Modulation of multi-level evolutionary strategies for artificial cognition

Oscar J. Romero López
Universidad Politécnica de Madrid
Boadilla del monte, Campus Montegancedo,
28660, Madrid, Spain
(+34) 913366925
ojrlopez@hotmail.com

Angélica de Antonio Jiménez
Universidad Politécnica de Madrid
Boadilla del monte, Campus Montegancedo,
28660, Madrid, Spain
(+34) 913366925
angelica@i.upm.es

ABSTRACT

Integrating different kinds of micro-theories of cognition in intelligent systems when a huge amount of variables are changing continuously, with increasing complexity, is a very exhaustive and complicated task. Our approach proposes a hybrid cognitive architecture that relies on the integration of emergent and cognitivist approaches using evolutionary strategies, in order to combine implicit and explicit knowledge representations necessary to develop cognitive skills. The proposed architecture includes a cognitive level controlled by autopoietic machines and artificial immune systems based on genetic algorithms, giving it a significant degree of plasticity. Furthermore, we propose an attention module which includes an evolutionary programming mechanism in charge of orchestrating the hierarchical relations among specialized behaviors, taking into consideration the global workspace theory for consciousness. Additionally, a co-evolutionary mechanism is proposed to propagate knowledge among cognitive agents on the basis of memetic engineering. As a result, several properties of self-organization and adaptability emerged when the proposed architecture was tested in an animat environment, using a multi-agent platform.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning – *connectionism and neural nets, knowledge acquisition.* I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence – *Intelligent agents, multiagent systems*.

General Terms

Algorithms, Design, Experimentation.

Keywords

Cognitive architectures, gene expression programming , artificial immune systems, neural nets, memetics.

GECCO'09, July 8–12, 2009, Montréal Québec, Canada.
ACM 978-1-60558-325-9/09/07.

1. INTRODUCTION

In the last fifty years, the study of artificial cognitive systems have involved a number of disciplines such as artificial intelligence, cognitive science, psychology and more, in order to determine the necessary, sufficient and optimal conditions and resources for the development of agents exhibiting emergent intelligence.

There are several theories of cognition, each taking a different position on the nature of cognition, what a cognitive system should do, and how a cognitive system should be analyzed and synthesized. From these, it is possible to discern three broad classes: the cognitivist approach based on symbolic information processing representational systems; the emergent systems approach embracing connectionist systems, dynamical systems, and enactive systems, all based on a lesser or greater extent of principles of self-organization [1], [2]; and the hybrid approach which combine the best of the emergent systems and cognitivist systems [3].

Some of the most relevant cognitive architectures which follow a cognitivist approach are: SOAR [4], ACT-R [5], ICARUS [3], and EPIC [3]. Some of the architectures of the emergent approach of major importance are: GW [6], SASE [3], and DARWIN [3]. The hybrid approach architectures are known as CEREBUS [3], KISMET [7], CLARION [8], Polyscheme[9], and LIDA[10]. Some of these architectures deal with aspects of cognitive modeling and representation; some others include learning modules, inference and knowledge generalization; and there are others that try to go further and add motivational and meta-cognition components.

The hybrid approach is more complex and of greater interest to us since it seeks to unify the different dichotomies of symbolic vs. sub-symbolic models, explicit vs. implicit learning, and cognitive vs. emergent approaches. However, a common weakness in the hybrid approach architectures is that they usually abridge the system functionality into a rigid structure of symbolic and sub-symbolic components resulting in a poor ability to self-organize and adapt to new environments.

The present research focuses on implementing a hybrid architecture for cognitive agents supported by both cognitivist and emergent approaches. On the one hand, the cognitivist approach provides an explicit knowledge representation through the use of symbolic AI techniques. On the other hand, the emergent approach defines three evolutionary strategies as observed in nature [11]: *Epigenesis, Ontogenesis*, and *Phylogenesis*, endowing the architecture with implicit knowledge learning, sub-symbolic representations, and emergent behavior guided by bio-inspired computational intelligence techniques. As the cognitivist approach is well known, we will briefly describe it here before elaborating on the emergent approach.

The remainder of this paper is organized as follows. The description of the proposed architecture is detailed in Section 2. Sections 3, 4, and 5 describe in more detail each module of the emergent approach according to the three evolutionary strategies. Section 6 outlines and discusses the results of the experiments. The concluding remarks are shown in Section 7.

2. PROPOSED HYBRID COGNITIVE ARCHITECTURE

Figure 1 gives an overview of the hybrid architecture which has six main modules: Attention module, Procedural module, Intentional/Declarative module, Motor module, Motivational module, and Co-evolutionary module. Each module is composed of sub-modules with more specific functionalities which are communicated to each other by broadcasting mechanisms.

