International Journal of Artificial Life Research.

A "Society of Mind" Cognitive Architecture based on the Principles of Artificial Economics

Darryl. N. Davis*
Department of Computer Science,
University of Hull, HULL,
HU6 7RX, United Kingdom
E-mail: D.N.Davis@hull.ac.uk

Vijayakumar Maragal Venkatamuni
Department Of ISE
New Horizon College of Engineering,
Marathhalli Outer Ring Road,
BANGALORE 560087, India
E-mail: mvvijay_76@rediffmail.com

*Corresponding author

Abstract: This research investigates the concept of mind as a control system using the “Society of Agents” metaphor, whereby the whole is described as the collective behaviour of simple and intelligent agents. This powerful concept for mind research benefits from the use of metacognition, and eases the development of a self configurable computational model. A six tiered SMCA (Society of Mind Cognitive Architecture) control model is designed that relies on a society of agents operating using metrics associated with the principles of artificial economics in animal cognition. Qualities such as level of decision making, its cost function and utility behaviour (the microeconomic level), physiological and goal oriented behaviour are investigated. The research builds on current work, and shows the use of affect norms as metacontrol heuristics enables the computational model to adapt and learn in order to optimise its behaviour.

Keywords: artificial economics, cognitive architectures, metacognition, society of mind, learning, affect, norms.

Introduction

A cognitive architecture can be viewed as an embodiment of a scientific hypothesis of (human and nonhuman, both animal and artificial) cognition. Cognitive architectures are designed to be capable of performing certain behaviours and functions based on our understanding of minds (Newell & Simon, 1972; Franklin, 1995; Davis, 2002). Cognitive science has developed in a number of directions including intelligent systems, reasoning, knowledge representation, and robotics. The evaluation of cognitive architectures has always been challenging. Several common concepts and different methodologies have been applied to developing new architectures.

There are many examples of cognitive architectures developed for different purpose through using different concepts available in different disciplines; for example, SOAR (Newell, 1990), ACT-R (Anderson, 1993), CRIBB (Bartsch & Wellman, 1989) (Wahl & Spada, 2000), CHREST (Gobet et al., 2001), EM-ONE (Singh, 2005), CogAff (Sloman, 2002) and CAMAL (Davis, 2002, 2008). Different cognitive architectures and paradigms can be said to be modelling different aspects of cognition, with different aims, using different metaphors, and from different contexts. To develop a better and more sophisticated cognitive architecture, researchers need to understand: (1) the sufficient description of theoretical, design and implementation levels of different architectures and; (2) the missing, common and generalised factors of relevant cognitive architectures.

The developing Society of Mind Cognitive Architecture (SMCA) (Venkatamuni, 2008) extends the Davis (2002; 2008) CAMAL cognitive architecture with extra processing layers using society of mind and metacognition concepts. Intelligent behaviour can be viewed as a combination of more simple behaviours. Imagine a simple reactive agent that can only move towards and collect a resource in the environment. Building an optimal or metacognition agent cannot be done with just a community of such simple agents, as they need to interact or take help from other agents. Hence from the perspective of Minsky (1985), developing a cognitive architecture requires the development of many different types of agents, with different behaviours and capabilities.

More importantly, such a community of agents requires agents that perform at more abstract levels. In order to provide a truly adaptable framework a “Society of Mind” needs a top layer catalyst like metacognition. Metacognition is a relatively new buzz word in cognitive theory (Adkins, 2004). Metacognition is defined as thinking about thinking and can be viewed in two ways:

- Monitoring a group of agents in an intelligent or cognitive or robotic architecture (i.e. self reflection)

- Making changes by adapting effective strategies in that group of agents (i.e. metacontrol).

Agent behaviours can be analyzed using many different metrics; for example, affect valencing, (pseudo-)metabolic activity, competition and social interaction with respect to environment and microeconomics. The application of economics on artificial life to analyse adaptive behaviors provides a coherent framework from a cognitive architecture across many levels and types of processing.

Artificial Minds

Minsky (1985) defines mind as the functioning of the brain. Franklin (1995) defines mind as a mechanism of the brain. Minsky says “minds are just what brains do”. Franklin (1995) argues that the foundation for exploring the mechanisms of mind can be done through the possibilities offered by artificial minds. This gives rise to artificial minds defined as man made systems that exhibit behavioral characteristics of natural minds. However, artificial minds need not be limited to being analogues of natural ones. The possibilities offered by computational techniques, and synthetic designs, are equally valid in defining what constitutes an artificial mind.

Reasons for Studying Artificial minds

Why do we need to study artificial minds? What is the need for studying nonhuman minds such as agents or robots? In “Artificial Minds”, Franklin (1995) gave three important reasons for studying artificial minds.

- Questions related to the nature of intelligence in human and nonhuman natural minds are inherently fascinating. The research on artificial minds may well throw a light on these questions.
- To better understand upcoming man machine mechanisms.
- To build better robots or intelligent machines and to work with them more effectively.

Stillings (1995) also gives some important reasons for simulating human and nonhuman minds in the form of artificial minds.

- Cognitive science theories are complicated and sometimes impossible to understand without simulating and observing in software.
- Comparing people with different capabilities and their cognitive processes via simulation. These different cognitive capabilities are applied in the arts and science to give rise to diverse practical applications.

Principles of Artificial Economics

The behavior of an animal has consequences which depend on situation. An important consequence of behavior is energy expenditure. Energy and other physiological expenditure, commodities such as social interaction, water, weather etc., must be taken into account, because they influence the animal state. According to Thorndike (1911) the behavior of animal intelligence is predictable and follows the uniformity of nature. He states that “any mind will produce the same effect, when it is in the same situation." If dissimilar responses are produced on two occasions, the animal must have changed (McFarland & Bosser, 1993).

The conversion from a life to artificial system can be done in three stages

- Understanding the fundamental properties of the living systems.
- Simulating a basic organism and their entire life cycle.
- And, finally, designing the rules and symbols for governing behaviour by interacting with an environment.

One perspective on mind is to consider it to demonstrate the organisational principles and emergent intelligence associated with artificial life systems. Economic theory can be applied in order to analyse and model adaptive or intelligent behaviours. The energy or resource spent in fulfilling any behaviour, drive or goal is the utility to be maximized; and follows the principles associated with economic concepts such as price (cost) and utility. There are many different bio-environmental metrics that can be used in agent-based computational economics (Tesfatsion & Judd, 2006). Major metrics include metabolic activity, competition and social interaction (Bedau, 2003). However, it can be argued that affect (Rolls, 1999; Savage, 2003) may also be considered to be an economic resource that modelled using these principles (Davis, 2003; 2008).

IF $E < ET_0$ *THEN* $0 \rightarrow Utility_{Hunger}$ (Dead)

IF $E < ET_1$ *THEN* $1 \rightarrow Utility_{Hunger}$

IF $E < ET_2$ *THEN* $(ET_2\text{-}E/ET_2\text{-}ET_1) \rightarrow Utility_{Hunger}$

IF $E > ET_2$ *THEN* $0 \rightarrow Utility_{Hunger}$ (Satiation)

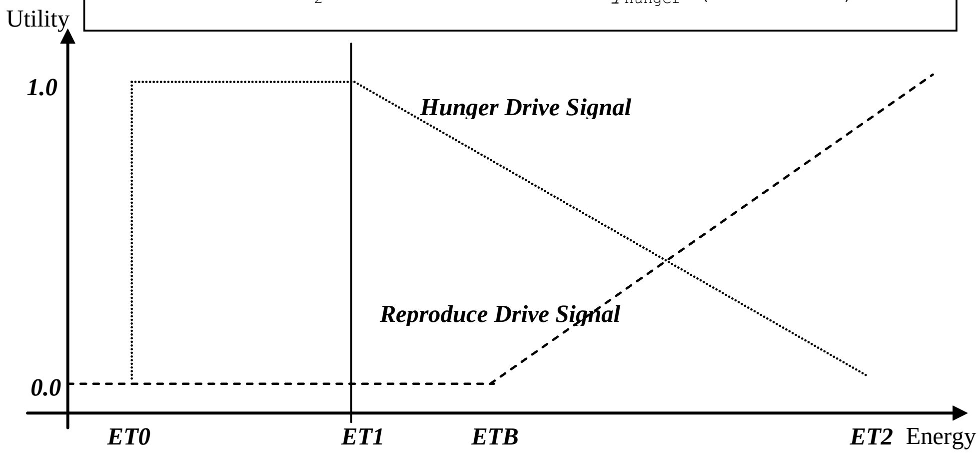


Figure 1. Model for two energy based drives. The four energy thresholds are examples of decision boundaries.

