LIDA: A Working Model of Cognition

Uma Ramamurthy (urmmrthy@memphis.edu)
University of Memphis, Memphis, TN 38152, USA

Bernard J. Baars (baars@nsi.edu)
The Neurosciences Institute, San Diego, CA 92121, USA

Sidney K. D'Mello (sdmello@memphis.edu)
University of Memphis, Memphis, TN 38152, USA

Stan Franklin (franklin@memphis.edu)
University of Memphis, Memphis, TN 38152, USA

Abstract

In this paper we present the LIDA architecture as a working model of cognition. We argue that such working models are broad in scope and address real world problems in comparison to experimentally based models which focus on specific pieces of cognition. While experimentally based models are useful, we need a working model of cognition that integrates what we know from neuroscience, cognitive science and AI. The LIDA architecture provides such a working model. A LIDA based cognitive robot or software agent will be capable of multiple learning mechanisms. With artificial feelings and emotions as primary motivators and learning facilitators, such systems will 'live' through a developmental period during which they will learn in multiple ways to act in an effective, human-like manner in complex, dynamic, and unpredictable environments. We discuss the integration of the learning mechanisms into the existing IDA architecture as a working model of cognition.

Introduction

Many of the current models of cognition are experimentally based models. Experimental evidence is the standard in science, though such experimentally testable models provide few variables to accomplish cognitive processes such as perception and action-selection. Most of these models do not have the broad spectrum to address all aspects of cognition. We suggest that workability must be combined with empirical evidence in models of cognition. Such working models of cognition (WMC) are complex, covering a vast breadth of cognitive processes including perception, working memory, transient episodic memory, action-selection, emotions and feelings, declarative memory, and various forms of learning. These working models also provide testable hypotheses to assist us in building better software agents and cognitive robots that can adapt and 'live' in complex worlds. Here we present one such working model of cognition.

The Learning Intelligent Distribution Agent (LIDA) architecture was designed to be consistent with what is known from cognitive science and neuroscience. In addition to being a computational architecture, it is a working model of human cognition. We'll describe below the LIDA architecture and its human-like learning capabilities.

The LIDA Technology

LIDA provides a working conceptual and computational model of cognition. She is the learning extension of the original IDA system implemented as a software agent. IDA 'lives' on a computer system with connections to the Internet and various databases, and does personnel work for the US Navy, performing all the specific personnel tasks of a human (Franklin 2001). With the help of feelings and emotions as primary motivators and learning facilitators, the LIDA architecture adds three fundamental, continuously active, learning mechanisms to our existing IDA system that underlie much of human learning: 1) *perceptual learning*, the learning of new *objects*, *categories*, *relations*, etc., 2) *episodic learning* of events, the *what*, *where*, and *when*, 3) *procedural learning*, the learning of new *actions* and *action sequences* with which to accomplish new tasks.

The LIDA Architecture

The LIDA architecture is partly symbolic and partly connectionist with all symbols being grounded in the physical world in the sense of Brooks (1990). The mechanisms used in implementing the several modules have been inspired by a number of different 'new AI' techniques (Drescher 1991; Hofstadter & Mitchell 1994; Jackson 1987; Kanerva 1988; Maes 1989). The architecture is partly composed of entities at a relatively high level of abstraction, such as behaviors, message-type nodes, emotions, etc., and partly of low-level codelets (small pieces of code). Describes below are LIDA's primary mechanisms. Perception, episodic memory, procedural memory, and action selection will be revisited in greater length later on in the paper.

Perception. LIDA perceives both exogenously and endogenously with Barsalou's perceptual symbol systems serving as a guide (1999). The perceptual knowledge-base of this agent, called perceptual associative memory, takes the form of a semantic net with activation called the slipnet, a la Hofstadter and Mitchell's Copycat architecture (1994). Nodes of the slipnet constitute the agent's perceptual symbols, representing individuals, categories, and perhaps higher-level ideas and concepts. Pieces of the slipnet containing nodes and links, together with perceptual codelets with the task of copying the piece to working memory, constitute Barsalou's perceptual symbol simulators (1999). Together they constitute an integrated perceptual system for LIDA, allowing her to recognize, categorize and understand.