Architecture is distributed in two dimensions: horizontal and vertical dimensions. At horizontal dimension, modules belong to either emergent or cognitivist level, whereas modules at vertical dimension are distributed according to their functionality (attention, procedural reasoning, intentions, motor processing etc.).

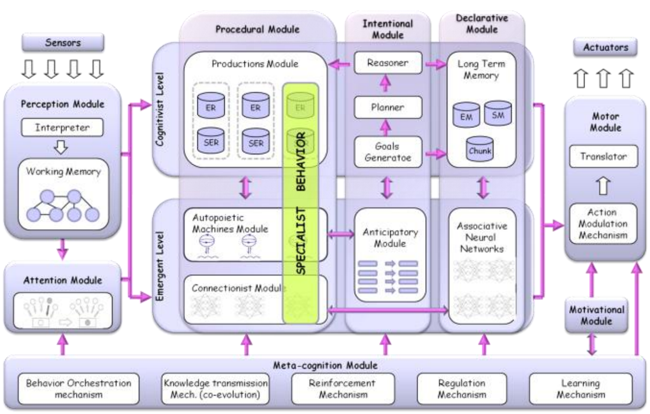


Figure 1. Hybrid Cognitive Architecture.

We will first give a brief description of all the modules of the architecture and then provide a more detailed description of those modules that have been developed so far. Initially, our work has focused on developing the procedural and co-evolutionary modules and their interaction with attention and motor modules. The remainder of the modules will be considered in subsequent stages of the research, and therefore are not described in this work.

The Procedural module corresponds to an area of the mammalian brain called Basal Ganglia [5] which is in charge of functions such as rule matching, conflict resolution, cognition, learning, and selection and the execution of actions. This module is composed of several components called *Specialist Behaviors (SB*), which are organised horizontally, and three sub-modules which are distributed vertically. The three sub-modules are: the connectionist module, the autopoietic machines module, and the productions module. The horizontally-formed components of the procedural module will be explained in the next section.

The Connectionist module (found at the emergent level of the diagram) models innate skills, which require less processing time in comparison with other deliberation processes. It therefore uses the *Backpropagation Neural Networks (BNN)* which is more appropriate for enacting reactive reasoning.

The Autopoietic Machines module is formed by multiple self-organized and self-regulated systems, where each one models a set of sub-symbolic rules on the basis of autopoietic principles [11].

The Productions module manages different sets of symbolic rules as follows:

- Expert Rule (ER): this rule simulates either the innate knowledge passed on genetically by an evolutionary process, or the knowledge acquired by a previous experience.
- Sub-symbolic Extraction Rule (SER): this rule represents the rules acquired by knowledge extraction using a bottom-up learning method which refines the rule set by means of experience, generalization, and specialization processes.

The Intentional module represents the agent's intentionality through goal and plan generation at the cognitivist level, as well as prospection strategies and internal simulation at the emergent level. This module will have a declarative knowledge representation composed of chunks of semantic and episodic memories which are accessed indirectly, as proposed in [8]. This module is able to predict the outcomes of the actions produced by the system and construct a model of events for controlling perception through stimuli anticipation.

The Attention module has the responsibility of interpreting the perceived information through the sensors and transforming it into percepts (sensory inputs translated into property/value pairs). This module is composed of a network of nodes, called the working memory, which is in charge of keeping information of the agent's current state, including elements of external perception, internal motivation, goals and intentions.

Furthermore, the Attention module is based on Global Workspace theory and Theater Metaphor for Consciousness [6]. This module has the responsibility for coordinating the execution of several SBs, which compete and cooperate in order to get the attention focus (consciousness). It is also responsible for broadcasting those SBs that have not been selected to obtain the attention focus but which have the capacity to generate new inferences, consolidate acquired knowledge and infer anticipatory signals. The most innovative aspect of this module is the behavior orchestration managed by a mechanism that uses *Gene Expression Programming* (GEP), an evolutionary programming algorithm proposed by Ferreira [12]. This mechanism will be discussed later in section 4.

3. EPIGENETIC APPROACH: IMPLICIT LEARNING AT EMERGENT LEVEL

The epigenesis refers to heritable changes in phenotype (appearance) or gene expression, caused by mechanisms other than changes in the underlying DNA sequence. Therefore, the epigenesis represents the final tuning process by means of each individual adapts efficiently to its environment from the abilities included in its genetic code.

In our work, the epigenetic approach references to the mechanisms that allow agent modifying some aspects of its both internal and external structure as a result of interacting with its environment, in other words, “learning”. Therefore, we propose the development of two main approaches which intend to simulate the most evident epigenetic systems observed in nature: the central nervous system and the immune system.

