Validating an Integrated Cognitive Architecture
via Intelligence Testing with Virtual Animals

Ben Goertzel, Cosmo Harrigan, Joscha Bach

Abstract: The "cognitive synergy" hypothesis states that, in order to achieve advanced general intelligence using limited computational resources, it is necessary for different cognitive processes corresponding to different kinds of memory to interact intimately, in a way that allows them to help each other squelch combinatorial explosions. This hypothesis lies at the core of the CogPrime Artificial General Intelligence design, an integrative cognitive architecture that is partially implemented in the current OpenCog software system. The proposed project would explore and attempt to validate the cognitive synergy hypothesis, via utilizing the OpenCog system to control a virtual parrot in a game world, and studying how the parrot performs on virtual-world implementations of various "animal-level intelligence" tests (including simple linguistic tests related to parrots' language abilities). Varying the degree and type of cognitive synergy inside the AI system, and observing the consequent impact on the virtual parrot's intelligence, should provide interesting information about cognitive synergy, OpenCog, CogPrime, and perhaps animal intelligence as well.

Introduction

One hypothesis regarding the nature of general intelligence is that, to achieve human-like or greater general intelligence using feasible computational resources, a significant amount of "cognitive synergy" will be required. That is: a significant amount of inter-operation between different cognitive processes focused on learning that regards different types of memory, in such a way that each cognitive process helps the others to scale and avoid the "combinatorial explosions" that habitually plague AI algorithms. This concept of cognitive synergy is a dynamical correlate to Karmiloff-Smith's (1992) classic notion of "representational redescription" in which different cognitive modules learn to speak each others' representational languages; it may be mathematically formalized in terms of algorithmic information theory, and quantitatively measured in operational AI systems (Goertzel, Pennachin and Geisweiller, 2014).

The CogPrime AGI design comprises an ambitious theoretical framework and practical blueprint for construction of an Artificial General Intelligence system with capability at the human level and perhaps ultimately beyond, founded centrally on the cognitive synergy hypothesis. CogPrime is described in moderate detail in the two-volume set *Engineering General Intelligence* (Goertzel, Pennachin and Geisweiller, 2014). A relatively concise overview is available online[1] at (Goertzel, 2012). Significant parts of the CogPrime design have been implemented in the OpenCog open-source software codebase, which is currently under active development by an international community.

---

[1] http://wiki.opencog.org/w/CogPrime_Overview

To partially understand the concepts underlying the CogPrime design, note that there has been a dichotomy in the AI field in recent years between the machine learning and probabilistic reasoning communities, which focus on the creation of advanced learning algorithms, and the cognitive architecture community, which focuses on human-like architectures for learning, supported by relatively simplistic learning algorithms. The OpenCog architecture is unique in its integration of a number of powerful learning algorithms (e.g. PLN, a probabilistic logic engine; MOSES, an evolutionary program learning algorithm; ECAN, a nonlinear-dynamical attention allocation framework; DeSTIN, a deep learning perception algorithm) within a human-like cognitive architecture. The integration of these multiple learning algorithms is achieved via a combination of mechanisms motivated by the concept of cognitive synergy.

CogPrime is a very large design and full implementation and testing would take a number of years for a substantial team. Current OpenCog work is largely aimed at specific goals of interest to academic funding sources, e.g. natural language comprehension and robot control; or else involves improving specific functionalities useful for immediate commercial work in e.g. genomics or financial analysis. This work is all quite valuable for progressing the overall OpenCog/CogPrime project, but due to its focus on specific practical capabilities is not generally optimal for advancing the central aspects of CogPrime as an AGI initiative.

The present proposal suggests a relatively modest initiative aimed at utilizing the OpenCog codebase to realize and test the cognitive synergy hypothesis. This initiative would utilize the CogPrime design in a simple context, aimed not at achieving practical applied capabilities but rather at rigorously exploring one of the key scientific ideas underlying CogPrime.

Cognitive Synergy

Broadly speaking, "cognitive synergy" is a term for the phenomenon wherein multiple cognitive processes, all working together, achieve intelligence beyond what would be predicted from looking at what the cognitive processes do individually. More specifically, in the theoretical framework underlying CogPrime it is proposed that cognitive processes centered on different types of memory (e.g. declarative, procedural, episodic, sensory) must display significant synergetic behavior in order for human-like general intelligence to emerge. Without interaction between these different cognitive processes, it is proposed, each of the processes would utilize excessive computational resources in carrying out learning. The interaction allows different cognitive processes to guide each other in a way that allows each process to avoid bad choices and move in useful cognitive directions.