Decision Variables

A decision of a person, animal or robot can be viewed as simply the activity whereby decision variables are compared to decision boundaries. From the economic point of view, the decision-making unit is the cost of performance. Decision-making with respect to use of cost and utility functions depends on given thresholds, decision variables and decision boundaries (McFarland & Bosser, 1993). Figure1 depicts two example drives modeled using this approach from earlier research (Davis, 2003). The decision variable (energy level) and decision boundaries (the four thresholds) give rise to a utility metric. Using this perspective, adaptation is the movement of these boundaries and learning is the generation of new metrics and decision functions.

Cost and Utility Function

The decision making level in animals is defined in terms of cost functions and utility behaviours - the microeconomic level. Cost functions and utility behaviour in animals operate in such a way that a utility (for example, the utility function defined over energy levels in Figure 1) is maximized or minimized (McFarland & Bosser, 1993).

Learning in Animals

Learning is a part of development. It is a result of adaptation to novel, accidental or uncertain circumstance. When an animal learns environmental situations for specific behaviours, it undergoes permanent change. We expect that learning should, in general, bring beneficial results. Much of animal learning (Rolls, 1999) is similar to and can be modeled as reinforcement learning in machine learning or robotics (McFarland & Bosser, 1993; Alonso & Mondragón, 2004).

Optimal Behaviour

Animal behavior is a tradeoff between the native courses of action, i.e. physiological, and goal oriented behavior. An animal engaged with activities optimizes its pattern of behavior with respect to the use of energy and time (McFarland & Bosser, 1993). Adaptation over the kind of utility function shown in Figure 1 would involve establishing the parameters of the utility function, and the establishment of optimal thresholds (decision boundaries) for the decision variables.

Society of Mind

The Society of Mind theorem was initially proposed by Marvin Minsky in the 1970s, with its inspiration dating back to the famous 1960s ‘copy-demo’ project in which Minsky, Papert and their students developed one of the first autonomous hand-eye robots. The demonstration involved the robot constructing children’s building block structures. From this idea Minsky framed the term “Society of Mind”, enabling the view of intelligence as not just a simple recipe or as an algorithm for thinking, but a combined social activity of more specialized cognitive processes. Minsky proposes that the mind consists of a great diversity of mechanisms and is made from simple and smaller entities called micro-agents. Minsky argues that each micro-agent is like a simple piece of code that can perform some tightly constrained task. The agents can be connected within a larger system. Each individual agent, having a different background, plays a different role in society. The society of mind results from combining more specialized cognitive processes (Minsky, 1985), and any societal cognitive architecture contains a large collection of micro agents.

Figure 2 depicts a tree like structure, similar to the tree concept in graph theory (Diestel, 2005). This contains nodes and branches. Each node represents an individual micro agent. Each branch represents a link between nodes. This illustrates Minsky's K-line theorem. The K-lines are data and control lines in the design of a processing architecture. Assume two different cognitive tasks T1 and T2 to be performed in a society of agents. Agents 2, 4 and 5 can perform T1 and, agents 3, 6 and 7 can perform T2 cognitive task. Afterwards T1 and T2 performing agents can be combined as one agency (Agent1). Similarly, any number of agents and agencies can be combined to form a "Society of Mind". For any specific task certain K-lines are triggered on while others remain in an off state. Hence it is possible to build large cognitive architectures as the agglomeration of smaller "complete" systems. The configuration of these various systems as an optimal system for the larger model requires adaptation, learning and meta reasoning. The architecture must learn which configurations are most appropriate and how they can be combined.

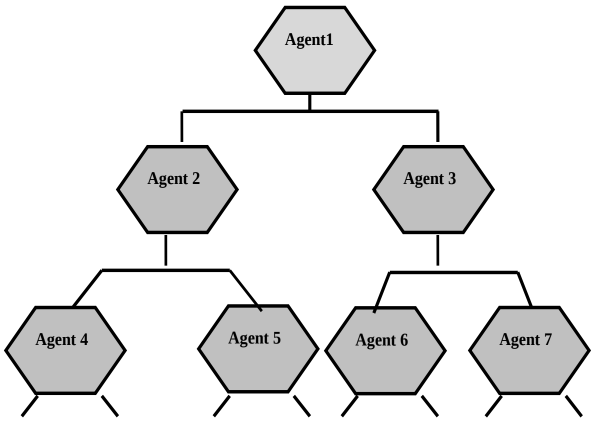


Figure 2. A Society of Agents

Baars (1988) "Global workspace theory" explains mind in terms of conscious and unconscious terms as similar to the on and off state of the "Society of Mind". In the "Society of Mind", the active agents are in the "on" state, and non-active agents are in the "off" state. The combined activity of active agents constitutes a "total state" of mind, and the subset of the activities represents a "partial state of mind." Minsky claims that K-lines arise from the selecting of the most commonly used agents in the Society of Mind theory. These K-lines turn agents on, and are interconnected to each other. K-lines can cause cascade effects. Many K-lines problems and their solutions are stored in a 'chunking' mechanism. If the system faces the same type of problem again, then it uses a previous solution to resolve it. At some level in this architecture, there must be agents capable of selecting active subsets; such agents can be viewed as acting as meta-cognitive agents.

Metacognition

The study of metacognition has grown since the 1970s. In educational psychology, Flavel (1979) and others developed a model of children's cognition about a memory (metamemory), understanding (meta-comphrension) and communication (meta-communication). Metacognition is often simply defined as "thinking about thinking" (Cox, 2005). Broadly defined, metacognition is any knowledge or cognitive process that refers to the monitoring and controlling of any other aspect of cognition. According to Adkins (2004) metacognition is thinking about knowing, learning about thinking, control of learning, knowing about knowing, and thinking about thinking.

Minsky (1985) states that we cannot think about thinking, without thinking about thinking about something. Where that something is a behavior or activity, the metacognitive act can be referred to as metacontrol. According to Flavel; there are three stages in metacognition: (1) metacognitive knowledge; (2) metacognitive experience; and (3) metacognitive regulation. Metacognitive knowledge can be viewed as a database of knowing about an environment, the nature of the task, and strategies used for determining facts. Metacognitive experience is, after processing a given task, forming knowledge on how to achieve results for that task. Controlling and (the self reflective) monitoring of progress using cognitive tasks is termed metacognitive regulation (Adkins, 2004; Cox, 2005; Flavell, 1979).

Metacognitive aids or metacomponents are used for the representation of thoughts (Adkins, 2004) are made with the help of some aids such as:(1) using an abstraction or metasyntactic variable (related to decision variables); and (2) goal setting variables such as perceptual range, parameters for affect, drive thresholds etc.. Norms and higher level (meta-) cognitive rules are examples of metacomponents (Cox, 2005). The term “norm” is an interdisciplinary term, and can be used to refer to a standard principle or a model used for the correct set of actions for specific scenarios (Yoshida, 2007). The executive processes that controls the other cognitive components are responsible for establishing the cognitive framework for specific tasks and problems, and then regulating the execution (and success) of those processes.