In the following sections, we describe the proposed computational model and their functionality for each one of these systems.

3.1 Connectionist Module

Rumelhart [13] and Sun [8] have provided support in regards to the argument that the *Backpropagation Newural Networks* (BNN) can represent the sub-symbolic and distributed representation of implicit knowledge. A BNN is a type of neural network which connections among nodes are adjusted to produce the optimal output.

Sun [8] explains that the units in BNN are capable of accomplishing computations but are sub-symbolic and not individually meaningful; that is, they generally do not have associated semantic labels. Just like neurons, nodes are not individually useful. Since implicit knowledge represents the lower-level, almost automatic processing of information, it is easy to see why BNN are used.

In our work, the connectionist module is organized in groups of BNN, each one representing the sub-symbolic specialization of a task. Each specialized BNN is classified according to its purpose, in order to filter the perceived stimuli from environment and select the respective reactive actions.

3.2Autopoietic Machines Module

An autopoietic machine is a system organised as a network of processes of production (transformation and destruction) of components which: (i) through their interactions and transformations continuously regenerate and realize the network of processes (relations) that produced them; and (ii) constitute it (the machine) as a concrete unity in space in which they (the components) exist by specifying the topological domain of its realization as such a network [11].

A good example of autopoietic machines is the immune system's ability to distinguish between self and non-self. Varela [11] has been pointing out for some time that this is an observed behavior, produced by the operational dynamics of the immune system in its environment, and that it is wrong to look for some discriminatory recognition mechanism within the immune system.

A computational model of immunological system is the *Artificial Immune System* (AIS) [15], an adaptive system inspired by immunological theory and immune functions, models and principles observed in nature as defense mechanisms.

Thus, we propose an AIS as autopoietic machine which starts with an sensory input data set (antigens) that stimulate an immune network, and then goes through a dynamic process until it reaches some type of stability. Specifically, each autopoietic machine is based on AiNet [15], a model which implements a discrete immune network that has been developed for data compression and clustering and later for optimization.

As in the connectionist module, the autopoietic machines are separated each one according to its specialization, focusing on a specific cognitive skill and communicating to both the BNN and the group of symbolic rules associated.

Each AIS (as an autopoietic machine) incorporates an evolutionary mechanism that uses a *Genetic Algorithm* in order to discover new knowledge as rules (antibodies). Additionally, each AIS uses a credit assignment method which has the ability to resolve the antibody selection given an specific antigen (sensory input) as in [18]. Some of the immunological mechanisms we have developed are: clonal selection, clonal maturation, diversification, cutting, and different kind of meta-dynamics. Finally, a reinforcement signal is used to reward the performance of the selected antibodies at each AIS.

3.3 Vertical Integration: Specialist Behaviors

The Specialist Behaviors (SB) are proposed as hybrid units of procedural processing which are in charge of specializing the cognitive skills of the architecture. These specialists are hybrid because of incorporation of both symbolic and sub-symbolic elements at the procedural module.

In particular, the procedural module arranges a set of SBs distributed vertically, every one made up of each horizontally-formed component (i.e., an specific SB has one BNN, one AIS, one ER set, and one SER set, as in Figure. 1).

Thus, SBs help the architecture to articulate the set of skills because of each SB attends on an specific set of stimuli signals and gives an appropriated response to the environment. Accordingly, each SB can be formalized as follows:

<SB> = <ER> ∪ <SER> ∪ <AM> ∪ <BNN>

The purpose of including multiple components in an SB is that each one compensates the weaknesses of the rest. For example, BNN are often better at making forward inferences about object categories than ERs, whereas ERs are often better at making forward inferences involving causal change than neural networks. AIS is able to make both kind of inferences from implicit representations but it involves more processing time discovering new rules than the other two components.

In addition, a reinforcement signal (as a factor of learning in the procedural module) is used to modify the future responses of the agent. This is achieved through adjusting the connections in BNNs, rewarding the activated antibodies in AISs, and extracting sub-symbolic knowledge from emergent level in SERs.

Several approaches have been proposed for BNN reinforcement, often reinforcement learning can be used, especially Q-learning as in [14]. In this learning setting there is no need for external teachers providing desired input/output mappings.

At each AIS, according to the environmental reinforcement (positive or negative), the fitness of the executed antibody will be either awarded or punished. Some parameters to compensate the antibody fitness are defined in the bucket brigade mechanism [18].

In SER, a successful action recommended by the emergent level is taken, and a new symbolic rule that represents it is developed. The new rule is then revised and based on future states of the world. Depending on whether the rule is successful or not, it will be expanded or narrowed, thus generalization and specialization process are used as in [8]. The agent will attempt to apply a successful rule to broader situations, meaning more opportunities for it to be applied.