A key example of the "cognitive synergy" concept is the synergy that is proposed to be achievable between:

- Learning of **declarative knowledge** via probabilistic inference, using the PLN reasoning system
- Regulation of **attention**, using nonlinear-dynamical attention allocation as embodied in the ECAN (Economic Attention Networks) framework
- Acquisition of **procedural knowledge**, using the MOSES automated program learning system

The potential for enhancing learning capabilities via leveraging synergy between these three cognitive processes, will be the focus of the proposed project.

The argument as to why this particular example of cognitive synergy is important, has been presented in detail in prior publications and won't be elaborated in detail here (Goertzel, Pennachin and Geisweiller, 2014). The core intuition, however, is that different kinds of memory are structured

very differently and demand different kinds of learning approaches. Semantic and declarative knowledge are better suited for logic-type approaches (though since most knowledge in any real-life context is uncertain, some sort of probabilistic logic is needed). Procedural knowledge is, in many cases, better suited for some sort of "subsymbolic" approach; and evolutionary learning is one frequently powerful approach for subsymbolic learning. In extensive commercial data analysis work carried out by two of the authors (Goertzel and Geisweiller) in multiple domains including genomics and financial analysis, the MOSES probabilistic evolutionary learning approach has proved more effective than standard genetic programming. Attentional knowledge (i.e. contextual knowledge affecting control of selective attention, cf. DiQuattro and Geng, 2011) is most effectively handled by some "activation spreading" type mechanism; ECAN is one such mechanism which has relatively tractable mathematics, and has been shown compatible with psychologically realistic models of the relation between working memory and long-term memory. The crux of the proposed synergy is to get learning regarding declarative, procedural and attentional knowledge working together; this will be effected in practice via leveraging high-quality learning algorithms associated with each of these memory types, that are already integrated into OpenCog.

Currently these three CogPrime components all exist in the OpenCog codebase, with varying levels of maturity (MOSES is quite mature and has been used in many commercial projects; ECAN and PLN are earlier-stage and have been used mainly for research purposes). However, these components have not yet been used together in any single task or application. Doing so would have significant scientific value as it would constitute the first significant empirical test of the "cognitive synergy" hypothesis underlying CogPrime.

Virtual Animals

Perhaps the simplest way to explore the potential of cognitive synergy for enhancing general intelligence is to experiment with animal-like intelligence. Some existing OpenCog applications have demonstrated cognitive synergy in limited ways; but in the context of complex practical applications, it is often difficult to study cognitive phenomena in a rigorous way. To understand the role and nature of cognitive synergy more thoroughly, from a scientific perspective, there is an advantage to studying it in a context that is relatively simple, yet still intuitively rich and meaningful. Toward that end, the context of a "virtual animal" appears favorable.

What is proposed here is a project involving creating a "virtual animal" in a 3D video-game world, that achieves intelligent behaviors similar to those of actual animals, via leveraging cognitive synergy phenomena within OpenCog. This project is proposed to take 18 months. It is viewed as the first stage in a series of projects focused on achieving progressively more complex intelligent behaviors using OpenCog. For example, a natural project to do immediately after the proposed work would focus on embodied language learning -- teaching the virtual animal English language pertaining to the game world in which it exists (the work proposed here will involve very simple English, at the level of high-functioning parrots like Pepperberg's Alex; a subsequent project could then aim at the "toddler level".)

Broadly speaking, to meaningfully explore and evaluate the value obtainable via synergetic interaction of PLN, ECAN and MOSES requires three ingredients:

1. Implementations of these three cognitive processes within a common software framework
2. An environment and task (or set of tasks) for an integrative AI system to do, involving these three cognitive components in a meaningful way
3. A method of measuring the intelligence of the AI system, and thus of gauging the synergetic value obtained via interaction of the three cognitive components

For these three aspects, we propose here to

1. Work with the existing implementations of PLN, MOSES and ECAN in the OpenCog codebase, enhancing their functionality as needed
2. Utilize the Unity3D game world currently in use with OpenCog, or a variation thereof, for research on planning and motivation
3. Implement measures of system intelligence inspired by biological animal intelligence testing (Commons and Ross, 2008)

Many types of virtual animal could help work toward these goals, but because we would like to explore primitive instantiations of linguistic behaviors, we propose to work with a virtual parrot. Parrot-like use of human language is of course simpler than full-scale human language, but can be surprisingly sophisticated (Pepperberg, 2002), and provides an interesting first step toward fuller experience-based AGI language learning and functionality.

Figure 1: Screenshot from a Second Life "virtual parrot" prototype

experiment conducted by Dr. Goertzel and colleagues in 2007

We will use OpenCog to control a virtual parrot in a game-world environment, having the virtual parrot carry out basic tasks corresponding to those in a natural animal's life, and measuring the animal's intelligence via methods drawn from those used by animal psychologists to measure the intelligence of natural animals. The animal behaviors studied will include simple verbal behaviors such as have been observed in real parrots (with parallels in chimpanzees as well), as well as more typically animal-like sensorimotor behaviors.