Design of SMCA

Artificial mind can be viewed as a control structure for an autonomous software agent. Pioneers, such as Selfridge (1959), McCarthy (1959), Newell and Simon (1972), Minsky (1985), Baars (1988), Anderson (1993), Franklin (1997), Sloman (2002), Nason and Laird, (2004), have all viewed computational theories of mind from the perspective of artificial agents (albeit not necessarily phrased using the term agent). Any cognitive (or indeed computational) architecture can be viewed as either a single agent or a large collection of agents. There is a long history of representing mind as collection of agents (or demons), dating back to Selfridges’s Pandemonium model (Selfridge, 1959). The current model attempts to explain aspects of mind as a collection of agents.

The architecture being developed here (Davis, 2002; Davis, 2003; Davis 2008; Venkatamuni, 2008) can be viewed from a number of perspectives. One leads to developing many different types of simple agents, with different behaviours. These agents are distributed across different layers of architecture, so as to cover all processing and functioning associated with the adopted model of mind.

The SMCA architecture has been designed and implemented as six layers; i.e. reflexive, reactive, deliberative (including BDI models), learning (Q-learner), metacontrol and metacognition; and three major columns: perception, intention, and reasoning; the latter is partitioned into affect, cognition and motivation. This approach leads to the development of many different agent behaviours. For example the presently working architecture has six reflexive behaviours, eight reactive behaviours, fifteen deliberative behaviours, nineteen perceptual behaviours, fifteen learning behaviours, thirty metacontrol behaviours and seventy seven metacognition behaviours. Each behaviour is modelled as an agent. Indeed, from an extreme perspective on distributed model of mind as control system, there may exist any number of, sometimes randomly created, reflexive, reactive, BDI (Belief, Desire, and Intention) agents, deliberative, perceptual, learner (Q learning), metacontrol, and metacognitive agents. The designs of the individual agents are discussed in the next sections.

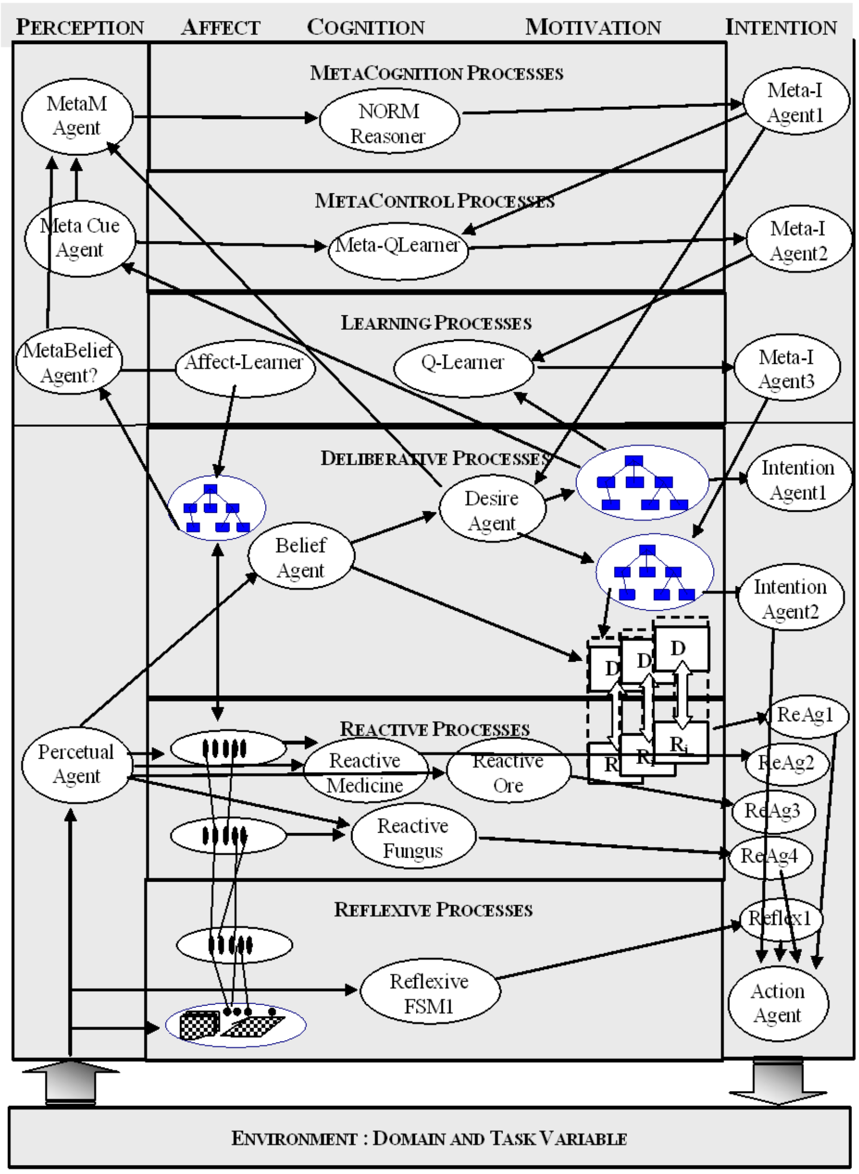


Figure 3. Architectural Design of **SMCA**

The fungus world testbed (Toda, 1962) used in this experiment involves cognitive and engineering perspectives. The fungus world environment has been created to contain dynamic and static entities. Static blocks are used to create particular variations of the environment. Dynamic (and time changing) entities include the presence and amount of standard fungus, small fungus, bad fungus, ore, golden ore, crystal and medicine. These entities affect agents through task orientation or physiological change. The agent's internal states relate to its energy level, and its biochemical engine. The agent uses energy in navigating the environment, and becomes inert if that level reaches zero. There are decision boundaries (which the agent can modify) to determine whether it is energy-hungry or likely to suffer an energy shortfall in performing a task. The agent replenishes its energy from gathering fungus. The agent does not know until gathered whether the fungus is the standard fungus, a small outcrop (with less energy affordance) or bad fungus. Bad fungus increases the biochemical engine level. The biochemical engine (or metabolism) governs how much energy the agent consumes per processing cycle and per environmental activity. The agent (typically) prefers to be in state of low metabolism, but this like other decision boundaries can be set by the agent's metacontrol processes.. Bad fungus increases the metabolism from low to medium to high, while medicine (also to be found in the environment) decreases the metabolism. There are a variety of testbed parameters for the testing of an agent's biochemical engine and task performance. A number of types of agents (random, reflexive, reactive, learner and BDI-models, metacontrol and metacognition) are introduced in this experiment (Venkatamuni, 2008).

Reflexive Level

Reflexive behaviours, at their simplest, are ballistic mappings from input (i.e. perception) to output (i.e. behavioural response such as an action on an environment). As such, reflexive agents fit to the first layer of the SMCA shown in Figure 3. Reflexive agents are designed to embody given environmental rules. Reflexive agents sense their immediate environment and understand environment sensors, such as the locations of edges or territorial centre points, etc. For each move, they check the corresponding adjacent positions and determine the direction of their next move. The tasks of such agents are to move in the environment and avoid collisions with other agents and objects in the environment. Given that there exist a range of possible such agents, the current reflexive action for any one combined agents is specified by deliberative agents.

7.2 Reactive level

There can be various different definitions of what constitutes a reactive component (Braitenberg, 1984; Brooks, 1999). For example the system may have no changeable internal state so that the current input determines the current output; in this case the output is always the same given the same input (we term these reflexive as described above). The definition of a reactive system taken here is that the systems output is determined not only by its input, but also by its internal state.

Reactive agents compromise the second layer of the architecture shown in Figure 3. Reactive agents are designed to perform goal oriented behaviour, building on the mechanisms associated with the reflexive agents. The goals they attempt to satisfy as specified by deliberative BDI agents (described in the next section).

Algorithm1. Example Reactive Algorithm

Goal based behaviour towards task resource

Goal: one of {ore, golden ore and crystal} (as set by a deliberative agent);

Find the nearest resource by their distance;

Select the direction towards nearest resource;

Move towards resource;

If no resource within the perceptual range activate reflexive actions.

