COMPREHENSIVE PROTOCOL FOR ARTIFICIAL INTELLIGENCE DEVELOPMENT

Bruce L. Toy

Lockheed Martin (Ret), Palm Harbor, FL USA

brucetoy@verizon.net

ABSTRACT

This paper presents a Functional Architecture and an Integration Protocol that support a comprehensive intelligent system. The structure is based on XML objects. Motivations drive all actions. The behavior processes use templates of stored information augmented with real-time sensor data to execute actions. Actions are controlled on independent "virtualized" segments of a processor. The brain forms predictive models of the behavior of objects it interacts with. It does this by creating simplified clones of its own behavior control structure. Secondary motivations and action preferences are linked to locations. Sensor processing and interaction with the environment is made more efficient through the creation and use of location models. A Frame of Reference defines the brain's individuality. Training, rather than programming is the focus of development.

Keywords: AI, artificial intelligence, brain modeling, behavior modeling, cognitive process, learning

1. INTRODUCTION

This paper defines the brain in terms of a complete functional architecture and captures its essential nature, defining an approach for Artificial Intelligence implementation. There are details that help to understand complex human behavior. The focus is on the "forest" rather than the trees. Instead of analyzing things like pattern recognition algorithms (trees) the discussion looks at how to break down the forest into functional components with well defined internal interfaces. The components are described in fundamental, practical, terms with a level of detail that would support system development. There are aspects that go outside the bounds of traditional AI theory but are necessary in order to provide a complete architecture.

There have been multiple projects that seek to define a Cognitive Architecture for handling AI. Most focus on behavior modeling with some success. One good example is Soar, developed by Newell and Laird. It includes refined logical algorithms for selecting actions, but it does have a drawback. Like others, it must be managed by an external entity – human or human-produced computer program in order to be effective. A true architecture must incorporate the totality of mental functions and be independent.- living or dying on its own. Note that I am not discounting such models such as the Cognitive Architectures created to date. Instead, I expect that this protocol could facilitate the integration of diversely developed AI elements.

1.1. Virtualization

There is an important software tool that is cited in this architecture, virtualization. (Adams and Ageson 2006) It has always been possible to set up sub-processor segments with specialized connections and functional programs. In the past this was a time-consuming task not easily automated.

Virtualization allows us to create a variable portion of a processor to be set aside for a particular task. In this protocol we call this an "action template" by which a function is activated. The template defines the way in which a processor segment is established and linked to other memory objects and input/output interfaces.

Lower animals generally have templates that exist without training. In intelligent systems there are basic defined templates but, in addition, training drives the system to build new templates of the action concepts. The application of the virtualization process will be discussed further in section 2.

1.2. Protocol Structure

The core of the protocol consists of seven layers that define the complete set of memory objects in the brain. Around this there is a wrapper consisting of a pair of functions that manage the operation and the accounting needed to control the intelligent system. The first function, the Control Program (CP), runs the operational environment and the second function, the Cognitive Structure program (CS), runs the development and administrative environment. Any computer developer will tell you that operating systems and development environments must be segregated. These two management functions are separate for similar reasons. This is one of the features that set this protocol apart from most other approaches to AI.

1.3. Protocol Definitions

The general elements in this architecture are defined as follows:

Memory concepts are groups of links in the memory that have their own identity, with three types: Object Concepts are the fundamental memory concepts that can be associated with things the system can detect or purely abstract things or ideas. Functional Concepts (also called Action Concepts or action templates) define and manage the actions your brain can execute. The functional concept is, in essence, a template for a specific activity. It is "hosted" on a virtualized segment of the processor. Motivation Concepts are the triggers that begin all action. It is only when a sensor input triggers a motivation that any action can occur.

Location Models are structures of concepts associated with locations that the system encounters. Sometimes a location model is purely imaginary, as one created while reading a book.

Behavior Models are created by the Cognitive Structure program and are attached to all object concepts.

The Cognitive Structure (CS) program manages the memory storage through creation of links and their values (prioritization.) It establishes definitive object concepts, location models, and behavior models.

The Control Program (CP) controls the execution of actions based on link analysis. The CP makes things happen, not by commanding actions but by *allowing* a functional concept to operate as an independent action with temporary control of a portion of the brain.

The following diagram shows the complete architecture/protocol that is the basis of this paper. Each element will be discussed in detail.