This will enable a careful exploration and (we conjecture) validation of the value of cognitive synergy for intelligence. As an additional benefit, it will also provide a solid foundation of integrated intelligence on which to build more sophisticated CogPrime-based systems – including a next-phase embodied agent capable of more advanced language processing, robotic OpenCog systems integrating sophisticated vision processing, and so forth.

For the proposed project, we will create a customized game world specifically for parrot intelligence, creating objects and situations oriented toward parrot intelligence testing as to be outlined below. This will comprise a natural extension of prior work done using OpenCog to control virtual animals in virtual worlds, e.g. recent work in a Unity3D based Minecraft-like world, and earlier work in a Multiverse based game world.

Figure 2: Screenshot of Unity3d "blocks world" currently being used for OpenCog experimentation

Figure 3: OpenCog's MOSES component has been used to learn procedures enabling virtual dogs to play fetch, dance and carry out other simple behaviors, in a game world built using the Multiverse game engine

It is also worth briefly noting what is not in the focus of the proposed work: for instance, sensorimotor learning and episodic memory. For real animals, the richness of sensorimotor data is the source of most of their creative learning of new behaviors. Episodic memory, of an individual animal's life history, also obviously plays a major role in animal learning (and ties together with sensorimotor richness, because it is this richness that gives the episodes in an animal's memory so much variety). These other aspects are also very important for creating AIs with animal-level behavior. However, we have chosen to focus on one particular cognitive synergy for this particular research project, in the interest of initial tractability and clarity.

Evaluation: Animal Intelligence Assessment

To explore and evaluate the impact of cognitive synergy on the intelligence of an OpenCog system controlling a virtual animal, we will implement a suite of simple but wide-ranging "animal intelligence tests" derived from the animal psychology literature.

Though a great deal of work has been done regarding animal intelligence, in cognitive ethology and related fields, there is no standard, accepted cross-species approach to animal intelligence assessment (Shettleworth, 2010). Commons and Ross (2008) have articulated a useful general framework for cross-species "g-factor" (general intelligence) assessment, but this work still falls well short of providing a concrete "cross-species IQ test".

Commons and Ross, extrapolating from human IQ tests, discuss the following aspects of animal intelligence:

- **Sensorimotor**: E.g. rats can learn to distinguish the "odd" food sample, the one that is different from the other samples in a batch, [2]
- **Nominal**: Associating words with sensory stimuli; associating multiple sample stimuli with a common "nominal" stimulus. This has been shown for example in pigeons. [3]
- **Sentential**: Irene Pepperberg's famous African Grey parrot Alex (which died in 2007 at the age of 30) could count two objects ("one, two") and speak in sentences that organized nominal labels and words. When a new question was introduced, "What matter [is this] four corner blue [object made of]?" he correctly responded, "wood."
- **Preoperational**: Comparing sizes of sets. Using tools to achieve goals (crows can do this, see Hunt, 1996 and Hunt et al, 2004). Rule learning; for example, Murphy *et al.* (2008 trained rats to discriminate between visual sequences. For one group ABA and BAB were rewarded, where A="bright light" and B="dim light." Other stimulus triplets were not rewarded. The rats learned the visual sequence, although both bright and dim lights were equally associated with reward. More importantly, in a second experiment with auditory stimuli, rats responded correctly to sequences of novel stimuli that were arranged in the same order as those previously learned.
- **Primary**: E.g. counting, associating numerals with objects. Rhesus monkeys and parrots can do this (Washburn et al, 2001).
- **Concrete**: Using simple tools, e.g. Kanzi the Bonobo was observed using sharp flakes, and testing candidate flakes for sharpness (Savage-Rumbaugh, 1994; de Waal, 1996). Also, coordination of multiple primary actions. For instance, showing another agent how to use a tool, so he can use it too. Or, as crows have done, using a tool to get a tool. It is striking to watch a crow using 3 tools in sequence correctly, for example.[4]

A simple game world seems more than adequate for evaluating the various aspects of animal-like intelligence in the spirit of the Commons and Ross paper cited above. In this spirit we can define 10 tests for evaluating the intelligence of our virtual parrot:

- **Sensorimotor**: A virtual parrot can be rewarded for correctly identifying the "odd" item in a batch. We can test if it appropriately learns this behavior (TEST 1).