The design for the fungus testbed includes seven different reactive agents. Algorithm1 summarises how task resource (gold, ore, crystal) reactive agents function. Similar algorithms were written for physiological engine resources (fungus and medicine). The deliberative BDI determines which of the reactive control mechanisms are active according to the goals the entire architecture attempts to satisfy.

Deliberative Level

Deliberative refers to any system whose output is not only determined by its input and current state, but also by its previous states and/or the current/previous states of other systems. In other words a deliberative system is one whose output is based upon an extended memory beyond that of its own current state.

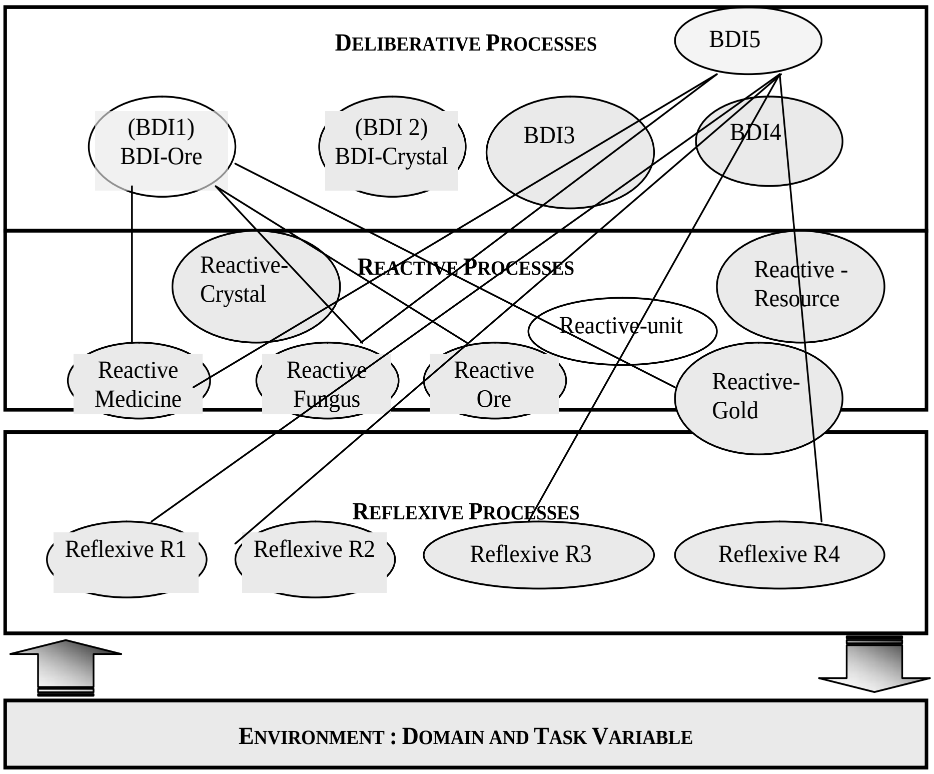


Figure 4. BDI and their control axes (K-lines)

Deliberative agents compromise the third layer of the distributed cognitive architecture shown in Figure 3. The design of deliberation mechanisms for the fungus testbed includes five different types of BDI (Belief-Desire-Intention) agents. The BDI agents determine which of the reactive or reflexive control mechanisms are active according to the goals the entire architecture attempts to satisfy. Animal based micro (deliberative) agents in a fungus world testbed are capable of performing different tasks related to principles of artificial economics. Each

BDI (Belief- Desire-Intention) model has a different group of coordinated capabilities to meet a particular intention. These goals are either task related or agent-internal resource related. Each BDI agent follows the reactive actions in each move based on given rules and decision variables. For example Figure 4 depicts BDI-Ore (BDI1) that selects and controls the combination of reactive-fungus, reactive-ore, and reactive-golden-ore and reactive-medicine behaviours; BDI5 agent selects and controls the combination set of reactive-fungus, reactive-medicine and reflexive behaviours.

In the example given in Figure 4, the BDI agents determine which of the reactive or reflexive control mechanisms are active according to the goals the agent attempts to satisfy. These goals are relate to either task resources or the agent's internal condition. The range of activities required to fulfill these deliberative goals determines the type and number of reflexive and reactive agent required for any specific testbed.

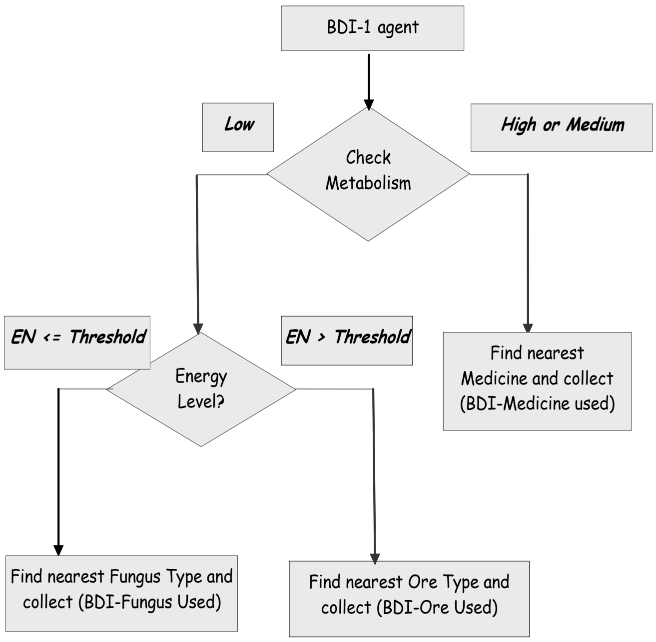


Figure 5. Decision variable design of BDI-1 Micro-agent.

Consider a scenario of an energy-hungry agent in the fungus world testbed. The agent intends to collect ore. If the agent is in a hunger state (i.e. its energy level is less than a given decision variable or the predicted energy use for some task will reduce the energy value to such a level) or in a high metabolic condition, then the agent changes its desire towards fungus or medicine (medicine reduces metabolic rate). Based on the agents needs and cost function, different deliberative agents can be framed. The difference between each BDI model is expressed in terms of decision variables, cost functions and economic utilities framed in terms of energy level, biochemical engine and resource goal. The reasoning of decision variables for one agent (BDI-1) is shown in figure 5. BDI-1 is equivalent to Agent1 in Figure2' sitting at the root of a society of agents that perform similar but different object specific tasks. Dependent upon two decision variables (metabolism and energy levels), one of three (one level down) BDI agents are activated. Many of the low level activities and behaviours (such as navigate, move to resource, collect resource) are similar and independent of whether the cognitive agent requires fungus, medicine or can collect ore.

The decision path of BDI-1 ensures that the correct BDI resource agent is activated, and that, in turn, ensures the appropriate reactive agents used in the simulation environment.

Figures 4 and 5 illustrate how the society of agents perform as group with K-lines being activated (or deactivated) according to decisions made over decision variables related to task goals, and agent internal resources. In short, if the energy level is sufficient, based on cost and utility function (microeconomic level), then the deliberative architecture activates the appropriate reactive agent to move towards its specified task goal. Similarly, the other BDI K-lines activate searches for the nearest medicine to collect to decrease the global metabolism, or find energy replenishing fungus. The deliberative agents effectively follow different goals based on cost and utility functions.

Learning Level

The fourth layer of the architecture encompasses the learning processes. Although these learning processes could be incorporated into the other layers, the incorporation of an explicit learning layer eases design and implementation, as suggested in the EM-One research (Singh, 2005). Learning changes decision making at one level about actions at another level for tasks defined at yet a further level. This layer is in effect controlled through connections to the metacontrol level.