Workspace. LIDA solves routine problems with novel content. The current percept (slipnet nodes over threshold with their appropriate links) are written to the workspace, which roughly plays the same role as the preconscious buffers of human working memory. Perceptual codelets write to the workspace as do other, more internal codelets. Attention codelets (see below) watch what's written in the workspace in order to react to it. Part, but not all, of the workspace, called the focus, by Kanerva (1988) is set aside as an interface with transient episodic memory (TEM) and declarative memory (DM). Items in the workspace decay over time, and may be overwritten.

Another pivotal role of the workspace is the building of temporary structures over multiple cognitive cycles (see below). Perceptual symbols from the slipnet are assimilated into existing relational and situational templates while preserving spatial and temporal relations between the symbols. The structures in the workspace also decay rapidly.

Episodic Memory. LIDA employs sparse distributed memory (SDM) as its major associative memory. SDM is a content addressable memory that, in many ways, is an ideal computational mechanism for use as a long-term associative memory (Kanerva 1988). The LIDA architecture uses variants of SDM to implement episodic memory (Ramamurthy, D'Mello, & Franklin 2004).

It has been hypothesized that humans have a content-addressable, associative, transient episodic memory (TEM) with a decay rate measured in hours (Conway 2001, Franklin et al 2005). Humans are able to recall in great detail events of the current day – where they park their cars, whom they met that morning, what they discussed, what they had for meals, etc. We hypothesize that for cognitive agents to recall such details of episodes while they interact with and adapt to their dynamic environments, they need a TEM. Therefore episodic memory in LIDA consists of a TEM and a declarative memory.

Consciousness Mechanism. LIDA's 'consciousness' module implements Global Workspace theory's (GWT) processes by codelets, small pieces of code each running independently. These are specialized for some simple task and often play the role of a daemon watching for an appropriate condition under which to act. The apparatus for 'consciousness' consists of a coalition manager, a spotlight controller, a broadcast manager, and of attention codelets that recognize novel or problematic situations.

Codelets also have activations. Upon noting a suitable situation, an attention codelet will increase its activation as a function of the match between the situation and its preferences. This allows the coalition (collection of related codelets), if one is formed, to compete for 'consciousness.' Such coalitions are initiated on the basis of mutual associations between attention codelets. During any given cognitive cycle (see below), one of these coalitions with the highest average activation finds its way to 'consciousness,' chosen by the spotlight controller. GW theory calls for the contents of 'consciousness' to be broadcast to every codelet (Baars, 1988). The broadcast manager accomplishes this.

Procedural memory. Procedural memory in LIDA is a modified and simplified form of Drescher's schema mechanism (1991), the scheme net. Like the slipnet of perceptual associative memory, the scheme net is a directed graph whose nodes are (action) schemes and whose links represent the 'derived from' relation. Built-in primitive (empty) schemes directly controlling effectors are analogous to motor cell assemblies controlling muscle groups in humans. A scheme consists of an action, together with its context and its result. At the periphery of the scheme net lie empty schemes (schemes with a primitive action, but no context or results), while more complex schemes consisting of actions and action sequences are discovered as one moves inwards. In order for a scheme to act, it first needs to be instantiated and then selected for execution in accordance to the action selection mechanism described next.

Action selection. The LIDA architecture employs an enhancement of Maes' behavior net (1989) for high-level action selection in the service of feelings and emotions. Several distinct feelings and emotions operate in parallel, perhaps varying in urgency as time passes and the environment changes. The behavior net is a digraph (directed graph) composed of behaviors (instantiated action schemes) and their various links. As in connectionist models, this digraph spreads activation. The activation comes from three sources: from pre-existing activation stored in the behaviors, from the environment, and from feelings and emotions. To be acted upon, a behavior must be executable, must have activation over threshold, and must have the highest such activation.