---

[2] http://opensiuc.lib.siu.edu/cgi/viewcontent.cgi?article=1511&context=tpr. Oddity learning has also been studied in other animals such as sea lions: http://www.ncbi.nlm.nih.gov/pubmed/16933800

[3] http://www.uky.edu/~zentall/pdfs/10.pdf

[4] http://www.youtube.com/watch?v=41Z6Mvjd9w0

- **Nominal**: Word-object and word-event associations can be taught, e.g. with common objects in the game world such as trees, rocks and walls (TEST 2), and common events such as walking and jumping and eating and sitting (TEST 3).
- **Sentential**: Asking simple questions about entities in the environment on the level that a parrot can understand and respond to ("Where is the green tree?" … "Who is holding the ball?") (TEST 4)
- **Preoperational**:
  - Comparing sizes of sets: in a scenario with two collections of blocks, one can repeatedly reward the parrot for moving toward the collection of blocks which has a larger cardinality. It should then learn to move toward the larger set, so as to get more reward. (TEST 5)
  - Corvid-like tool-using behavior can be easily simulated in a video game world – e.g. needing to use a large stick to get a desired object out of a deep hole; needing to use a small stick to get a large stick out of a shallow hole; etc. (TEST 6)
  - Suppose that at night, red boxes tend to have food in them; but during the day, blue boxes do. The parrot should be able to learn this rule by experience, in order to get food. (TEST 7)
- **Primary**: The parrot can be taught to count via repeatedly telling it number words in the context of collections of objects. We then can test if it has made the abstraction correctly, by showing it collections of objects it has not seen before, and seeing if it assigns each collection a number based on its cardinality. (TEST 8)
- **Concrete**: The execution of sequential tool-using actions, in the manner of corvids, is one example of complex sequencing of actions. (TEST 9) Planning, in which navigation must be sequenced together with tool-using actions, is another. (TEST 10) (OpenCog can already do this in the Unity3D game world to a significant extent, but the capability should be made more robust.)

Cognitive Synergy in an Animal Intelligence Context

Having articulated the types of intelligence evaluation we intend to perform on our OpenCog controlled virtual parrots, we can now explicate more carefully the ways in which cognitive synergy is expected to manifest itself in the context of these specific tasks. The overall cognitive architecture intended for carrying out the proposed experimentation using the OpenCog system is depicted in Diagram 1 just below:
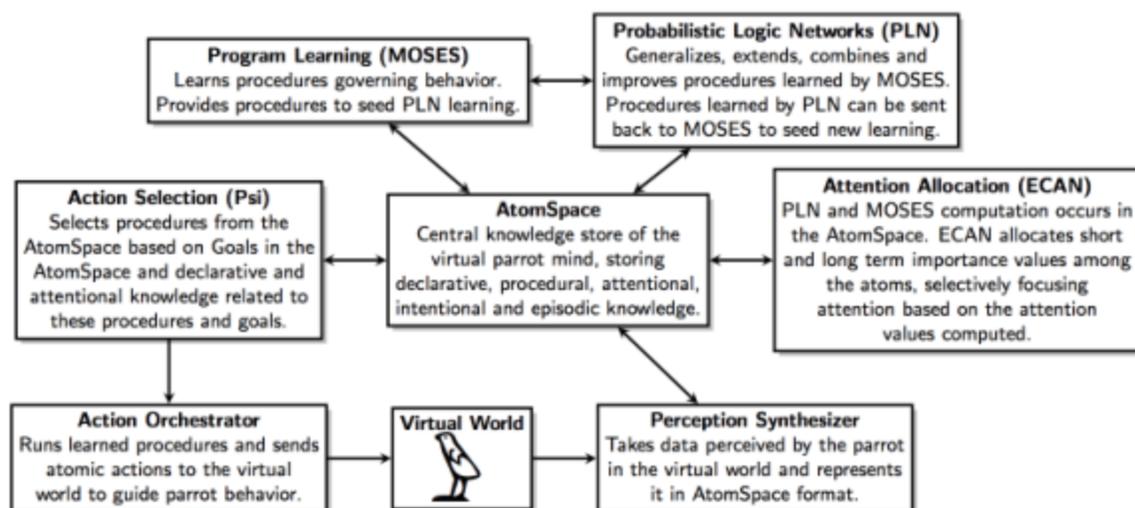
Diagram 1*: High level depiction of the cognitive architecture to be utilized for the proposed experimentation. This is a subset of the larger OpenCog/CogPrime architecture.*

This is a somewhat limited and specialized version of the overall OpenCog architecture, but encapsulates the spirit and key dynamics of the whole in a way that provides for more tractable experimentation in the virtual animal intelligence domain.