The current implementation makes use reinforcement learning (Sutton & Barto, 1998). This learning attempts to improve the effectiveness of the planning, and action selection paradigm based on maximizing reward. Q-learning algorithms work by estimating the values of state-action pairs. The value $Q(s, a)$ (refer Algorithm 2) is defined to be the expected discounted sum of future payoffs obtained by taking action a from state s and following an optimal policy (i.e. delta value to find Q values) from the current state s, on selecting an action a. This will cause receipt of an immediate utility unit a, updating the state as: $s' \rightarrow s$.

Algorithem2. (Micro agent) Q-Learning Algorithm

Let $\mathbf{Q}(s, a)$ be the expected discount of reinforcement of taking action a in state s, then continue by choosing actions optimally.

- Initialize a table $\mathbf{f}$ with states $\mathbf{S}$, actions $\mathbf{A}$ and the $\mathbf{Q}$ (utility or reward) value estimates.
- Select an action a (where $a \in \mathbf{A}$) and execute it.
- Observe the immediate reward r. Reward is defined using some agent relation, for example distance to desired object. Observe the new state s', achieved by action a on state s, where $a \in \mathbf{A}$ and $s \in \mathbf{S}$.
- Update the table entry for $\mathbf{Q}$ value using an appropriate rule, for example, New $\mathbf{Q}(s, a) = \text{Old } \mathbf{Q}(s, a) + (r(s) - r(s'))/ r(s)$. The **Q** values converge to their optimal values

Metacontrol level

Metacontrol agents decide which deliberative agents are to be learnt and ready to perform in different conditions. These meta-deliberative actions are architecture control actions. A metacontroller determines the relevant control actions, to optimise the society of agents that compromise the cognitive architecture Metacontrol agents comprise the fifth layer of SMCA as shown in Figure 3.

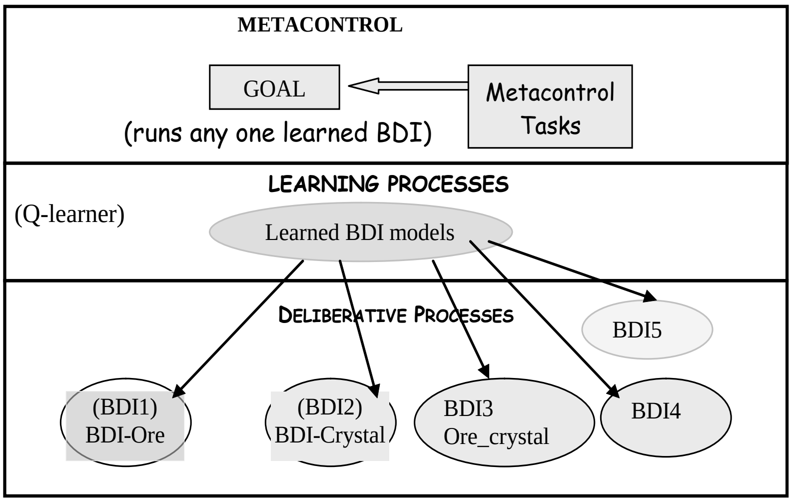


Figure 7. Metacontrol Task

The metacontrol agent learns actions upon the environment guided by the current set of goals and needs. The agent calculates all the combinations of deliberative agent's states (inputs) and actions. Metacontrol agents can affect different levels of skills, such as reflexive, reactive, deliberative, and learning capabilities. As Figure 7 depicts, metacontrol agent can select and controls any of one of the decision models such as: (1) learned-BDI-ore, (2) learn-BDI-crystal, (3) learned-BDI-ore and crystal, (4) learned-BDI-adaptive and (5) learned-BDI-reflexives. Metacontrol agents can select and produce BDI models; but cannot reason as if a BDI agent. Due to this reason metacontrol agent can not reason and change existing BDI models; only extend them with new versions.

Metacognition Level

This is the final layer of the SMCA mind model. This layer uses Norms to configure the architecture. Norms are rules that a group uses for appropriate (and inappropriate) values, beliefs, attitudes and behaviours (Kahneman & Miller, 1986). The metacognition level agents work at an architectural level, and can change the framework of currently active BDI agents.

This level works to control and monitor the deliberative model, based on Norms (refer Figure 8 and Algorithm 3). Different combinations of agents are organized for different goals and tasks through the use of K-lines. These K-lines turn agents on (or off) at the top level of Figure 4 and can cause cascade effects within SMCA.

Metacognition in SMCA works via the processes of Observation, Metacomphrehension, Metamanagement, and Schema training. Metacomponents, such as Norms, help in the switching of BDI agents between on and off states; and detail the (deliberative) actors, for each goal-agent-environment situation. Norms are adjusted, based on feedback from environmental and agent situation (the self-reflection process), and enables the architecture to compare, review and change its configuration for specific goals. Metacomphrension is the remedial action of selecting the most appropriate Belief-Desire-Intention combination (BDI-Ore, BDI-Crystal, BDI-Crystal, etc.), through the analysis of past architectural configuration results. The sequence of events and processes over the metacognition, metacontrol and learning levels in sketched in Figure 8, and described in the following section.

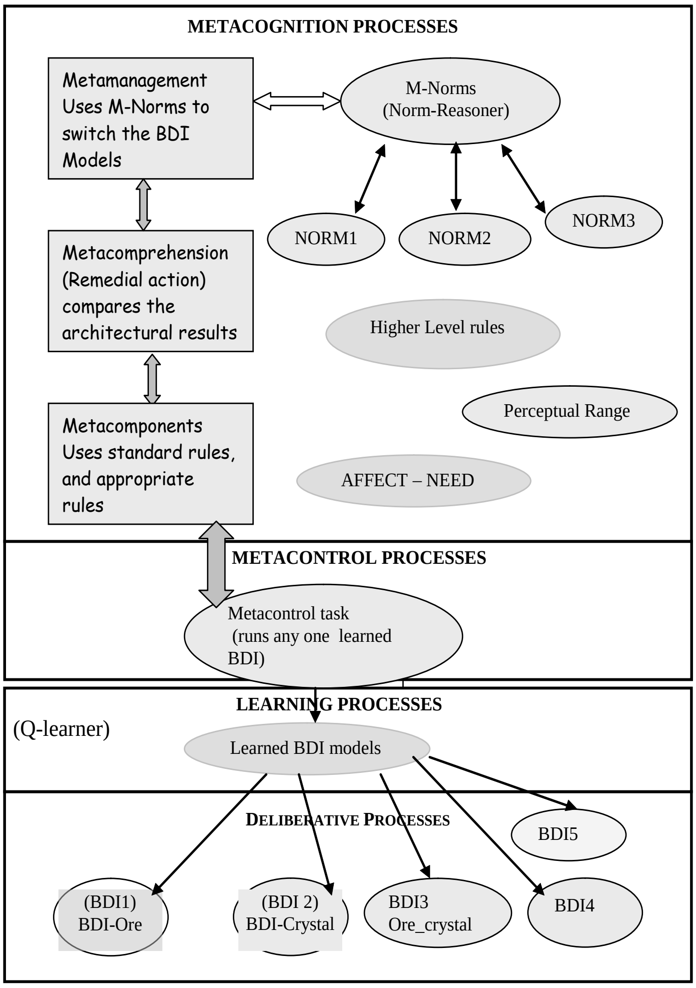


Figure 8. Metacognition agent design

Processing Cycle

In general terms, intelligent agents operate in a perception-action cycle. Mind is therefore often split into three separate components, sensory systems, central processing systems and motor systems. Overall the SMCA architecture runs in such a Perceive-Decide-Act,; however not all processing capabilities of the architecture are used on every processing cycle. A typical sequence of events in SMCA compromise of the following steps:

- Use Perceptual Agents (at current perceptual level) to sense environment. If this results in *freespace* descriptors (i.e. empty space in area sensed), then metacontrol called to set perceptual range or perceptual level for sensing of task or resource related entities in an environment. For example, if current level 3 Perceptual Range return Sense List = [3-spacefree, 2- spacefree, 1-spacefree], increase to level 5 so Sense List = [5- fungus, 4-agent, 3-spacefree, 2- spacefree, 1-spacefree]. The agent only senses sufficient environment to satisfy its current needs and tasks.
- Updates Belief Set with Perceptions at Deliberative level.