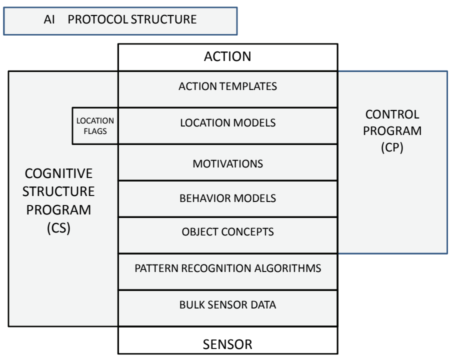

Figure 1: AI Protocol Structure

2. CONTROL PROGRAM (CP)

The Control Program (CP), one of two separate management functions, is responsible for executing actions. In order to maximize efficiency the CP has the primary role of deciding which of the available functional concepts (action templates) deserve to use a portion of the limited processor resources. It ensures that each one uses only as much of the processor as necessary and only for as long as necessary.

In this architecture, the brain is represented by a central processor that can be partitioned into independent processing units of variable size and quantity, in essence, a divisible processor.

The CP applies no logic or decision making beyond the evaluation of link prioritization – the relative strength of the link values associated with each motivation and action. These link values are managed by the CS

The control of a portion of the processor is delegated to the individual functional concept. The "permission" to use the partition of the processor is managed by the CP based on motivations and link analysis. Any functional concept allowed to use a part of the processor will function independently until such time as its permission is withdrawn and it is disconnected.

The activation process begins when a motivation is triggered by a sensor input. The CP then reviews all active motivations to assess the priority of responding to the motivation. The nuances of this process will be enumerated as we describe the complete protocol. For now simply note that stored motivations, both positive and negative, along with the priorities assigned to their links will drive the selection of action by the CP.

If the AI structure is hosted on a reasonably powerful computing platform, one purpose of the biological CP, optimizing efficiency, is not as important. It does, however, remain as a vital element in replicating intelligence with a machine through its control of the operating environment and implementing motivations.

3. MOTIVATIONS

Some motivations are obviously inherent in the brain. Some attempts to introduce motivations focus on emotion (Minsky 2006). Such approaches have been more philosophical than practical. Biological motivations have a strong emphasis on survival, reproduction, and factors that are often not the primary interests of the AI system. We must define specific motivations for the AI and integrate them in a measurable way.

Following that, secondary motivations are one of the keys to intelligent behavior. They are generated based on experience and/or instruction and are linked to actions that result from the learned behavior. These links are given preference by link value based on instruction or experience, reinforced by training.

In addition, motivations are linked to location models, which are described later. They allow us to organize behavior patterns and link motivations and actions within specific locations.

When you go beyond the basic motivations, the use of an instructor becomes critical. Furthermore, the ability of an instructor to guide the learning process exists only to the extent that the student has an associated motivation framework. In instructional settings this takes the form of a motivation by the student to satisfy or please someone else. For an AI this motivation should focus on an educational team, to the exclusion of other individuals. Human students will have additional internal self-motivations, in varying degrees, to support the learning process. The AI should, at some stage, transition to an internal self-motivated driver to partly, or even completely, assume an independent pursuit of learning.

Secondary (learned) motivations are critical and without them it is quite possible you will not have an intelligent system. These will be discussed in more detail in the following sections.

Negative motivations (inhibitors) are needed to identify actions that produce incorrect results, waste time, consume too much energy, or are otherwise undesirable. To achieve this we will allow the AI to contain only positive motivations with both positive and negative link values. While this probably does not mimic the biological system, which contains separate positive and negative motivations, it simplifies the design and makes the analysis task of the CP much more straightforward and less prone to errors.

It should also be noted that in some animals, but especially in humans, there are many strong motivations that appear in stages as the brain matures. The most active time for new motivations is probably the adolescent puberty stage. Also with age some motivations are reduced or disappear entirely.

This strongly suggests that for complete development of the AI we should introduce motivations in steps, indexed to achievement of learning. This would support a more passive, instructor-based system in the beginning and a transition to a more active system with added interaction with the environment, humans, and other AI systems. Maintaining control of the motivations means that we add and subtract some of them, or change priorities, throughout the life cycle of the AI.

3.1. Frame of Reference

Motivations are linked to a Frame of Reference. The AI system must have a “frame of reference”. In its most basic form this defines the biological individual but there are variations that allow the individual to identify itself with some group beyond its physical self. This may be a social group, a family, a religion, a culture, or any of a variety of groups that the individual identifies with.

The importance of the frame of reference is both subtle and powerful. It is closely tied to motivations and helps to define consciousness. In developing an AI system, the frame of reference chosen can allow the AI to evolve a distinct consciousness of its own. The consciousness can be human-like or intentionally modified to an even more complex form not yet seen in the biological realm.