The core (cognitive-synergy-based) learning approach underlying this architecture is as follows:

- Use of MOSES, in an unsupervised reinforcement learning mode, to learn small programs that will guide the virtual parrot in achieving its goals (where its goals are managed via the OpenPsi motivational system inside OpenCog, loosely inspired by Joscha Bach's MicroPsi system).
- Use of ECAN to learn associational relationships between entities in the game world, and actions and relationships related to the game world; so that "HebbianLinks" will be formed within OpenCog, between entities that are generally likely to be related in a game-world context
- Use of PLN to generalize the procedures learned by MOSES. For instance, if MOSES has learned a procedure for getting a ball out of a hole using a stick, and has also learned another procedure for getting a block out of a hole using a stick – then PLN may generalize this into a procedure for getting any object out of a hole using a stick. This more general procedure will then be usable for other cases, like using another stick to get a stick out of a hole.
- Use of PLN to learn relationships between actions, contexts and goals in a game-world context
- Use of ECAN to guide PLN inference. This means that PLN, as it does its probabilistic reasoning regarding aspects of the game world and the best strategy for achieving its goals in the game world, will choose which logical reasoning steps to execute based on which options the ECAN "attention allocation" dynamic tells it deserve more attention.

Action selection will be done via OpenPsi according to the Psi model, roughly similarly to how it is done in Joscha Bach's (2012) MicroPsi model. This means that actions will be chosen based on the probabilities with which they are estimated to lead to achievement of the currently high-weighted goals, in the currently perceived/inferred context. These probabilities will be provided primarily via PLN. However, the relationships used by PLN in inferring these probabilities, will be largely generalized from specialized procedures that are learned using reinforcement learning by MOSES. And PLN will be guided in its inference by ECAN. The three learning mechanisms will all work together in supplying OpenPsi with intelligently-dynamically-weighted links to guide its action selection.

Interaction with the virtual world will be mediated by two additional software components:

- An Action Orchestrator, which chooses specific actions for the virtual parrot to execute in the virtual world based on the current running procedures
- A Perception Synthesizer, which takes data from the virtual world and translates into hypergraph format for storage in the OpenCog AtomSpace and utilization by MOSES, PLN, ECAN and OpenPsi
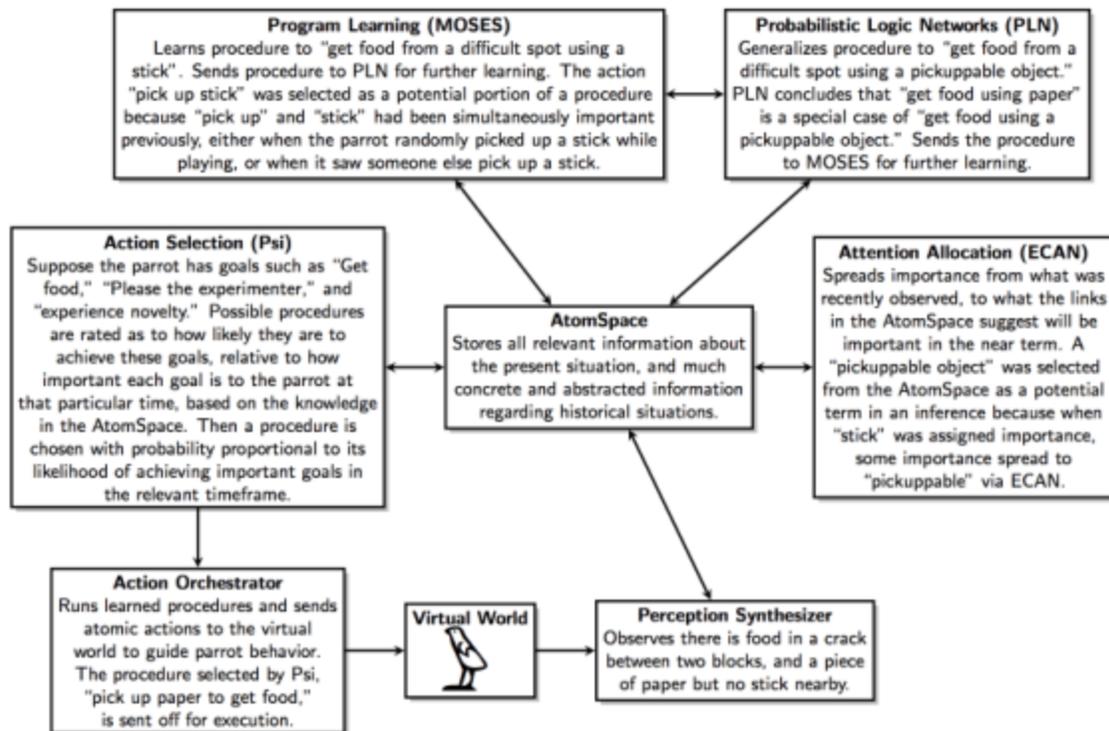
The proposed work will be the first time these three key learning methods will all be used together, and hence will constitute a major milestone from the OpenCog development perspective. It will also be interesting from a pure science perspective, as it will allow us to evaluate a fundamental hypothesis pertaining to the theory underlying CogPrime and OpenCog.