Therefore, we stated that the main architectonic features of procedural module are parallelism, behavior specialization, and autonomy.

4. ONTOGENETIC APPROACH: BEHAVIOR ORCHESTRATION

Ontogenetic principles involve all the mechanisms in charge of developing an agent on the basis of the stored information in its own genetic code without interposing the environment influence. Additionally, it defines the history of structural change in a unity without the lost of organization that allows that unity to exist. Some outcomes of these principles as self-replication and self-regulation properties in biological systems can be valued.

In our work, the ontogenetic approach is simulated through the interaction among different modules: the Attention module, the Goal module, the Anticipatory Module, and the SBs in Procedural module. The main idea in this approach is that the attention module supported by Global Workspace theory, orchestrates the different SBs in such a way that either cooperate or compete among them.

The attention module defines a set of *attention machines* (AM), which are systems implemented as attention fixations that execute algorithms by sequences of SBs. Each AM has a set of premises that describe the pre-conditions of AM activation, the stream of SBs, and the post-conditions that will have to guarantee after the execution of the stream. Te pre-conditions indicate the goals, expectations, emotions, and stimuli (provided by the working memory) which the agent will have to process and satisfy at any given time. The stream of SBs is a sequence of SBs and relations among them which describes how the agent must behave in a particular situation. Finally, post-conditions are a set of states and new goals generated after the execution of the stream. The typical AM structure is depicted in Figure 2.

The main question that addresses the development of the attention module is how it will be able to arbitrate autonomously the execution of SBs in each AM given a set of stimuli, expectations, goals, and emotions?

As a result, we propose an evolutionary model based on GEP [12] that is used to evolve the AMs in order to generate an appropriated behavior orchestration without defining a priori the conditions of activation about each SB.

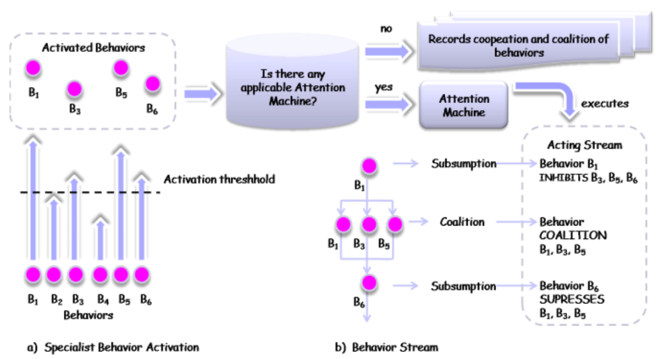


Figure 2. Information flow of an Attention Machine

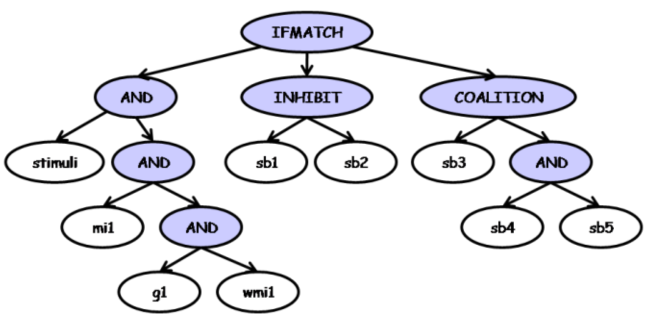
GEP uses two sets: a function set and a terminal set. Our proposed function set is: IFMATCH, AND, OR, NOT, INHIBIT, SUPRESS, AGGREGATE, COALITION, and SUBORDINATION. The AND, OR and NOT functions are logic operators used to group or exclude subsets of elements (SBs, goals, working memory items, etc.). The conditional function IFMATCH is an applicability predicate that matches specific stimuli. This function has three arguments; the first argument is the rule's antecedent, an eligibility condition which correspond with a subset of sensory inputs, motivational indicators (internal states, moods, drives, etc.), and working memory elements, which model the agent's current state. All elements of these subsets are connected with logic operators. If the whole set of conditions exceeds a threshold, then the second argument, the rule's consequent, is executed, otherwise the third argument is executed. Second and third argument should be a set of functions such as INHIBIT, SUPPRESS, AGGREGATE, COALITION, or SUBORDINATION, or maybe an AND/OR function connecting more elements when is necessary.

The INHIBIT, SUPPRESS and AGGREGATE functions have two SBs as arguments (SB_A, SB_B) and indicate that SB_A inhibits, suppresses, or aggregates SB_B.