The AI should have a “displaced” frame of reference, where the focus of its motivation is outside its own structure. This will allow it to better represent a particular object or entity. More importantly, however, this can be used to prevent the AI from developing an egocentric behavior pattern.

The key point to remember at this stage is that memory data is recorded based on the connections to, and the impact on, the Frame of Reference. The strength of a link will be directly proportional to that level of impact.

4. COGNITIVE INTEGRATION

One thing that makes it difficult to explain this model in one straight narrative is that there is considerable interaction between the processes. I will now describe the interrelated functions of location modeling, sensor dynamics, learning (including the memory management of learning) and, most importantly, predictive behavior modeling. Each is dependent on the others and it is difficult to begin with any one of them because each description will not seem clear until all are integrated.

4.1. Location Modeling

Location models support a significant part of intelligent behavior and the organization the brain. Though models may contain a large number of objects the location models themselves are simple, with links to a collection of specific object concepts in memory.

One aspect of learning is building the location models of our environment. This is supported by a fundamental motivation to identify one’s location and identify objects associated with that location. With maturity we develop the location models for all the places we encounter. We can also compose models for places that are described to us.

Many children’s books focus on location modeling. They establish generic models of such places as farms, deserts, jungles, oceans, and so forth then link other object and activities to them. This activity both entertains and educates the kids,

Three specific concepts are linked with location models. First, there are secondary motivations, developed by training. These can be triggered by entering a location or, by object concepts found there. Second, and very important, are action templates tailored specifically to the location. These lead to preferred behavior patterns in specific locations. This means the CP will automatically activate the action templates for the preferred behavior functions in a given location (absent other negative motivations, of course). This will be expanded upon later. Third, there are links to behavior models of objects found in a location. This, in turn, impacts other actions we take.

It is important to note that the vast majority of the data in a location model is simply linked into it from other memory objects.

A location model can be purely abstract, or imaginary. When you read a novel you create location models of places described in the book. A good author builds these models of places and the reader can experience them through links to their own mental concepts.

The Cognitive Structure (CS) program is not only the tool that constructs the location models; it serves another function as well. It registers a “flag” to indicate all models that you are currently part of. When you change locations it changes the status flag. You will be part of several locations at one time, starting with your immediate vicinity and moving up to community, state, and so forth.

It may seem obvious to some but I should point out that any but the lowest forms of animal life will create and remember location models. A rabbit, for example, will have a complete model of its home, feeding areas, trails, water sources, and other features of its local environment. The lab rat that learns to negotiate a maze is doing nothing more than creating a specific location model.

4.1.1. Controlling Action with Location Models

Once we have a location model in memory it serves as a linking point for action concepts.

The models affect preferred actions, or default behavior, through the attached action templates. For example, when you enter a neighbor's home your behavior will not be the same as in your own home. While the two locations are essentially similar in characteristics, the actions are driven by a largely separate set of default behaviors – all driven by learning and secondary motivations.

These memory links for action templates are structured in such a way that the CP sees a high link value when a motivation is triggered in association with a specific location model. Secondary (learned) motivations that are created in association with specific locations will produce actions in those situations and may not even be options in other locations. This occurs partly through directed training and partly through simple experience. For example, the hunger motivation produces different actions in different locations simply based on experience. Just entering a movie theater (along with specific smells) may compel many individuals to buy a large tub of popcorn, an action that they would never execute anywhere else.

This arrangement defines what could be called the "default" set of functional templates that are active in the processor at a given time. Each location model essentially guides the CP to maintain a processor segment for specific actions during normal activity in that location. Note, of course, that the location does not actually dictate action. The CP still has ultimate control and any overriding motivation and higher link values can change the priorities.

We act and react differently in different settings. This is not news to anyone but now, at least, you know how it happens.

I should also note that, in addition to location flags, the CS maintains another sort of flag for each location. It appears to identify the most recent and/or the most important action templates associated with a given location – sort of a "save game" function that allows you to pick up where you left off when you re-enter a location.

4.1.2. The Shopping Algorithm Problem

Some who study intelligent behavior point out the difficulty of creating an automation of the process of shopping in, for example, a grocery store. Perhaps the model offered here gives a methodology to address this issue.

One curious thing about location models is that you can create temporary motivations within them, either consciously or not. I would almost call them bookmarks and they can motivate an action when the location is encountered.