- **Hypothesis**: Utilizing the three learning algorithms together (MOSES, PLN and ECAN) provides significantly better performance on the animal intelligence tests, than using the algorithms individually or pairwise.

Illustrative Example

As a very simple illustrative example of cognitive-synergetic learning in the context of parrot intelligence testing, let us now consider the case of a parrot learning to obtain food from difficult locations via using tools grasped in its beak. This section is slightly more technical than the other ones in this proposal, but we consider it important to give a more concrete flavor of the kind of learning we intend our virtual parrot to do.

A high level depiction of some of the learning that would be expected to occur in this case is given in Diagram 2.

Program Learning (MOSES)
Learns procedure to "get food from a difficult spot using a stick". Sends procedure to PLN for further learning. The action "pick up stick" was selected as a potential portion of a procedure because "pick up" and "stick" had been simultaneously important previously, either when the parrot randomly picked up a stick while playing, or when it saw someone else pick up a stick.

Probabilistic Logic Networks (PLN)
Generalizes procedure to "get food from a difficult spot using a pickuppable object." PLN concludes that "get food using paper" is a special case of "get food using a pickuppable object." Sends the procedure to MOSES for further learning.

Action Selection (Psi)
Suppose the parrot has goals such as "Get food," "Please the experimenter," and "experience novelty." Possible procedures are rated as to how likely they are to achieve these goals, relative to how important each goal is to the parrot at that particular time, based on the knowledge in the AtomSpace. Then a procedure is chosen with probability proportional to its likelihood of achieving important goals in the relevant timeframe.

AtomSpace
Stores all relevant information about the present situation, and much concrete and abstracted information regarding historical situations.

Attention Allocation (ECAN)
Spreads importance from what was recently observed, to what the links in the AtomSpace suggest will be important in the near term. A "pickuppable object" was selected from the AtomSpace as a potential term in an inference because when "stick" was assigned importance, some importance spread to "pickuppable" via ECAN.

Action Orchestrator
Runs learned procedures and sends atomic actions to the virtual world to guide parrot behavior. The procedure selected by Psi, "pick up paper to get food," is sent off for execution.

Virtual World

Perception Synthesizer
Observes there is food in a crack between two blocks, and a piece of paper but no stick nearby.

Diagram 2: This diagram is structured identically to Diagram 1, but each box contains comments on the specific manifestations of the corresponding component of the cognitive architecture in the context of the learning example discussed in this section (learning to use sticks and paper as tools for getting food).

Given a situation in which food is available but not reachable using the beak, and a stick is nearby, MOSES could be expected to learn a procedure such as (to lapse briefly into OpenCog formalism)

```
if
        AND
                Evaluation see $X
                Inheritance $X food
                NOT (Evaluation reach $X)
                Evaluation see $Y
                Inheritance $Y stick
then
        Do PickUp $Y
        Do PushWith $X $Y
```

This is a bit less complex than some of the procedures MOSES has learned in virtual-world experiments in the past (Goertzel, Pennachin and Geisweiller, 2014), such as the virtual dog experiments indicated in Figure 3 above, in which MOSES learned procedures for playing fetch, follow the leader and other common games. Note that the learning of this procedure will be accelerated if the parrot has seen others pick up sticks before, or has picked up sticks himself (while playing, for example). In this case, ECAN will build "shared attention relationships" (HebbianLinks) between the Atoms in the Atomspace representing "stick" and "pick up", and these will bias MOSES toward learning procedures involving the two Atoms together -- an aspect of the cognitive synergy under study here.

Once this procedure has been learned and stored in memory, then when the "get food" goal is reasonably important, and the food is in a difficult spot and a stick is present, OpenPsi can be expected to trigger the stored procedure, so that the parrot will use the available stick to get the food. There is a little subtlety here, because OpenPsi relies on the declarative version of the procedural knowledge MOSES has learned, which looks like

```
Implication
        Sequential AND
                SimultaneousAND
                        Evaluation see $X
                        Inheritance $X food
                        NOT (Evaluation reach $X)
                        Evaluation see $Y
                        Inheritance $Y stick
                SequentialAND
                        Do PickUp $Y
                        Do PushWith $X $Y
        Evaluation MaintainAppropriateFullness
```

(where MaintainAppropriateFullness is the system's version of a "food" goal, meaning it wants to be neither to hungry nor too overstuffed). But the conversion from procedural to declarative knowledge can be carried out in an automated way in this simple situation. In the virtual world context we are abstracting away the difficulties of sensory pattern recognition and motor manipulation, so this becomes a relatively straightforward example.