The LIDA Cognitive Cycle

Be it human, animal, software agent or robot, every autonomous agent within a complex, dynamical environment must frequently and cyclically sample (sense) its environment and act on it, iteratively, in what we call a cognitive cycle. We suspect that cognitive cycles occur five to ten times a second in humans, cascading so that some of the steps in adjacent cycles occur in parallel (Baars & Franklin 2003, Franklin et al 2005). Seriality is preserved in the conscious broadcasts. We now describe the cognitive cycle dividing it into nine steps. In each step of the cycle, the role of feelings and emotions is emphasized by being italicized.

1) Perception. Sensory stimuli, external or internal, are received and interpreted by perception, producing the beginnings of meaning. Note that this stage is preconscious. *Pertinent feeling/emotions are recognized along with objects and their relations by the perceptual associative memory system, entailing simple reactive feelings based on a single input or more complex feelings requiring the convergence of several different percepts over multiple cycles.*

2) Percept to preconscious buffer. The percept, including some of the data plus the meaning, as well as possible relational structures, is stored in preconscious buffers of LIDA's working memory (workspace) by perceptual codelets. In humans, these buffers may involve visuo-spatial, phonological (Baddeley & Hitch 1974), and other kinds of information. *Feelings/emotions are part of the preconscious percept written during each cognitive cycle into the preconscious working memory buffers.*

3) Local associations. Using the incoming percept and the residual contents of the preconscious buffers, including emotional content, as cues, local associations are automatically retrieved from transient episodic memory (TEM) and from declarative memory and stored in long-term working memory. *Feelings/emotions are part of the cue that results in local associations from transient episodic and declarative memory. These local associations contain records of the agent's past feelings/emotions in associated situations.*

4) Competition for consciousness. Attention codelets view long-term working memory, form coalitions, and compete to bring relevant, urgent, or insistent events to consciousness. *Present and past feelings/emotions influence this competition for consciousness. Strong affective content strengthens a coalition's chances of coming to consciousness.*

5) Conscious broadcast. A coalition of codelets, typically an attention codelet and its covey of related information codelets carrying content, gains access to the global workspace and has its informational contents broadcast.

In humans, this broadcast is hypothesized to correspond to phenomenal consciousness. The conscious broadcast contains the entire content of consciousness including the affective portions. The contents of perceptual associative memory are updated in light of the current contents of consciousness, including *feelings/emotions*, as well as objects, categories and relations (perceptual learning). *The stronger the affect, the stronger the encoding in memory*. Transient episodic memory is also updated with the current contents of consciousness, including *feelings/emotions*, as events (episodic learning). *The stronger the affect is, the stronger the encoding in memory*. (At recurring times not part of a cognitive cycle, the contents of transient episodic memory are consolidated into long-term declarative memory.) Procedural memory is updated (reinforced) with *the strength of the reinforcement influenced by the strength of the affect* (procedural learning).

6) Recruitment of resources. Relevant schemes respond to the conscious broadcast. These are typically schemes whose context is relevant to information in the conscious broadcast. Thus consciousness solves the relevancy problem in recruiting resources. *The affective content (feelings/emotions) together with the cognitive content, help to attract relevant resources (schemes, processors, neural assemblies) with which to deal with the current situation.*

7) Setting goal context hierarchy. The recruited schemes use the contents of consciousness, including feelings/emotions, to instantiate new goal context hierarchies (copies of themselves) into the behavior net, bind their variables, and increase their activation. *It is here that feelings and emotions most directly implement motivations by helping to instantiate and activate goal contexts, and by determining which terminal goal contexts receive activation.* Other, environmental, conditions determine which of the earlier goal contexts receive additional activation.

8) Action chosen. The behavior net chooses a single behavior (scheme, goal context), from a just instantiated behavior stream or possibly from a previously active stream. *This selection is heavily influenced by activation passed to various behaviors. This activation was influenced by the various feelings or emotions.* Each selection of a behavior includes the generation of an expectation codelet (see the next step).