As for shopping, let's say that during the course of a day you note that some of your kitchen supplies are low but you are not able to go to the market until tomorrow. You create a temporary motivation to replace the ketchup. Some organized people make a list. Others make bookmark motivations in either the kitchen model or the store model. Thus, when you arrive at the store the sight of the ketchup bottles triggers the bookmark motivation to get more ketchup. This can be accomplished by either linking the motivation to the store model or by linking the kitchen model to the store model and I would expect that some individuals may do it either way.

In fact we can explain why men and women might seem to approach the task of shopping differently. Some people seem to move through a store, looking at everything, waiting for something to trigger a motivation. Others seem to have the motivation set to drive action by moving straight to the objects of interest – that is, the motivation triggers an action, as a result of entering the location, and takes the individual to a specific part of the location.

It becomes clear that the difficulty in creating a "shopping" algorithm is that the shopper needs the context provided by the location models and attached secondary motivations. In this AI structure we can implement a straightforward solution.

4.1.3. Other Links Within Locations

Other links exist that are not directly relevant to artificial intelligence but are noteworthy nonetheless. An "emotional" link is a behavior model that is attached to an object within the location that has an output (input to our system) that strongly satisfies one of our own motivations or, conversely, is a threat to our system.

Some sensor inputs will activate the imagination to visualize and experience a specific location and its links. In some people this can be very vivid. The input may be very simple. The sight of an object, the sound of the ocean, or a gunshot, as examples, can trigger strong images and reactions, almost as if one was in the specific location.

4.2. Sensor Processing

One of the more difficult problems associated with designing an AI is the handling of sensor inputs. Most of the discussion for this topic will center on visual data, but the ideas will apply to any type of sensor input for the AI system. Any AI must be compatible with the use of a variety of sensors, however not all AI systems will require the same kinds of complex sensing systems.

The processing of visual data is a highly specialized field. What I will describe here is a system for managing the data rather than the actual processing. I will only describe the framework in which the sensor data will be used and the general approach for processing. I will not attempt to duplicate or improve upon the various processing pattern recognition algorithms and analysis techniques that exist.

One of the keys to this process is the object concept we store in memory. At the core of each object is a pattern recognition algorithm that allows us to correlate the object in memory with an object in the visual field. Each pattern recognition algorithm is attached to the associated object concept. We should note at this point, though it will be discussed in more detail later, that some pattern recognition relies on simple high-level characteristics and associated location models.

You do actual pattern recognition on the small, central part of your visual field. The rest of the visual space is processed only for basic details. This includes color, intensity, texture, distance, motion, and perhaps a few others. I will refer to this broad group of processed data generally as "field data". This data is then presented in the brain as a simple flat array with the characteristics of the pixel fields measured and shown. The objects outside the fovea are identified primarily by the determination of a specific location model (i.e. where you are) and the "field data". This does not rise to the complexity of actual pattern recognition.

In this regard, location models are very important. The field data combined with the objects expected for that location give a presumptive identification of those objects outside the central viewing field, outside the region of pattern recognition. Pattern recognition, on the other hand, is applied to the objects you focus your attention on.

If, for example, trees are part of your location model then any objects outside the fovea that have the correct basic "field data" (color, movement, size, etc.) are identified as trees without you needing to focus on them or do any special pattern recognition. When you walk through a forest you do not need to focus your attention on an object to classify it as a tree or a bush. Note, particularly, that if there is something in that location that is similar to a bush you are not likely to notice the difference even if it is not really a bush. That is the purpose of camouflage.

What are the implications? The point is that in real-life situations, when you finally get around to doing pattern recognition you have already (1) identified your location, and (2) know the typical objects you will find there. You then apply the pattern recognition algorithms for the candidate objects that you decide to focus your interest on. This does not mean that you will not identify other "unusual" objects. It just means that you first match objects with a particular location. If there are no matches for something unusual, only then do you do a full analysis of that object.

You can even direct yourself to ignore objects that are not expected—meaning you do not apply extraneous algorithms outside a specific group of your choosing. There is a popular video on the internet that asks the viewer to watch a group of individuals wearing different colors pass large balls around. You are asked to count the number of times individuals of a specific color pass a ball. In the middle of the video a guy in a gorilla costume walks through the scene. If you are good at focusing your attention you never even "see" the gorilla. You have specifically directed your mind to focus on specific types of pattern-recognition algorithms and process data for them only. In essence you have created, by direction, a temporary location model with a limited number of objects—and it does not include any gorillas. You can then focus all of your attention on that location model to perform an assigned task. Extraneous objects may not be "seen".