Next, suppose the parrot is in a situation where there is some food in a difficult place, and there is no stick present, but there is piece of paper present. We don't want the parrot to have to repeat the trial and error it went through with the stick, once again with the paper. Rather, we want it to generalize. PLN can do that: if it knows that

Inheritance stick PickUppable

both hold, it can conclude that

if

> *AND*

>> *Evaluation see $X*

>> *Inheritance $X food*

>> *NOT (Evaluation reach $X)*

>> *Evaluation see $Y*

>> *Inheritance $Y PickUppable*

then

> *Do PickUp $Y*

> *Do PushWith $X $Y*

may potentially be a useful program to execute for getting food. If it then knows that

Inheritance paper PickUppable

this is enough (via one more PLN inference step) for it to figure out to pick up the paper and try to use it to get the food. Again, this conclusion will be more rapidly drawn if the parrot has previous experience seeing paper getting picked up (which ECAN has noted and used in its link-building). Furthermore, if the parrot notes that sticks and paper have many properties in common (as opposed to, say, sticks and rocks or sticks and gerbils), then this will also nudge PLN to draw the relevant inference rapidly.

Once this abstracted procedure is judged useful, it can then be fed back to MOSES, so that MOSES can use it within its ongoing processes. Knowing that paper, and more general pickuppable objects, can also be used to get food from difficult places, may help MOSES to learn additional procedures (say, multi-part procedures involving using one kind of object to get another).

Of course there are many further steps behind the scenes required for OpenCog to carry out this simple example. Even in a simplified context like our virtual animal scenario, the simplest learning behavior ends up involving hundreds of small cognitive steps. Exploring these small steps and how they come together to yield simple instances of synergetic learning is the core purpose of the proposed research.

Plan of Experimentation

To explore the hypothesis articulated above, regarding synergetic learning performance among PLN, MOSES and ECAN, we will compare multiple approaches to the animal intelligence tests implemented:

1. PLN as the sole learning mechanism, guided by simple pruning heuristics
2. PLN with ECAN-based guidance
3. MOSES-learned procedures only, without PLN generalization

4. PLN, guided by simple pruning heuristics, generalizing MOSES-learned procedures
5. PLN, guided by ECAN, generalizing MOSES-learned procedures

Each of these options can be relatively straightforwardly implemented and tested within the OpenCog framework.

Each test mentioned above comes with its own metric.; e.g. the first test *"A virtual parrot can be rewarded for correctly identifying the "odd" item in a batch. We can test if it appropriately learns this behavior"* comes with the metric "percentage of correct responses in identifying the odd item", as measured across a standard test suite. The existence of standard, already validated test metrics is a major advantage of using assessment tasks drawn from the animal intelligence literature.

Hypothesis 1 proposes that Option 5 will outperform the other ones, overall, on the implemented battery of animal intelligence tests.

Another important area for exploration is the comparison of MOSES, PLN and ECAN with other roughly comparable algorithms from the literature (e.g. MOSES vs. Genetic Programming, PLN versus Markov Logic Networks, ECAN versus attractor neural nets). We have chosen not to include this sort of comparison in the core of the proposed work, because it would increase the amount of work required dramatically (since these alternative algorithms exist only as fairly crude research-grade software systems, and tuning any one of these to work in the context of our animal intelligence experiments would be a non-trivial, Masters-thesis sized research project in itself). However, we do plan to place a well-documented, easily-installed version of our Unity3D game world and associated intelligence testing framework online in the OpenCog GitHub open source code repository. This will allow others to use the framework as a testbed for their own AI algorithms, and will allow students to experiment (as coursework or thesis projects)with well-known AI algorithms in this context. This will allow a gradual build-up of comparative results and observations on the 3D game-world animal-intelligence testbed, similar to what we now have on standard machine learning testbeds like UCI (Bache and Lichman, 2013), or computer vision testbeds like CiFAR (Krizhevsky and Hinton, 2009).