The COALITION/SUBORDINATION functions, instead of binomial functions mentioned above, perform a set of SBs. The COALITION function describes a cooperation relationship among SBs where actuators may activate multiple actions. The SUBORDINATION function defines a hierarchical composition of SBs which are activated in a specific sequence.

In addition to, the terminal set is composed by the SB set, the motivational indicators, the goal set, and the working memory elements. Additionally "do not care" elements are included so whichever SB, motivational indicator, goal, or working memory item can be referenced.

Each AM represents its conditions using a chromosomal structure that can be transformed into expression trees, i.e., one agent might have a chromosome, as shown in Figure 3, which is a valid eligibility rule because both the antecedent and the consequent of IFMATCH function matches to each required argument type.



Eligibility Rule:
IFMATCH:
    (Stimuli) AND (mi_1) AND (g_1) AND (wmi_1)
THEN:
    INHIBIT (sb_1, sb_2)
ELSE: COALITION ((sb_3) AND (sb_4) AND (sb_5))

Figure 3. Tree structure representation of a chromosome (eligibility rule). From Karva notation [17] this chromosome would be: {[IFMATCH] [AND] [INHIBIT] [COALITION] [stimuli] [AND] [sb_1] [sb_2] [sb_3] [AND] [mi_1] [AND] [sb_4] [sb_5] [g_1] [wmi_1]}, where mi_n is the motivational indicator n, sb_n is the specialist behavior n, g_n is the goal n, and wmi_n is the working memory item n

Analyzing this eligibility rule we can infer that the agent has five SBs: sb_1, sb_2, sb_3, sb_4, and sb_5, the first inhibits the second one, and the last three make a coalition when the pre-conditions include: the motivational indicator mi_1, the goal g_1, the working memory item wmi_1, and an specific stimuli. However, these chromosomes (eligibility rules) do not have always a valid syntax, so the GEP mechanism is used to evolve the chromosome until it becomes in a valid syntactic rule.

Each agent has a multigenic chromosome, that means, each chromosome has a gene set where each gene is an eligibility rule like in the example, so the agent has several rules (genes) as part of its genotype and each one is applied according to the situation that matches the rule antecedent. Each gene becomes to a tree representation and afterwards some genetic operators are applied among genes of the same agent and genes of other agents, as in [12]. Some of these genetic operators are: selection, mutation, root transposition, gene transposition, two-point recombination and gene recombination, in order to evolve chromosomal information.

After certain number of evolutionary generations, valid and better adapted AMs are generated. A roulette-wheel method is used to select individuals with most selection probability derived from its own fitness. Fitness represents how good interaction with environment during agent's lifetime was.

5. PHYLOGENETIC APPROACH: CO-EVOLUTIONARY MECHANISM

In biology, phylogenesis (evolution) collects all those mechanisms which, leaded by natural selection, have given place to the broad variety of species observed in nature. Evolutionary mechanism operates in populations and as a result, it gets a genetic code which allows individuals of a concrete population to adapt to the environment where they live in.

On the basis of phylogenetic theory [11], a co-evolutionary mechanism is proposed to evolve fine-grained units of knowledge through the multi-agent system, taking the foundation of meme and memetic algorithms. The term "meme" was introduced and defined by Dawkins [16], as the basic unit of cultural transmission or imitation that may be considered to be passed on by non-genetic means. In our work, each meme contains a symbolic and sub-symbolic representation of knowledge, and also a set of indicators such as demotion, reliability, rule support and fitness.

As a result, our co-evolutionary mechanism is based on a Memetic Algorithm [17] which is inspired by both Darwinian principles of natural evolution and Dawkins' notion of a meme. This mechanism can be viewed as a population-based hybrid genetic algorithm coupled with an individual learning procedure capable of performing local refinements.

Most evolutionary approaches use a single population where evolution is performed; instead of this, in our co-evolutionary approach, the SBs are discriminated in categories and make them evolve in separate pools without any interaction among themselves.

After certain period of time a co-evolutionary mechanism is activated. For each behavior pool, a stochastic selection method is executed, where those SBs that had the best performance (fitness) will have more probability to reproduce. Then, a crossover genetic operator is applied among each pair of selected SBs and some memes are both selected and interchanged with other ones. In Figure 4 is shown the basic co-evolutionary mechanism.