Why did I mention this example? You must be prepared for your AI system to overlook a gorilla if you have specifically asked it to focus on other specific objects. Only then will you know you have created a truly human-like AI.

This all makes it possible to store longer-term information about visual inputs very efficiently, but it must be noted that we are storing processed data, not the complete sensor input. This makes us susceptible to inaccurate memories, based upon external suggestions or misperceptions of what we see. If we design a comparable AI we must be prepared to accept the reality that memories (not the real-time input) will contain inaccuracies, though it may be possible to minimize this by careful attention to the design of the sensor processing algorithms.

4.3. Linking Sensor Data

The XML linking structure must have two basic types of links in the memory. The first is long-term memory. These links decay over time as a normal process, but the time constant for this decay is quite long. The second is short-term links with a rapid decay time. This type of linkage is principally used with real-time sensor data. The sensor programs analyze inputs and the short-term links are created to connect the input information with the stored memory concepts.

This allows action concepts to act on the information, through their specific interaction with that long-term object concept. The need to establish other more persistent long-term links for any input data is determined by the template defined for that activity.

Perhaps more importantly, we can execute actions on objects simply because they are part of a location model. We do not actually need to focus our attention on, and do pattern recognition for, specific familiar objects that appear in a location.

Another consequence of this aspect of the model is that it enables us to better understand and define the differences between higher and lower animals. The higher animals, with greater mental capacity, will store correspondingly more details with their memory objects. The lower animals store only a few details of any given concept, using a simple set of object types. In interacting with those objects they rely on the short-term links to provide the details they need to execute functional actions. Experiments have shown that a frog, for example, responds in the same way to any small object that moves like an insect. The frogs stored memory concept appears to be based exclusively on the size and motion characteristics.

5. ABSTRACT TOOLS

We need a tool set that supports the CS and, to some extent, the CP. There is nothing particularly new or noteworthy about this set included here but it is necessary to describe it for completeness.

The four basic management tools are: **Identity**, **Linking**, **Matching** and **Difference**. This selection is supportable as a logical set of functions but its makeup may be arbitrary in some respects. There could be other breakdowns, but these four are used here both for constructing logical thought and performing other system management action of the CS.

Identity: This is a tool that creates a fundamental object concept of something you perceive. The concept is then further defined by links to characteristics.

Linking: This tool would link concepts with one another for object concepts, location models or templates.

Matching: This tool would evaluate two or more objects for similarity.

Difference: This tool would evaluate the differences found between two or more concepts by comparing them and linking them into appropriate sets.

For this AI model, these tools form the fundamental building blocks used by the CS Program. In part, this assertion is justified because any general abstract mental task can be broken down into some combination of these basic functional actions. It is also possible to see very distinct parallels between these basic functions and the corresponding first abstract abilities that appear in developing children. It is probably no coincidence that these functions (identity, linking, matching, and difference) play a fundamental role in most standardized intelligence tests.

6. DEVELOPING COMPLEX CAPABILITIES

While the CP manages actions, it is not the reasoning function. The CS manages the linking structure of the brain and provides the framework in which the actions can occur. It builds the action templates, the behavior models and the location models. These arise from sensor inputs and, more importantly, from instructive input.

The CS also has the capability to establish a temporary representative linking structure for the object concepts based on the input data itself and comparisons between concepts. Read a novel and your CS creates location models based on what the author describes. It can even proceed with analysis based upon a presumed, or temporary, set of links. This would occur, for example, when directed by a teacher to perform a new specific action.

The CS is always running in the background to create links. Beyond that, however, I must emphasize that the CS is not exactly a free-wheeling process. Like all other mental activity it must be triggered by a motivation. In that regard it is subordinate to the CP. Whether the motivation is an instructor's command or an internally-driven motivation such as boredom, it is only by consent of the CP that the CS can perform analytic functions or "test" action templates.

6.1. Difference between CS and CP

In this model the CP controls action by activating established templates. The CS controls reasoning by a tentative or limited activation (often without connection to outputs) of a template that is in the developmental stage. The prototype templates consist of temporary action templates and location models, complete with the object and behavior model connections. The CS undertakes control of a portion of the processor for the limited execution of the template. Action can be either executed or shunted to a register of projected action as controlled by the CS. The CS then reviews actual or projected results for effect on the AI system.