9) Action taken. The execution of a behavior (goal context) results in the behavior codelets performing their specialized tasks, having external or internal consequences, or both. LIDA is taking an action. The acting codelets also include at least one expectation codelet whose task it is to monitor the action, attempting to bring to consciousness any result of the action, particularly any failure.

Perceptual Learning

Perceptual associative memory (PAM) is implemented in the LIDA architecture as a slipnet, a semantic net with passing activation (Hofstadter and Mitchell 1994). The nodes in the slipnet may represent primitive feature detectors (perceptual codelets), individuals (e.g. a person or particular object), a category (e.g. chair, woman, animal), or a relation (e.g. cup on table). An important aspect of PAM is that regardless of the semantics of a particular item (whether, conceptually, it is a feature detector, an object, a category or an abstract relation or concept) all are represented in the slipnet as nodes of identical structure. Nodes in the slipnet can also represent relations between objects, including spatial, temporal or causal relations. Such abstract relation nodes (e.g. on(cup, table)) must include in its structure placeholders for arguments for its various roles, for example, which cup, which table. Please don't be misled by the symbolism we've just used. These nodes are not symbolic, but are still grounded in reality by their ultimate connection to the primitive feature detectors. They may best be thought of as perceptual symbol simulators in the sense of Barsalou (1999). In this way they may be viewed as templates for structure building in the workspace (preconscious working memory buffers) as in the Copycat architecture (Hofstadter and Mitchell 1994).

An incoming stimulus, say a visual image, is descended upon by a hoard of perceptual codelets (primitive feature detectors). Each of these codelets is looking for some particular feature (a certain color, a line at a particular angle, etc) or more complex features (a T junction, a red line). Upon finding a feature of interest to it, the codelet will activate an appropriate node or nodes in the slipnet. Activation is passed. The slipnet will eventually stabilize. Nodes with activations over threshold, along with their links, are taken to provide the constructed meaning of the stimulus.

Perceptual learning in the LIDA model occurs with consciousness. This learning is of two forms, the strengthening or weakening of the base-level activation of existing nodes, as well as the creation of new nodes and links. Any existing concept or relation that appears in the conscious broadcast (Step 5 of the cognitive cycle) has the base-level activation of its corresponding node strengthened as a function of the arousal of the agent at the time of the broadcast. The base-level activation curve of a slipnet node is modeled along a saturating, sigmoid curve,

A new individual item that comes to consciousness results in a new node being created, together with links into it from the feature detectors of its features. Such a new item gets to consciousness by means of some *new-item attention codelet* that notices a collection of active features in the percept without a common object of which they are features. Such a new item-attention codelet might be looking for such features as spatial contiguity, common motion, and persistence over time. In this latter case, perceptual learning of new objects become somewhat top-down, depending also on local associations from transient episodic memory during prior cognitive cycles. If this attention codelet succeeds in bringing the resulting new item to consciousness, a node for it is created in PAM by the perceptual learning mechanism.

Here's how a new category may be formed. If a *similarity-attention codelet* notices in long-term working memory (see Step 4 of the cognitive cycle) two items with several common features, and succeeds in bringing this similarity to consciousness, a new category is created by the perceptual learning mechanism with *is-a* links into the new category from each of the items.

New relation nodes occur similarly, in a manner suggested by the work of Doumas and Hummel (2005). New relations are learned into nodes in PAM from structures built in the preconscious working memory buffers by perceptual codelets, that instantiate existing relation nodes from PAM and bind objects to their arguments. These new relation nodes are learned when *relation-noting-attention codelets* succeed in bringing the new relations to consciousness, that is when the relation 'pops into mind' or 'occurs to me.' New links are learned along with the nodes. The initial base-level activation of new nodes, be they object, category or relation nodes, are assigned as a function of arousal at the time of the conscious broadcast (Step 5 of the cognitive cycle above).