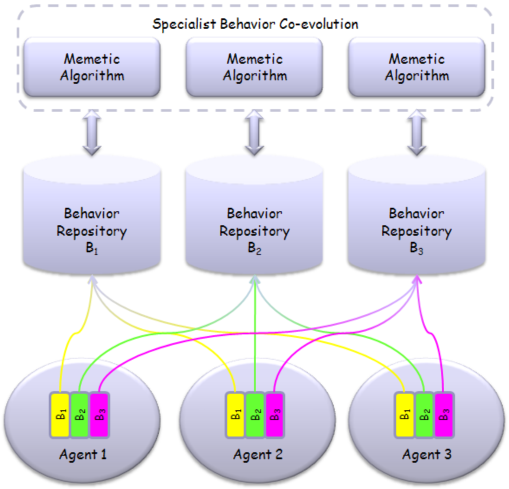


Figure 4. Co-evolutionary mechanism

It is important to notice that the proposed Memetic Algorithm has the responsibility of:

- evolving the set of memes acquired by each agent through the interaction with its environment in a continuous perception-action loop,
- broadcasting and sharing the knowledge acquired by each agent reducing the curve of learning new knowledge, and
- generating new knowledge units as memes through the evolution of memes obtained by learning.

6. EXPERIMENTATION

In order to evaluate the proposed cognitive model, following aspects were considered:

- Learning convergence and knowledge generalization of the procedural module,
- Analysis of eligibility rules evolved by GEP in the attention module, and
- Learning convergence of the co-evolutionary mechanism.

An artificial life environment called Animat (animal + robot) is proposed to test the experiments. The environment simulates virtual agents competing for getting food and water, avoiding obstacles, and so forth. Each animat, driven by an agent, disposes a set of 8 proximity sensors around itself.

Some environmental changes were introduced during the experimentation in order to validate the generalization and adaptation ability.

The experiments were made using three agent behaviors: looking for food (SB-eat), avoiding obstacles (SB-avoid-obstacles), and escaping from predators (SB-escaping-from-predators).

Thus, some experiments designed to evaluate the performance aspects mentioned above are described next.

6.1 Learning and Knowledge Generalization

In this experiment we chose three types of environments gradually increasing in complexity. The purpose was to validate the adaptation and generalization level acquired by the agent through learning.

Thus, the main idea is to see how autopoietic machines controlled by AIS can serve as a bridge between BNN (in emergent level) and production rules (in cognitivist level), integrating the implicit and explicit knowledge from each one. It is important to notice how the AIS algorithm adapts to new environmental changes and broadcasts the refined knowledge to other levels at procedural module.

In Figure 5 is depicted the three environments and the desired paths that the agent should cross. The environment in Figure 5a. was a basic learning scenario where the agent had to follow a simple path. In Figure 5b. some little changes were included, and in Figure 5c. the environment was more complex because of a new path of food and more elements that were introduced.

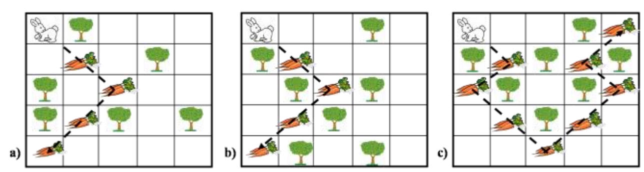


Figure. 5. Animat environments. a) basic environment, b) modified environment, and c) complex environment.

In the first environment, the three sub-modules at the procedural module of the agent's architecture were trained. In the other two environments the agent was tested using the previous knowledge acquired from the first environment. Table 1 shows the learning parameters used for training.

Table 1. Learning parameters of procedural module

Parameter	AIS	BNN
Life Tax	0.005	-
Bid Tax	0.003	-
Cloning Rate per rule	4	-
Mutation Rate per rule	2	-
Similarity Threshold	0.8	-
Alpha α	-	0.1

Beta β	-	-
Delta δ	-	0.02
Gamma γ	-	0.8
Lamda λ	-	0.8
Processing Layers	-	3
Number of epochs	40	
Number of runs per epoch	20	

Figure 6 illustrates the convergence curves obtained for a harmonic mean value for 50 runs in the three different environments. It is important to notice how the agent in the basic environment took more time for training but the learning curve decreased when the agent was tested on the other two environments. In the first experiment, the unified algorithm outputs of procedural module converged in 33 learning epochs whereas in the experiments 2 and 3 the algorithms converged in 5 and 3 learning epochs respectively.

This demonstrated that the agent has generalized the knowledge acquired in the previous training phase and it was able to adapt quickly to new environments.