In some regards we may be tempted to design a system where the CP and the CS are one and the same. You can use almost the same software to run both. However, in software terms, the CP must control the "operating environment" while the CS controls the memory structure and the "development environment". Attempts to design an AI system that combines the two functions of the CS and the CP in a single process are almost certain to fail. It is mandatory that while the actual software could overlap, the functionality must be separated in a carefully conceived architecture.

Reasoning skills of the CS depend upon access to an adequate database of information, either memory links or input data. That is, after all, the only way it can actually build a prototype template. Perhaps the preferred method would be to use an existing template and make some modifications. The acquisition of data and the training can be achieved by either external instruction or by internally controlled responses to external inputs.

In the instructional case the input and direction, especially link associations, will be explicitly provided by someone or something else, to be assimilated by your brain. Internally controlled development is driven by the whims of nature and the motivations of the individual.

I would suggest that as the biological system passes from its developing stages into a state of more routine operations, the use of this CS would be reduced to identifying familiar locations and habitual behavior patterns associated with them. The biological system will have established preferred linking structures for many input-motivation-result-action sequences and would not require the services of the CS to evaluate alternate linkages. It will have a set of location models and actions that satisfy its needs.

This will lead to a decline in the reasoning skill. The links will diminish without use and the system will function primarily in established behavior patterns. The obvious alternative is that a system motivated, either internally or externally, to continue using the CS program to develop new action concept templates will remain more capable. Thus we can understand the importance, at least in humans, of providing new mental challenges following maturity.

6.2. Training

We can make a minor departure from the biological process that simplifies the AI learning in a structured environment where the trainer can specify exactly what the system must learn. As a programmer we can take a short cut by directing the "link values" for a concept. This would emulate a lengthy training period that produces strong links in order to reduce training time. There would still be training sessions but repetitions to produce high link values would be unnecessary (Hibbard 2004).

It may seem logical to expand this approach to the point of making "modular" action concept templates that could be separately developed and integrated into an AI as single units. I would not categorically reject this thought. However, many links in a template specifically depend on the Frame of Reference. If these are incorrectly or incompletely produced then the results could be less effective or, worse, could be destructive.

If we have a "simplified learning" mode it must be controlled by an instructor to mimic the result of repeated exercises by establishing high link values at the outset.

7. INTELLIGENCE IS PREDICTIVE – NOT REACTIVE

Now we have come to one the core principles of intelligence. This is one of the most important features of the brain's architecture.

The intelligent brain creates a behavior model of every object in its memory data base. This model is attached to each object and provides a prediction function for that object. This is not just a passive prediction. The behavior model includes, quite importantly, the object's responses to your actions. For every object concept there will be an associated model that predicts behavior for that object, either as an individual or general class.

You may expect this to be a complicated process and very difficult to explain; however, it is actually rather straightforward and a relatively simple process if you accept the functional model defined for the brain up to this point in this paper.

The key element is what I would call the backbone of your mental structure. That would be the motivation-action process managed by the CP. The sensor detection of an object activates a motivation and the CP assigns a responsive action.

To create each behavior model, the brain uses this self-contained function to simulate the behavior it expects from other entities.

7.1. Behavior Models

The predictive model for each object is merely a brief, simplified version of this larger AI model—in essence it is a modified copy of your own mental process. It is created for every object and tailored to the perceived actions of that object.

The CS creates and executes a model for each object and its predicted behavior in a simple and straightforward way. It establishes a set of input, motivation, action linkages that represent the way it perceives that object to respond to its environment. Each model functions identically to your own biological mental process. The examples below will help explain that idea.

I would suggest that, for simplicity, there are three general types for the predictive model, based on specific characteristics. The processes for all three are much the same and these three model types differ only in the way we might perceive them.

7.1.1. Human-Equivalent Behavior Model

The first, and most involved, would be the Human Equivalent Model. This model is a copy of your own motivation-action concept structure. It can be applied just as it exists in your own mind but usually it is modified based upon learned data you have for the object involved. You may add a motivation, add an action, or change a priority link. Obviously we can apply it to other humans. In addition, we often extend this model to animals. In this case it is tailored to remove a portion of motivations and actions. However, children are particularly prone to use the complete model to define animal behavior.

7.1.2. Instruction Manual Behavior Model

The second type is the Instruction Manual Model. This simplified model replaces the sensor inputs with basic physical inputs, such as pushing a button, turning a knob, or manipulating some mechanical aspect of an object. This model provides defined results from the specific mechanical interactions, either by you or some other object. The obvious application of this type of model is for mechanized objects but you can apply it to a wider range of items.