One may object that all these new nodes and links, sometimes created as often as several times a second, might prove computationally intractable. But nature is often profligate; witness the vast numbers of acorns or sperm produced, so few of which come to any fruition. Here we have another example of such profligacy in perceptual learning. We are saved from computationally intractability by the rapid decay of almost all of the new nodes and links, by virtue of their inverse sigmoid decay curves. Only those new nodes and links that come to consciousness often and/or at high arousal levels have much chance of not quickly decaying away. In the AI literature, a similar mechanism is referred to as *generate and test*. In the LIDA model, perceptual learning generates trial nodes (combined feature detectors, individual items, categories, relations, etc.) and links, and rapidly discards those that don't quickly prove useful.

In keeping with Barsalou's Perceptual Symbol Systems (1999), the nodes and links in LIDA's slipnet form perceptual symbol representations that carry forward throughout the entire architecture, including working memory, episodic memory (with a detour back through perception), long-term working memory, 'consciousness,' procedural memory (the scheme net) and action selection. There are no amodal representations.

Episodic Learning

In the LIDA model memory is hypothesized to interact with conscious events for its normal functioning. Within the context of episodic memory we are concerned with *interpreting the contents of 'consciousness' so as to be able to encode the what, where and when* of each cognitive cycle into episodic memory. LIDA is endowed with two types of episodic memories, one with a small capacity for short term retention of detailed sensory-perceptual information (transient episodic memory, TEM) and the other for long term storage of lifelong events and facts (declarative memory, DM). In the LIDA model, declarative memory (DM) is composed of autobiographical memory, and semantic memory. Autobiographical memories are typically reexperienced in vivid detail when accessed while semantic memories are mainly comprised of fact or belief. Semantic memories typically lack a particular source with a time and place of acquisition. Semantic memories are believed to have lost their association with their original autobiographical source.

Episodic learning in the LIDA architecture results from events taken from the contents of 'consciousness' being encoded in our modified sparse distributed memory (SDM) representing TEM. The major modification of SDM consists of replacing its binary content space with a ternary content space by including a don't-care symbol, '*', while retaining the SDM binary address space. This modification allows more efficient encoding of partially specified events, as well as more efficient recall using partial cues (Ramamurthy, D'Mello, and Franklin 2004).

Each primitive feature detector in the slipnet corresponds to a set of dimensions of the vectors to be stored in SDM. Therefore, the dimensionality of the sparse distributed memory space is roughly equivalent to the number of primitive feature detectors of the agent. Perceptual symbols (slipnet nodes) making up an event in 'conscious' contents are traced back along slipnet links to primitive feature detectors. Using their correspondence with dimensions, a vector is formed and written to transient episodic memory (TEM) implemented as a modified SDM, thus effecting episodic learning within each cognitive cycle. The recall of events (local associations) from TEM and from DM will require routing the read vector through perceptual associative memory so as to recover the corresponding perceptual symbols. This rerouting is suggested by reverse neural pathways from the frontal cortex back to the various anterior sensory cortices, found in human nervous systems (Koch 2004, p. 245).

In addition to the encoding of the sensory perceptual details of each episode manifested through the contents of consciousness, this episodic learning includes the encoding of feelings and emotions, and of actions taken by the agent.

Periodically, and offline, the not yet decayed contents of TEM are consolidated into declarative memory (DM) (autobiographical memory plus semantic memory) which is also implemented as a modified SDM system. Conway stipulates that as an aftermath of the consolidation process, previously volatile events may acquire high stability and durability (2001). This scenario mirrors our view of the still controversial question of how human episodic memory works.

Procedural Learning

Procedural memory in the LIDA architecture bears close similarity to perceptual associative memory. Recall that objects, categories, and relations in perceptual memory were all represented by nodes. In a similar vein, behavior codelets, behaviors, and behavior streams all share the same representation in what we call a *scheme* (motivated by Drescher's schema mechanism (1991)). A scheme consists of an action, together with its context and its result, as well as a base-level activation that estimates the likelihood of that result occurring as a result of taking the action in its context. Each scheme should be thought of as a template for a behavior codelet, a behavior, or a behavior stream. The context of a scheme corresponds to preconditions, its results to post conditions. Pre and post conditions of a scheme are simply nodes in the slipnet that are appropriately grounded to their primitive feature detectors. Just as the primitive feature detectors form the periphery of the slipnet in perceptual associative memory, primitive schemes, that is actions with no specified context or result, constitute the periphery of procedural memory (called the scheme net). These primitive schemes are also referred to as empty schemes. As one moves inwards into the scheme net more complicated schemes are discovered that are templates for behavior codelets executing in parallel (behaviors) or in sequence (behavior streams).