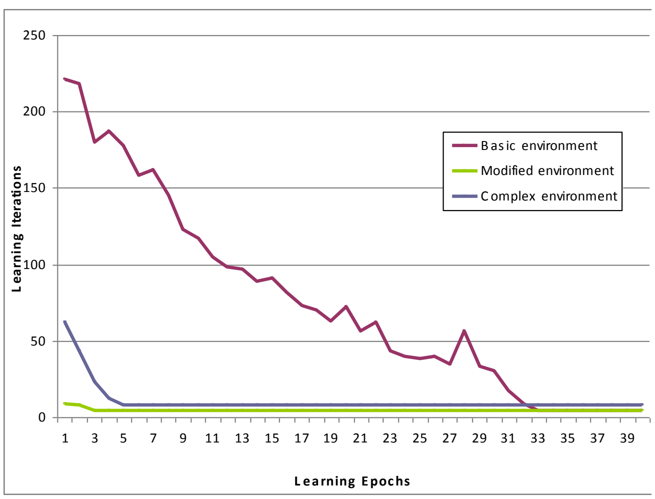


Figure. 6. Learning curves in the three different environments

Thus, learning convergence guided by the genetic algorithm in AIS was demonstrated. Additionally, the knowledge generalization in the experiments 2 and 3 starting from the rules learned in the experiment 1 was probed.

6.2 Analysis of eligibility rules evolved by GEP

After the attention machines in the attention module have evolved during a specific number of generations, we analyze the final eligibility rules of the best adapted agents where emergent properties arose.

Initially, we present an initial eligibility rule which has syntax conflicts; therefore an evolved eligibility rule syntactically well-formed emerges from GEP.

We have chosen an eligibility rule from a non-trained agent and afterwards we show the same evolved eligibility rule but now it has no syntax conflicts and also it's better well-suited than its predecessor.

```
Eligibility Rule at generation 0:
IFMATCH:

{food},{tree},{empty},{empty},{empty},{empty}
,
   {empty},{tree} AND {goal-is-eat}
THEN:
   {SB-eat} INHIBITS {SB-avoid-obstacles} AND
   {SB-avoid-obstacles} SUPRESSES {SB-eat}
ELSE:
   SUBORDINATION {SB-avoid-obstacles} AND
   {SB-eat}
```

The above eligibility rule means that when the agent senses “food” around it, it must do something to get the food while is avoiding obstacles, but is contradictory because `{SB-eat}` can't inhibit `{SB-avoid-obstacles}` while `{SB-avoid-obstacles}` is suppressing `{SB-eat}` at the same time. So, the evolved consequent of the eligibility rule after 17 epochs is:

```
IFMATCH:

{food},{tree},{empty},{empty},{empty},{empty}
,
   {empty},{tree} AND {goal-is-eat}
THEN
   COALITION    {SB-eat}    AND    {SB-avoid-
obstacles}
ELSE
   {SB-avoid-obstacles} INHIBITS {SB-eat}
```

It is important to notice that evolved eligibility rule does not present any syntax conflict and is a valid rule which forms a coalition among `{SB-avoid-obstacles}` and `{SB-eat}` behaviors when the agent reads food and obstacles around it. Otherwise, the agent always will execute the rule: `{SB-avoid-obstacles}` inhibits `{SB-eat}`, focusing the agent attention on obstacles because of `{SB-eat}` behavior has a lower priority and is less reactive than `{SB-avoid-obstacles}` behavior.

6.3 Learning convergence of the co-evolutionary mechanism

This experiment examines if the fitness of every separate behavior pool increments gradually until it reaches a convergence point while evolution takes place. The experiment was carried out with the parameters on Table 2.

Three behavior pools were selected for the experiment:

- avoiding-obstacles,

- looking-for-food, and
- escaping-from-predators.

The results are depicted in Figure 7.

Table 2. Co-evolution learning parameters

Parameter	Value
Epochs	50
Number of epochs per run	50
Crossover probability	0.7
Mutation probability	0.3
Mutation rate η	0.85
Mutation rate θ	0.25
Mutation rate κ	1.03
Mutation rate γ	0.01

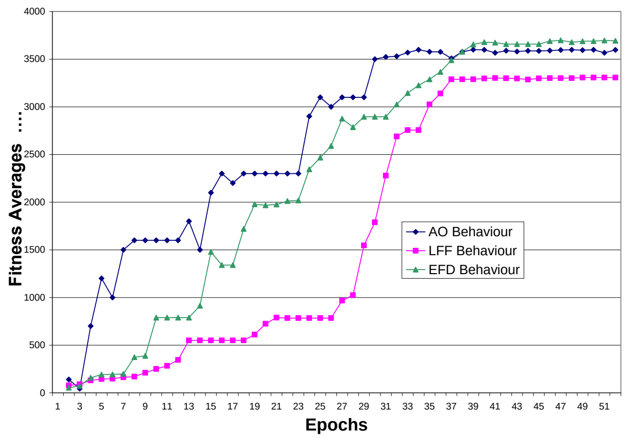


Figure. 7. Evolution convergence rate in 3 behaviour pools

Figure 7 depicted some differences in each learning curve, because of environmental conditions, however the pools always tried to converge and reach certain knowledge stability at the same number of epochs (approximately after 30 epochs), that means the evolution has been effective and each behavior pool has established a coherent knowledge base getting a consensus among its own behavior instances and about what the "behavior category" should do.