7.1.3. Physics Model

The third is the Physics Model. This model generally provides action-reaction links and may be somewhat complex depending upon the types of objects being addressed. For physical objects in your environment the model will include responses to natural forces as well as your interaction with them (responses to pushing, kicking, sliding, etc.). In the physics model, motivation concepts are replaced by physical laws (e.g., an object is "motivated" to fall toward the earth because it "detects" the force of gravity); otherwise the predictive model is the same as your own mental process.

Remember, the behavior characteristics are based upon the way you interact with them as well as external forces. For example, if you push on the object, does it move? This is tied to a perception of relative mass. The Physics Model incorporates the physical characteristics of each object. That is to say, once you identify the relative mass of an object (whether by touching or by visual clues) you can project, using a physics model, responses to physical interaction. This is one reason children are always messing with things. They are constructing their models of all of those objects around them.

As a simple example, you see someone holding an object in their hand. You can predict what will happen when they release it simply based on the model you have attached to that object. You do not need to analyze the situation; you simply run an existing model. You identify that object by sight and you have a predictive model linked to that object concept in your memory.

Let's say you perceive the object to be a rock. Your predictive model says that the object, when released will fall to the ground under the influence of gravity. Alternatively, you perceive the object to be a paper airplane. Your predictive model says that it will glide away when released. If you perceive the object to be a live bird, then it should fly away. And, finally, you may perceive the object to be a balloon with yet another prediction of what happens to it when released.

7.2. Building Models

The learning process allows you to build this predictive model of any object you encounter. For example, children learn that dogs are motivated to chase and retrieve a thrown object, a very simple template of motivation and action to create in a child's mind. That and other functions are incorporated in the concept of a dog. As the learning base is expanded the child learns that observable characteristics of the dog can be used to distinguish different behavior models for different dogs. Then, whenever you see a dog you predict its behavior based on the model. You will have sub-groups of the model based on individual types of dogs or physical appearance. The model will tell you that if you approach the dog and try to interact with it there will be an expected response. For example, growling equals aggression (triggering a danger motivation); tail wagging equals friendliness, and so forth.

Now, moving to the highest level of complexity, you will have a model of behavior for other humans. You will likely have a generic model that would be applied to any new acquaintance. This could closely match your own behavior algorithms. Then, for each individual, the model will be tailored based upon what you have observed from that individual.

You will also develop secondary models for individuals based on secondary characteristics. These you likely know as stereotypes. You meet someone in a business suit and you link them to a generic "businessman" model. You meet someone with dirty, shabby clothes and you link them to the "bum" model. When someone says: "I can't believe he just did that;" it means that some behavior did not correspond to a model. As you witness a person's behavior your own CS function will refine the model.

One important offshoot of the predictive model is that we take the output of each model—the action that is expected of the object—and link that back to our own structure of inputs and corresponding motivations. Thus when you interact with someone or something that can affect you through their action, you are then motivated to create an input to that object that produces a desired action—something that would satisfy a motivation within your own structure. This produces the results seen in many social and business interactions.

7.3. Relation to Intelligent Agents

There are many efforts to develop Intelligent Agents. These are, by definition, autonomous and self-contained intelligent units that can take action on their own and/or provide information to a higher level entity.

One of the most complete descriptions of intelligent agents is provided in Artificial Intelligence, a Modern Approach (Russell and Norvig 1995). There are a variety of types of agents described but they share some general features for using sensor data and behavior models to produce intelligent behavior.

I would suggest that, within the protocol structure offered here, the intelligent agent represents a sub-set of sensor links, behavior models, and actions. All objects are created and maintained in a single action structure (rather than linked from a reference base.) The decision rules are created for limited scenarios and updated with observed inputs. Perhaps the biggest differential, however, is that there is as single management function that executes direct control and decision authority.

That said, it should still be possible to reconcile the differences in order to incorporate intelligent agents into this type of architecture.

8. VISUAL PROCESSING AND PREDICTION

At this point the discussion concludes by returning to something that was partly covered before. Re-consider the dynamic nature of your visual input. How do you handle the constantly changing tableau of images that your system must process? This is perhaps the most difficult part of the brain function to understand. Though it is sometimes presumed, there is not a recurring analysis of a scene. A basic tenet here is that the brain works from stored memory concepts with augmented short-term data from sensors. Any object we sense around us is identified based on an array of knowledge. Furthermore, this is always a presumptive identification, though sometimes this is a near certainty.

This is where the location models and the behavior models come together for mutual support. They are linked together through the object concepts contained in a location.