Project Plan Overview

The work required to execute the project proposed here falls into four primary categories:

1. Game-world development:
   a. Developing an appropriate game world framework for virtual parrot experimentation, leveraging existing OpenCog game-world technology as appropriate
   b. Implementing the 10 animal intelligence tests mentioned above in the game world, including making any needed improvements to the game world along the way

2. OpenCog AI development, including
   a. Fundamental improvements to the ECAN "attention allocation" component
   b. Development of predicates and concepts required to map game-world observations and actions into Atoms in a manner amenable to tractable PLN inference
   c. Experimentation with PLN inference on Atoms representing procedures learned by MOSES
   d. Tuning of ECAN-based guidance of PLN inference
   e. Tuning of MOSES for learning procedures in the context of the "animal intelligence test" scenarios (which are a bit more difficult than the "virtual dog" scenarios in which MOSES has been used for virtual world learning before)
   f. Customizing the OpenPsi-based motivational system of OpenCog for control of virtual animals

     g. Implementation of an "animal intelligence test harness", a set of software scripts enabling automated running of a suite of game world based intelligence tests
3. Experimentation, data analysis and interpretation

Incremental milestones during the course of the work would be:

Time Point	Milestones
6 months	- Appropriately customized game world developed - Intelligence tests except 6, 9 and 10 implemented in Unity3D game world - MOSES adapted to learn procedures in game world - ECAN code updated to enable scalable real-time attention allocation - Predicates and concepts coded enabling PLN to do inference on game world events - Simple examples of ECAN-guided PLN inference tuned/tested
12 months	- Tests 6, 9 and 10 (involving tool use) implemented in game world - PLN tuned/tested for generalization from MOSES procedures learned in game world - ECAN-guided PLN inference control tuned/tested in animal intelligence tests - Technical report produced, and research paper written
18 months	- ECAN-guided PLN inference tuned/tested for generalization from MOSES procedures learned in game world - Intelligence testing carried out on Tests 1-10, with various options regarding use of MOSES, ECAN and PLN - Analysis of results of intelligence testing - Evaluation of Hypotheses 1 and 2 based on data gathered - Technical report produced, and research paper written

An appropriate team for this work will consist of:

Team Member	Role	Time Dedication	Annual Cost
Dr. Ben Goertzel	Team leader	75%	$75K
Cosmo Harrigan	Lead programmer, OpenCog code expert	75%	$60K
Dr. Nil Geisweiller	MOSES/PLN specialist	50%	$40K
Dr. Joscha Bach	Expert in motivated behavior	20%	--
Dr. Matthew Ikle'	ECAN/ PLN specialist	45%	$40K
Junior AI Programmer: OpenCog Learning	Co-located with Ben, focused on learning	100%	$50K
Junior AI Programmer: Psi Motivation	Co-located with Joscha, focused on Psi/OpenCog structures and dynamics	100%	$50K
Game Programmer / Tester		100%	$45K
TOTAL			$360K

For an 18 month project the total cost would then be $530K.

Assuming the project starts in early 2015, Matthew Ikle' would work full-time on the project for the first 6 months (on sabbatical from his university position), and then 20% for the following 12 months (while back at university).

Time is not billed for Joscha Bach in the above table because it is assumed he is already funded by the Epstein Foundation.

References

- Bach, Joscha (2012). Principles of Synthetic Intelligence. Cambridge U. Press.
- Bache, K., and M. Lichman. "UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science." *See http://archive. ics. uci. edu/ml* (2013)
- DiQuattro, Nicholas and Joy Geng (2011). Contextual Knowledge Configures Attentional Control Networks. Journal of Neuroscience 31(49)
- Goertzel, Ben (2012). CogPrime: An Integrative Architecture for Embodied Artificial General Intelligence, http://wiki.opencog.org/w/CogPrime_Overview
- Goertzel, Ben, Cassio Pennachin and Nil Geisweiller (2013). Engineering General Intelligence, vol. 1 and 2. Atlantis Press.
- Hunt, Gavin R. (January 1996). "Manufacture and use of hook-tools by New Caledonian crows". Nature 379 (6562): 249–251
- Hunt, G.R. and Gray, R.D. (2004). "Direct observations of pandanus-tool manufacture and use by a New Caledonian crow (Corvus moneduloides).
- Karmiloff-Smith, A. (1992). Beyond Modularity: A. Developmental Perspective on Cognitive Science. MIT Press/Bradford
- Krizhevsky, Alex, and Geoffrey Hinton. "Learning multiple layers of features from tiny images." *Computer Science Department, University of Toronto, Tech. Rep* (2009).
- Murphy, R. A.; Mondragon, E.; Murphy, V. A. (2008). "Rule learning by rats" (PDF). Science 319: 1849–1851.
- Pepperberg, Irene (2002). The Alex Studies. Harvard U. Press.
- Savage-Rumbaugh, S., & Lewin, R., (1994). Kanzi: The Ape at the Brink of the Human Mind. Wiley
- Shettleworth, S.J. (2010). Cognition, Evolution and Behavior (2 ed.). Oxford Press, New York.).
- de Waal, F. B. M. (1996) Good natured: The origins of right and wrong in humans and other animals . Cambridge: Harvard
- Washburn, D. A., and Rumbaugh, D. M. (1991). Ordinal judgments of numerical symbols by macaques (Macaca mulatta). Psychological Science 2(3): 190–193