7. CONCLUSIONS

The experimentation demonstrates that specific parameter configurations in AIS, BNN, GEP and the Co-evolutionary mechanism are required to reach certain capabilities such as robustness, adaptability and learning, in the overall system. Nevertheless, emergent properties did not arise always or in a fast way. In several experiments animats died quickly and they could not learn to survive, but this allowed appearing more robust individuals later.

The evolutionary mechanisms used in this work, provided a plasticity feature allowing the agent to self-configure its own underlying architecture; thus, it can avoid creating exhaustive and extensive knowledge bases, pre-wired behavior structures of behaviors, and pre-constrained environments. Instead of this, the cognitive agents which use our architecture only need to interact with an arbitrary environment to adapt to it and take decisions in both a reactive and deliberative fashion.

In the experimentation, the emergent properties were difficult to discover because it took a lot of time to evolve the overall system despite of using a multi-agent platform with a distributed configuration. Maybe, it can be similar to the natural evolution where adaptation occurs slowly and sometimes produces poor adapted creatures.

In our future work we expect to continue working on designing more adaptive and self-configurable architectures, incorporating intentional and meta-cognition modules. One concrete application of this research will be the development of a cognitive module for Emotive Pedagogical Agents where the agent will be able to self-learn of perspectives, believes, desires, intentions, emotions and perceptions about itself and other agents, using the proposed approach which will be responsible of driving the cognitive architecture.

8. ACKNOWLEDGMENTS

Supported by the Programme Alban, the European Union Programme of High Level Scholarships for Latin America, scholarship No. E05D056455CO".

9. REFERENCES

[1] M. L. Anderson, "Embodied cognition: A field guide," Artificial Intelligence, vol. 149, no. 1, pp. 91–130, 2003.

[2] A. Berthoz, The Brain's Sense of Movement. Cambridge, MA: Harvard Univ. Press, 2000.

[3] D. Vernon, G. Metta, and G.Sandini, "A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents," IEEE Trans. Evolutionary Computation, vol. 11, No. 2, April 2007.

[4] P. Rosenbloom, J. Laird, and A. Newell, Eds., The Soar Papers: Research on integrated intell. Cambridge, MA:MIT Press, 1993.

[5] J. R. Anderson, D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, and Y. Qin, "An integrated theory of the mind", Psy. Rev., vol. 111, no. 4, pp. 1036–1060, 2004.

[6] M. P. Shanahan and B. Baars, "Applying global workspace theory to the frame problem," Cognition, vol. 98, no. 2, pp. 157–176, 2005.

[7] Breazeal, "Emotion and sociable humanoid robots," Int. J. Human-Computer Studies, vol. 59, pp. 119–155, 2003.

[8] R. Sun, E. Merrill, T. Peterson, "From implicit skills to explicit knowledge: A bottom-up model of skill learning". Cognitive Science, vol. 25, pp. 203-244, 2001.

[9] N.L. Cassimatis, Adaptive Algorithmic Hybrids for Human-Level Artificial Intelligence. Advances in Arti. Intell., 2007.

[10] S. Franklin, "The LIDA architecture: Adding new modes of learning to an intelligent, autonomous, software agent". In Proc. of the Int. Conf. on Integrated Design and Process Technology, 2006.

[11] H. R. Maturana and F. J. Varela, "Autopoiesis and Cognition: The Realization of the Living", ser. Boston Studies on the Philosophy of Science. Dordrecht, Holland: D. Reidel Publishing Company, 1980.

[12] C. Ferreira, Gene Expression Programming: A new adaptive algorithm for solving problems, Complex Systems, vol. 13, pp. 87-129, 2001.

[13] D. Rumelhart and J. McClelland, "Parallel Distributed Processing: Explorations in the Microstructures of Cognition", MIT Press, 1986.

[14] [14] C. Watkins and P. Dayan, Q-learning. Machine Learning, vol. 8, num. 3, pp. 279-292, 1992.

[15] L. N. de Castro, J. "Timmis, Artificial Immune Systems: A New Computational Intelligence Approach", 2002.

[16] R. Dawkins. "The Selfish Gene". Oxford University Press. 1989

[17] P. Moscato, "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms". 1989.

[18] D. Romero, L. Niño, An Immune-based Multilayered Cognitive Model for Autonomous Navigation, CEC, Vancouver, 2006, 1115-1122.