- Use Affect mechanism to establish metabolic needs and the need of physiological resources; and then map internal states onto Belief Set.
- Call meta-control in order for Metacomprehension to select appropriate Belief-Desire-Intention combinations (e.g. BDI-Ore, BDI-Crystal, BDI-Crystal, etc.), by analysing past architectural results. This uses Metacomponents such as Norms or M-Norms to decide which BDI model to make optimal by comparing resources available and balance the resources in a testbed to the goals and needs of the architecture
- Trigger the BDI model to cause actions at the reactive and reflexive level and so behaviourally interact with the environment.

If tasks and goals are not being achieved effectively (as decided by the Metacomprehension agents) then the Metamanagement agents are called to use M-Norms to switch the BDI Models (such as BDI-Ore, BDI-Crystal, BDI-Crystal, etc.). If these also prove inadequate or if no further BDI models are available, the schema training components are activated. Schema training uses Q-Learning for finding the optimal steps to be taken for the resolving of goals by using M-Norms and the Affect Mechanism at the metacognition level. Schema training results in the opportunistic establishment of new BDI-models for specific tasks in certain environmental situations.

Experiments with SMCA

The fungus world testbed is implemented using Prolog and developed using cognitive and engineering perspectives. The fungus world environment has been created to have both dynamic and static features. The static feature can be specified to create a particular configuration of the environment. Dynamic features include the generation and use of fungus, medicine and goal resources (ore and cruystal). There are different parameters in the environment for an agent's biochemical engine and performance (Table1*)*.

Resource parameters in the environment consist of: (1) standard fungus; (2) small fungus; (3) bad fungus; (4) ore; (5) golden ore; (6) crystal and (7) medicine. Fungus is a nutrient for the agents. Each standard fungus gives an agent 10 energy units. Initially, each agent has a predetermined energy level. For each cycle, the agent consumes a fixed number of energy units. If the energy level (nutrients) reaches 0, the agent will die. The small fungus gives an agent 5 energy units. If the agent consumes a small fungus, 5 energy units (default) are added to the energy storage. The bad fungus has 0 energy units. If the agent consumes bad fungus, it gets null energy. Moreover, bad fungus increases the metabolism rate, and changes the metabolism of the agent in the testbed. The collection of medicine decreases the metabolism. The metabolic effect is exactly opposite that of the collection of bad fungus

The collecting of ore is the ultimate goal of each agent. Each agent group tries to collect as much ore as possible in the environment. At the same time, an agent has to maintain the energy level necessary to live in the environment. Initially, collection is 0, and one value is added after collecting each piece of ore. Collection of golden ore increases the performance of an agent. One piece of golden ore is equal to five standard ore units. Collection of crystal increases the performance of agent by a factor that is double that of standard ore.

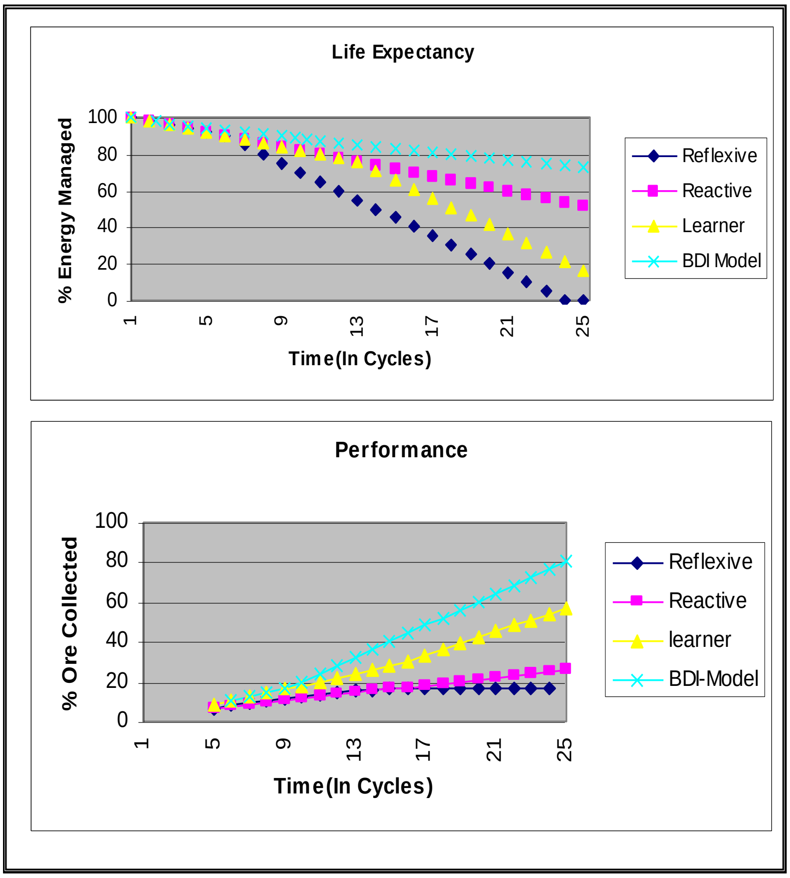
Parameter	Type	Value	Default Effect
Fungus	Object : Numeric	10	Increases the energy level by 10 energy units, to live in the environment
Small Fungus	Object : Numeric	5	Increases the energy level by 5 energy units, to live in the environment
Bad Fungus	Object: Numeric	0	Increases the energy level by 0 energy units, to live in the environment Decreases the performance by Increasing metabolism
Ore	Numeric	1	Increases the Performance by 1.
Golden Ore	Numeric	5	Golden Ore increases the agent performance 5 times More than an ore.
Crystal	Numeric	2	Crystal Increases the agent Performance 2 Times more than a Ore.
Medicine	Object: Numeric	0	Increases the performance by Decreasing metabolism
ENP (Energy storage)	Object: Numeric	N/A	Stores the energy based on consumption of Fungus, Small Fungus, and Bad Fungus.
Cycle	Object: categorical	1 or 2 or 5 Energy units	Agent consumes the Energy
Low Metabolism	Categorical-atom	1	Agents use energy at 1 unit per cycle
Medium Metabolism	Categorical-atom	2	Agents use energy at 2 unit per cycle
High Metabolism	Categorical-atom	5	Agents use energy at 5 unit per cycle

Table 1. Parameters for fungus world environment.

The environment supports the running of the various types of agents within the SMCA architecture, where each agent uses a different type of rules and mechanisms. In these experiments, a maximum of 50 agents were defined. The experiments were conducted with the same parameters for the presence of fungi (including standard, small, and bad), ore (including standard and golden ore) and the same objects (including obstacles). The time scale and maximum cycles were kept constant by adding the same type of agent in each experiment. To compare the results for each agent, the following statistics were collected: life expectancy, fungus consumption (including standard fungus, small fungus and bad fungus), ore (standard ore and golden ore), crystal collected and metabolism. The life expectancy or age of the agent is noted, along with the agent's death (or age after the end of the maximum cycles or time). The agent's total performance will be calculated by amount of resources (ore, golden ore and crystal) collected, and based on life expectancy. The simulation experiments were executed several times with the same initial settings, and the result graphs were generated by taking an average of ten simulation experiments.

Study One (Systematic comparison of multiple agents)

This experiment compared the performance of agents from four levels in the SMCA architecture. Each agent was selected from the population of agents at that level as the best (from that layer) to collect Ore. The life expectancy and ore collection for reflexive, reactive, learner and BDI agents were compared.



