drawbacks described in the introduction, i.e., CPU micromanagement and coprocessor idleness, present in the traditional multiprocessing framework.

REFERENCES

Balme, D. M. (1991). *Aristotle: History of animals*. Cambridge, MA: Harvard University Press.

Norman, M. (2000). *Cephalopods: A world guide*. Hackenheim, Germany: Conch.

Boyle, P. R. (1991). *The UFAW handbook on the care and management of cephalopods in the laboratory*. Potters Bar, England: UFAW.

Linden, E. (2002). *The octopus and the orangutan*. Dutton (NY). 242 pp.

Boal. J.G. (1996). *A review of simultaneous visual discrimination as a method of training octopuses*. Biol. Rev. 71:157-190.

Mather, J.A. (1991). *Navigation by spatial memory and use of visual landmarks in octopuses*. Journal of Comparative Physiology. 168(491-497).

Mather, J.A. (1994). *"Home" choice and modification by juvenile Octopus vulgaris* (Mollusca: Cephalopoda): specialized intelligence and tool use? J. Zool. Lond. 233:359-368.

Anderson, R.C (2006). *The Festivus* Seattle Aquarium 38(1):7-9 TheCephalopodPage.org.

Nuwer, R 2013. *Ten Curious Facts About Octopuses*. smithsonian.com.

Harmon K. (2013a). *Octopus Chronicles, Even Severed Octopus Arms Have Smart Moves.*

Wülker (1910). *Choice of arm usage in a Giant Pacific Octopus, Enteroctopus dofleini.*

Harmon K. (2013b). *Octopus Chronicles How Octopus Arms Regenerate With Ease.*

Levy G. (2014). *Self-Recognition Mechanism between Skin and Suckers Prevents Octopus Arms from Interfering with Each Other.*

Íñiguez A. I. (2016). *The Five Principles of Swarm Intelligence*, LinkedIn.

Íñiguez A. I. (2013). *Parallel processing with solidarity cells by proactively retrieving from a task pool a matching task for the solidarity cell to process*. U.S. Patent 9,145,777.

Geiver L. (2016). *Intel's Entrance Into UAS, UAS Magazine*

Hotz R. L. (2014). *Harvard Scientists Devise Robot Swarm That Can Work Together.* The Wall Street Journal.