Once we identify a location and associated objects, the subsequent visual processing is determined by any motivations and the predictive model we have for the associated objects. An object could be of great interest, meaning it is one whose outputs can benefit (or harm) our system. In that case the brain will perform continual updates of the predictive behavior analysis, evaluating the impact to our system. Note that this is a behavior analysis not a visual analysis. On the other hand miscellaneous objects of no direct interest are summarily identified and receive a prediction, e.g., that picture on the wall will stay right there.

The background of the location is defined based on the location model and it is that stored memory concept we use until something changes. Until routine objects depart from the predicted behavior there is little visual processing to be done. Background objects, such as trees, are identified primarily by “field data.” That is to say we do not use explicit visual processing algorithms if they fit the background structure of our location model. Motivations can, of course, change all that. After all we could be looking for a specific kind of tree.

The impact of this analysis structure is to drastically reduce the demand for processing power because it is no longer necessary to perform the extensive scene-by-scene visual processing. Those cars passing by you on the highway are one of the best examples. You identify them summarily and use a standard memory object. You only care about them if they divert from the predicted pattern in such a way as to be a threat to you. A beginning driver, who is still in the process of completing the predictive models for all these new objects, is flooded with information. That new driver must learn to focus attention and that does not happen until a reliable set of predictive models are formed.

9. OTHER USES OF LOCATION MODELS

There is an adjunct to the use of location models that shows a possibility for understanding how we construct and use sentences to communicate with each other. If true, this would have a significant impact on other aspects of AI and communication dynamics.

Within the framework of this architecture we could reasonably believe that sentences, whether spoken or written, could communicate their thought by the creation of a temporary location model in the receiver, especially when you consider the additive nature of a conversation or string of sentences.

This process would also allow one to identify and resolve conflicts in a sentence. If, following a statement, your location model is not incomplete or has ambiguities. You would seek to resolve the ambiguity by asking for more exact information, but only if you are motivated to do so. If you have no interest in the subject then you might have no reason to seek resolution.

This use of location models also leads to an understanding of some forms of humor. When someone is telling a joke you are building a location model to match their narrative. At the conclusion of the joke, you discover that your model was not the same as the one the narrator finally revealed. At that point the intention is that you find it funny, though that is not guaranteed.

This use of location models for understanding sentence structure deserves further consideration. The concept does appear to be compatible with the architecture and supports the other aspects of interactions involving this AI protocol.

10. SUMMARY

What is shown in this paper is both a Functional Architecture and an Integration Protocol. We have defined the modular components of the system, their functions, and interfaces. This protocol will enable coordinated individual development of the fundamental components of a system that can be integrated in a dynamic environment. As long as a strict motivational structure and frame of reference are maintained the components can be remotely connected to form a single intelligence.

The greatest strength of this architecture, beyond its internal consistency, is that individual development teams could be assigned to produce defined components. As long as functional interfaces are controlled the pieces could be integrated to work as a single entity. Conversely, if one attempts to cut out portions of this architecture and splice them into other kinds of systems, the results are likely to be disappointing.

For a potential application, imagine if you will a system that would integrate large numbers of different types of sensors in such a way that many objects can be controlled or addressed by a single entity, without the need for external coordination among separate systems. This can be applied to a range of operations from business to battlefield.

REFERENCES

Adams, Keith and Agesen, Ole (2006) A Comparison of Software and Hardware Techniques for x86 Virtualization, VMWare, ASPLOS’06 Oct 21–25

Hibbard, B. 2004. Reinforcement learning as a Context for Integrating AI Research. Presented at the *2004* AAAI Fall Symposium on Achieving Human-Level Intelligence through Integrated Systems and Research.

Minsky, Marvin *The Emotion Machine*. New York, NY: Simon & Schusterl

Russell, S. J. and Norvig, P (1995) *Artificial Intelligence: A Modern Approach.*. New Jersey: Prentice Hall

AUTHOR’S BIOGRAPHY

The author is an engineer who has been successful in a related area of satellite system architecture and development though not directly involved in the academic AI world. **Bruce Toy** boasts a 40-year career as an engineer primarily in the development of large-scale systems and supporting infrastructure. He received a BS in Physics from UNC and an MSEE from AFIT. He is a retired USAF officer with experience in ballistic missile defense as well as satellite design and development. The AF career was followed by 20 years in private industry, primarily with Lockheed Martin, where he became a respected systems engineer and systems architect for complex integration activities. He is currently semi-retired and living in Florida.