In accordance with global workspace theory, Step 6 of the cognitive cycle serves to recruit internal resources with which to deal with the current situation. Procedural memory, the scheme net, receives the broadcast of the contents of consciousness. Schemes are activated by these contents in proportion to how well the contents coincide with the context of the scheme, and how well the results of the scheme satisfy some current goal as specified by feelings and emotions within the contents. After activation passes in the scheme net and it stabilizes, those schemes that are over threshold are instantiated into behavior codelets, behaviors, or behavior streams in the behavior net. Instantiation includes the binding of variables in instantiated behavior codelets, as well as the assignment of both environmental and motivational activation.

We propose a combination of both *instructionalist* as well as *selectionist* motivated agendas (not discussed here) for procedural learning, with consciousness providing reinforcement to actions. Reinforcement is provided via a sigmoid function such that initial reinforcement becomes very rapid but tends to saturate. The inverse of the sigmoid function that produces the reinforcement curve, serves as the decay curve. Therefore, schemes with low base level activation decay rapidly, while schemes with high (saturated) base level activation values tend to decay at a much lower rate.

Empty schemes, each containing only a primitive action with no context and results, lie at the periphery of the scheme net. Each scheme has two types of activation, a base-level activation and a current activation. The base-level activation measures the reliability of the scheme, that is, the likelihood of its result occurring when the action is taken within its context. The current activation measures the applicability of the scheme to the current situation. For learning to proceed initially, the behavior network must first select the instantiation of an empty scheme for execution. Since an empty scheme has no context, it is assumed to be always applicable to the situation at hand. Before executing its action, the instantiated scheme (behavior codelet) spawns a new expectation codelet. After the action is executed, this newly created expectation codelet focuses on changes in the environment as a result of the action being executed, and attempts to bring this information to consciousness. If successful, a new scheme is created, if needed. If one already exists, it is appropriately reinforced. Conscious information just before the action was executed becomes the context of this new scheme. Information brought to consciousness right after the action is used as the result of the scheme. The scheme is provided with some base-level activation, and it is connected to its parent empty scheme with a link.

Collections of behavior codelets form behaviors. This corresponds to collections of processors forming goal contexts in global workspace theory. The behavior codelets making up a behavior share preconditions and post conditions. Certain attention codelets notice behavior codelets that take actions at approximately the same time, though perhaps in different cognitive cycles. These attention codelets attempt to bring this information to consciousness. If successful, a new scheme is created, if it does not already exist. If it does exist, the existing scheme is simply reinforced, that is, its base-level activation is modified. If a new scheme has to be created, its context is taken to be the union of the contexts of the schemes firing together. The result of the new scheme is the union of the results of the individual schemes. Additionally, this new scheme is provided with some base-level activation and is connected by links to the original schemes it includes. If this composite scheme executes in the future it will pass activation (positive or negative) along these links.

Collections of behaviors, called behavior streams, with activation passing links between them, correspond to goal context hierarchies in global workspace theory. They can be thought of as partial plans of actions. The execution of a behavior in a stream is conditional on the execution of its predecessor and it directly influences the execution of its successor. When an attention codelet notices two behavior codelets executing within some small time span, it attempts to bring this information to consciousness. If successful, it builds a new scheme with links from the first scheme to the second, if such a scheme does not already exist, in which case the existing scheme is simply reinforced. If a new scheme has to be created, its context is the union of the contexts of the first scheme and the second, excluding the items that get negated by the result of the first. Similarly the result of the new scheme formed will be the union of both results, excluding the results of the first that are negated by the result of the second. Using such a learning mechanism iteratively, more complex streams can be built. Again, these newly created schemes can decay and disappear rapidly if they don't prove useful. It's another example of generate and test at work.