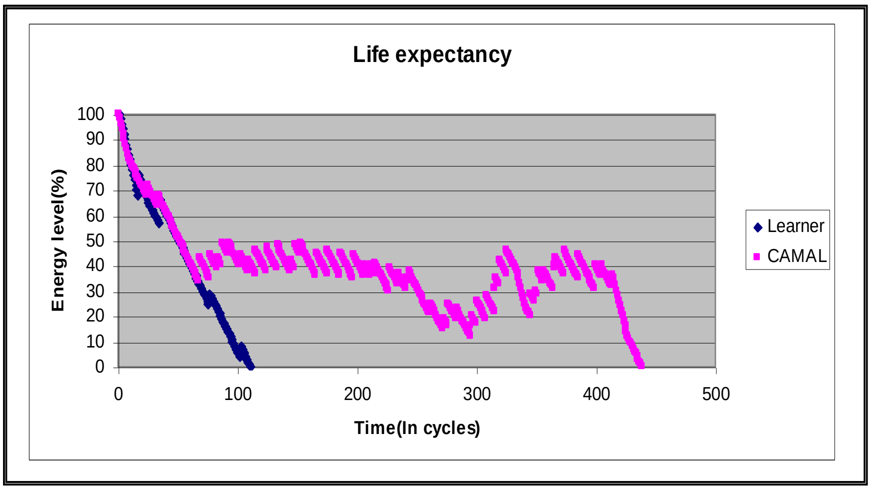
Graph 1. Performance of SMCA lower level to higher level agents

The results of this experiment (Graph 1) shows that BDI model agent maintains a higher level of life expectancy than other simpler agents. Reflexive agents collected 16% of ore, while reactive agents collected 26%, simple-learning agents collected 57% and BDI agents managed to collect 80%. The BDI agents maintained a higher level of life expectancy at72.5%, than reflexive (26%), reactive (36.5%) and learning agents (41%), over the repeated experiment.

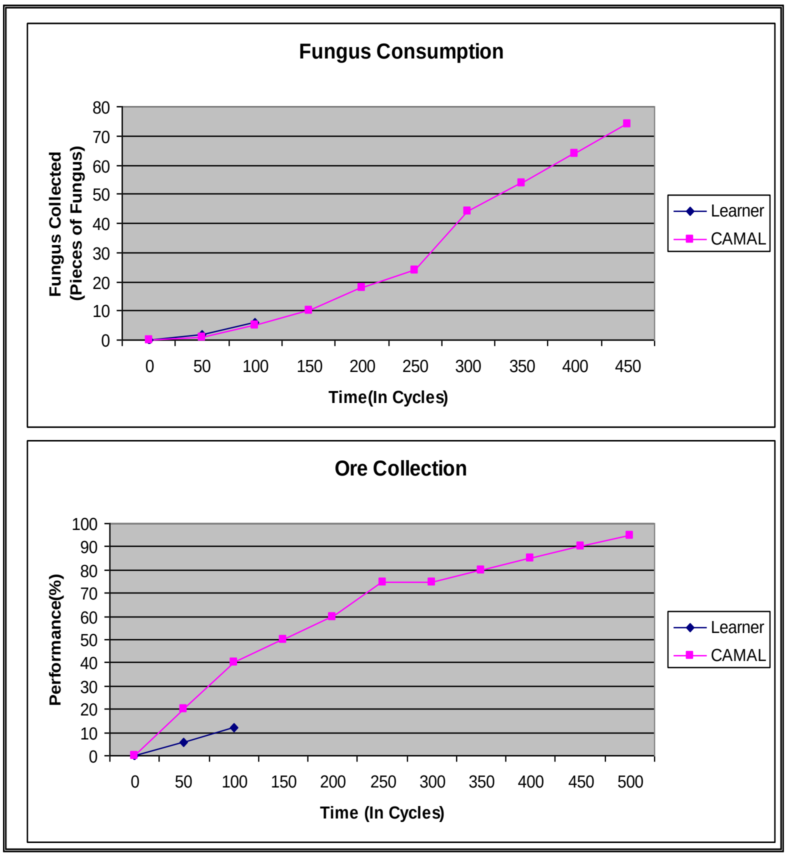
Case2: Learning and Control

This experiment compared a BDI agent with a complete control mechanism (CAMAL agent) with a reflexive learner. The CAMAL BDI agents were designed using the principles of artificial economics applied on deliberative mechanisms such as cost function, optimal decision making and decision boundary variables. The learner agent combined the same reflexive behaviours (as used in the BDI agent) with the learning layer.

As shown in Graph 2, the BDI agent manages to live up to 438 life cycles. The cognition (reflexive-learner) agent only managed to live up to 110 life cycles in the fungus world environment. The artificial economics based micro agent demonstrates a control mechanism capable managing an energy level of the assigned threshold using its decision variables; and trying to manage the same line for the maximum time of its life cycle. The agents exhibit optimal decision making capabilities near the decision boundary.



Graph 2. Life Expectancy of Two Micro-agents.

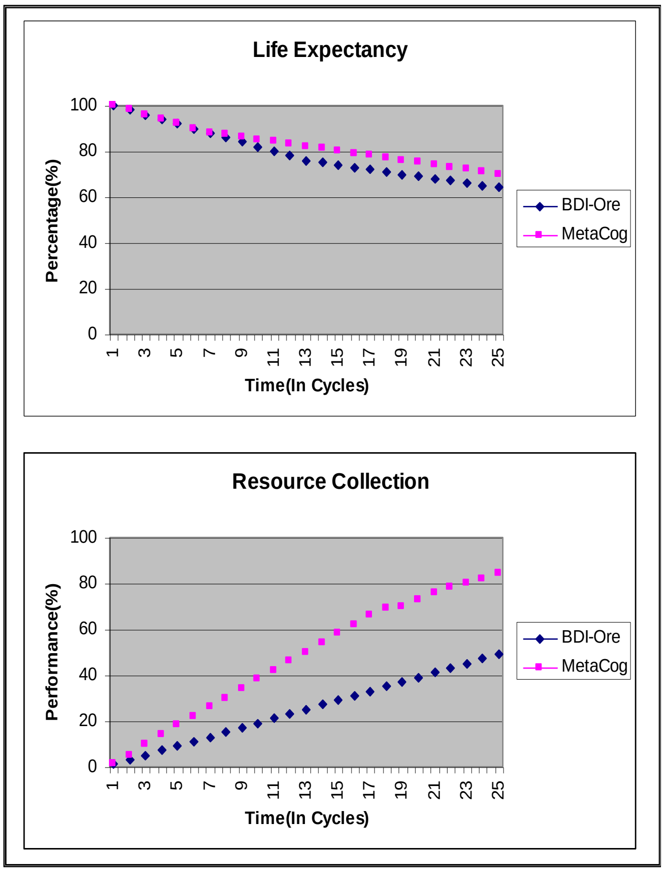


Graph 3. Fungus and Ore Collection of Two Micro agents.

Resource (ore, golden ore and crystal) collection (Graph 3) shows the cognition agents managed to collect 12 pieces of ore, while the CAMAL BDI micro agents managed to collect 95 pieces of ore. The life expectancy and energy management of Graph 2 impacts here, as if an agent acquires more than the threshold or predicted energy level, then the agent tries to collect ore. If the agent has a lack of energy, then it collects fungus (hunger condition). The cognition (reflexive-learner) agent managed to collect 6 pieces of fungus and the BDI micro agent managed to collect 74 pieces of fungus. As Graph 3a illustrates, the (reflexive-learner) cognition agent was initially found to collect more fungus than the BDI micro agent. The BDI micro agent was not so concerned about fungus at this stage. As the Graph illustrates, in between the 250 and 300th life cycle, the BDI micro agent's fungus consumption rate is found be very high. At this stage, the agent is in the hunger condition, and needs more fungus. This follows dip in the fitness or life expectancy. Hence it switches towards the collection of fungus, and the ore collection (in the lower) graph flattens. This result show that BDI micro agents are better at self regulation (or self control), and the achievement of their goals than a straight forward learner agent.