Discussion

With the design of three continually active incremental learning mechanisms we have laid the foundation for a working model of cognition that produces a cognitive architecture capable of human like learning. The architecture can be applied to control autonomous software agents as well as autonomous robots "living" and acting in a reasonably complex environment. The perceptual learning mechanism allows each agent controlled by the LIDA architecture to be suitably equipped so as to construct its own ontology and representation of its world, be it artificial or real. And then, an agent controlled by the LIDA architecture can also learn from its experiences, via the episodic learning mechanism. Finally, with procedural learning, the agent is capable of learning new ways to accomplish new tasks by creating new actions and action sequences. With feelings and emotions serving as primary motivators and learning facilitators, every action, exogenous and endogenous taken by an agent controlled with the LIDA architecture is self-motivated.

The LIDA architecture has vast breadth covering many aspects of cognition including perception, memory processes, emotions and action-selection. It is a complex model with many parameters. While it may not be possible to test so many parameters of a working model, we argue that such working models are needed to address real world problems. In such a working model, we have been able to integrate three type of learning modulated by feelings and emotions, illustrating how cognitive science principles can be applied towards the hard problems of AI. Several tasks could be learned concurrently with transfer of knowledge to new tasks. A number of the old, hard AI problems would potentially be solved.

References

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

Baars, B., and Franklin, S. 2003. How conscious experience and working memory interact. *Trends in Cognitive Science* 7:166-172.

Baddeley, A.D., & Hitch, G.J. 1974. Working memory. In G.A. Bower (Ed.), *Recent advances in learning and motivation* (Vol. 8, pp. 47–90). New York: Academic Press.

Barsalou, L. W. 1999. Perceptual symbol systems. *Behavioral and Brain Sciences* 22:577–609.

Brooks, R. A. 1990. "Elephants Don't Play Chess", *Robotics and Autonomous Systems* (6), 1990, pp. 3–15.

Conway, M. A. 2001. Sensory-perceptual episodic memory and its context: autobiographical memory. In *Episodic Memory*, ed. A. Baddeley, M. Conway, and J. Aggleton. Oxford: Oxford University Press.

Doumas, L. A. A., and J. E. Hummel. 2005. A Symbolic-Connectionist Model of Relation Discovery. In Proceedings of the XXVII Annual Conference of the Cognitive Science Society.

Drescher, G. 1991. *Made Up Minds: A Constructivist Approach to Artificial Intelligence*, Cambridge, MA: MIT Press.

Hofstadter, D. R., and M. Mitchell. 1994. The Copycat Project: A model of mental fluidity and analogy-making. In *Advances in connectionist and neural computation theory*, Vol. 2: logical connections, ed. K. J. Holyoak, and J. A. Barnden. Norwood N.J.: Ablex.

Franklin, S. 2001. Automating Human Information Agents. *Practical Applications of Intelligent Agents*, ed. Z. Chen, and L. C. Jain. Berlin: Springer-Verlag.

Franklin, S., B. J. Baars, U. Ramamurthy, and M. Ventura. 2005. The Role of Consciousness in Memory. *Brains, Minds and Media* 1:1-38, pdf.

Jackson, J. V. 1987. *Idea for a Mind*. Siggart Newsletter, 181:23–26.

Kanerva, P. 1988. *Sparse Distributed Memory*. Cambridge MA: The MIT Press.

Koch, C. 2004. *The Quest for Consciousness: A neurobiological approach*. Englewood, Colorado: Roberts & Co.

Maes, P. 1989. How to do the right thing. *Connection Science* 1:291-323.

Ramamurthy, U., D'Mello, S., and Franklin, S.,. 2004. Modified Sparse Distributed Memory as Transient Episodic Memory for Cognitive Software Agents. In Proceedings of the International Conference on Systems, Man and Cybernetics. Piscataway, NJ: IEEE.