Case 3: The Effect of Metacognition

This experiment provides a comparison between the BDI from the previous experiment and metacognition agents. It aims to show how the metacognition is used on BDI models to improve performance through unification. The experiment also demonstrates how the cognition and metacognition agents demonstrate the "Society of Mind" concept. A range of experiments were conducted (Venkatamuni, 2008), but here we show just the life expectancy and resource collection results to allow comparison with the earlier experiments.



Graph 4. Performance of One BDI and Society of Agents with Metacognition.

As shown in the Graph 4, while the BDI-model agent collects 50% resource, the metacognition agent (i.e. full SMCA) collects 82% of resource. The energy level of each type of agent is noted after the maximum cycles. BDI agents are left with 64% energy, while metacognition agents are left with 71%. This shows that the computational expense associated with running the meta levels of the SMCA architecture results in more task effective, and resource efficient models of behaviour.

Conclusion

This research paper illustrates the design and implementation of a six tiered Society of Mind Cognitive Architecture (SMCA). This architecture combines reflexive, reactive, deliberative, learning, metacontrol and metacognition processes based on principles from artificial life and cognitive science. This research demonstrates how a large collection of agents can act as a single agent manifest optimal decision making and intelligent behaviours. The control model is designed to operate using metrics associated with the principles of artificial economics. Qualities such as level of decision making, its cost function and utility behaviour (the microeconomic level), physiological and goal oriented behaviours are combined across the architecture using an affect mechanism described elsewhere (Davis, 2008). The micro agents combine to create a complete control mechanism for managing pseudo-physiological internal states such as energy level and metabolism. Decision making capabilities over decision variable boundaries enable these micro agents to engage in activities to utilize their pattern of behaviour with respect to the use of energy and time. The complete architecture outperforms its constituent (intelligent subparts) in management of the two incompatible activities of energy replenishment (fungus consumption) and goal-oriented behaviour (e.g. the collection of ore). The utilisation of metacognition provides a powerful concept for mind research that can be used in a society of agents for control, unification and self-reflection. Finally, this research gives a clear roadmap for researches, to develop a society of mind approach to cognition using metacognition concepts (including norms) that are to be found in the wider contexts of the cognitive sciences.

Further research is required to develop the perspectives outlined in this paper. The metacognition concept is being applied to the control of cognitive robots (Davis, 2008); and can therefore be tested in a physical (‘real”) world rather than the simulations of this and other testbeds. There are large extensions and directions that can be made to SMCA. Presently SMCA is implemented using a society of around fifteen agents, and one seventy seven behaviours. This can be extended with more complex skills and more challenging testbeds.

REFERENCES

Adkins, J. (2004). *Metacognition*, University of Saskatchewan, Canada.

Alonso, E. & Mondragó, E. (2004). Agency, Learning and Animal-Based Reinforcement Learning, In: *Agents and Computational Autonomy Potential, Risks, and Solutions*, Lecture Notes in Computer Science Springer Berlin/Heidelberg.

Anderson, J. (1993). *Rules of the Mind*, Lawrence Erlbaum Associates, Hillsdale, NJ.

Baars, B J. (1988). *A Cognitive Theory of Consciousness*. New York: Cambridge University Press.

Bartsch, K. & Wellman, H. (1989). Young Children's Attribution of Action to Beliefs and Desires. *Child Development*, 60.946-964.

Bedau M A. (2003). Artificial life: Organization, Adaptation, and Complexity from the Bottom Up, *Trends in Cognitive Science*, 7(11):505-512.

Braitenberg, V. (1984). *Vehicles, Experiments in Synthetic Psychology*. MIT Press.

Brooks , R. A. (1999). *Cambrian Intelligence: The Early History of the New AI*. Cambridge, Mass. MIT

Press.

Cox, M T. (2005). Metacognition in Computation: A selected research review, *Artificial Intelligence*. 169(2):104-141.

Davis, D N. (2002). Computational Architectures for Intelligence and Motivation, *17th IEEE International Symposium on Intelligent Control*, Vancouver, Canada.

Davis, D N. (2003). Architectures for Cognitive and A-Life Agents. In: *Intelligent Agent Software Engineering*, V. Plekhanova (ed.), IDEA Group Publishing.

Davis D N. (2008). Linking Perception and Action through Perception, *Journal of Experimental & Theoretical Artificial Intelligence*, 20(1):37 - 60.

Diestel, R. (2005). *Graph Theory* (3rd ed.), Berlin, New York: Springer-Verlag

Flavell, J.H. (1979). Metacognition and cognitive monitoring: A new area of cognitive developmental inquiry. *American Psychologist*, 34, 906-911.

Franklin, S. (1995). *Artificial Minds*, MIT press.

Franklin, S. (1997). Autonomous Agents as Embodied AI. *Cybernetics and Systems*, 28(6):499-520.

Gobet, F., Lane, P.C.R., Croker, S., Cheng, P. C-H., Jones, G., Oliver, I., & Pine, J. M. (2001). Chunking mechanisms in human learning. *Trends in Cognitive Science*, 5:236-243.

Kahneman, D. & Miller, D.T. (1986). Norm Theory: Comparing reality to its alternatives, *Psychological Review*, 93(2):136-53.

McCarthy, J. (1959). Programs with Common Sense, *Proceedings of the Teddington Conference on the Mechanization of Thought Processes*, Her Majesty's Stationery Office, London.

McFarland, T. & Bosser, T. (1993). *Intelligent Behavior in Animals and Robots*, MIT press, Massachusetts.

Minsky, M. (1985). *The Society of Mind*, Simon and Schuster, New York.

Nason, S & Laird, J E. (2004). Soar RL integrating reinforcement learning with Soar, *International Conference on Cognitive Modeling*, 2004.

Newell, A. & Simon H. (1972). *Human Problem Solving*. Englewood Cliffs, NJ: Prentice-Hall, 1972.

Newell, A. (1990). *Unified Theories of Cognition*. Cambridge, MA: Harvard University Press, 1990.

Rolls, E.T. (1999). *The Brain and Emotion*, Oxford University Press.

Savage, T. (2003). The grounding of motivation in artificial animals: indices of motivational behaviour, *Cognitive Systems Research*, 4:23–55.

Selfridge, O. (1959). Pandemonium: A paradigm for learning. *Proceedings of Symposium on the Mechanization of Thought Processes*.

Singh, P. (2005). *EM-ONE: Architecture for Reflective Commonsense Thinking*, Laboratory, MIT, PhD Thesis.

Sloman, A. (2002). How to think About Cognitive Systems: Requirements and Design, *DARPA Workshop on Cognitive Systems*, Warrenton, Virginia, USA

Stillings, N. (1995). *Cognitive Science*. Cambridge, MA: MIT Press.

Sutton, R.S. & Barto, A.G. (1998). *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA.

Tesfatsion, L. & Judd, K.L. eds. (2006). *Handbook of Computational Economics: Volume 2, Agent-Based Computational Economics*, Handbooks in Economics Series, North-Holland, Elsevier, Amsterdam, the Netherlands.

Thorndike, E. L. (1911). *Animal intelligence: Experimental studies*. New York: Macmillan.

Toda, M. (1962). Design of a Fungus-Eater. *Behavioral Science*, 7:164-183. The Hague: Martinus Nijhoff Publishing.

Venkatamuni, M. V. (2008). *A Society of Mind Approach to Cognition and Metacognition in a Cognitive Architecture*, PhD Thesis, Computer Science, University of Hull, August 2008.

Wahl, S. & Spada, H. (2000). Children's Reasoning about Intentions, Beliefs and Behaviour. *Cognitive Science Quarterly*, 1:5-34.

Yoshida, E. (2007). *The Influence of Implicit Norms on Cognition and Behaviour*, MA Thesis, Dept. of Psychology, University of Waterloo, Canada.