Biologically Inspired Cognitive Architectures II:
Papers from the AAAI Fall Symposium (FS-09-01)

The Constructor Metacognitive Architecture

Alexei V. Samsonovich

Krasnow Institute for Advnced Study, George Mason University
4400 University Drive MS 2A1, Fairfax, VA 22030-4444, USA
asamsono@gmu.edu

Abstract

A true human-level learner should be able to deliberately construct its own knowledge, its processes of reasoning resulting in a new knowledge, its system of values and goals, and the scenario of its cognitive growth. These capabilities require a cognitive architecture of a new kind that supports metacognition, self-awareness, and self-regulation. An example architecture design called Constructor is described here. The main distinguishing feature of this architecture is its virtually unlimited self-regulated cognitive growth ability. Other features include metacognition, self-awareness, and an intrinsic embodiment in virtual reality that is used, e.g., for active construction of cognitive and learning processes.

Introduction

The present historical epoch is unique in the sense that now people may have the opportunity to create something equal to them, if not greater: machines capable of human-like intellectual and cultural development. The reason is not only that the hardware available today is compatible in its raw computational capacities with the human brain. The main reason is the emergent understanding of how the human mind works. It appears that implementing the same principles of the human mind in a machine would not take yet unavailable today computer resources.

Since the onset of the research in cognitive modeling, it was understood that a successful approach should be based on integrative cognitive architectures (Newell, 1990). Since then, cognitive architecture designs proliferated extensively (SIGArt, 1991; Pew & Mavor, 1998; Ritter et al., 2003; Gluck & Pew, 2005; Gray, 2007). While today many of the early ideas are forgotten, a mature stream of research has formed in the field, with two well-established dominant paradigms, associated with them communities and descending branches. The two dominant (but not exclusive) practice paradigms are: (a) cognitive modeling aimed at providing an accurate computational account of the human psychology and neuropsychology (mainly associated with ACT-R: Anderson & Lebiere, 1998; Anderson et al., 2004), and (b) efficient task solving in special practical applications, probably more typical of researchers in the Soar community (an example is *Tac-Air Soar*: Jones et al., 1999).

At the same time, the big goal of (c) creating machines that are intellectually comparable to humans (McCarthy et al., 1955) was apparently not so easy to reach. While some early attitudes may seem discredited, one could notice that today the goal is closer than ever. In order to make a progress toward it, we need to understand clearly:

- What is the goal?
- What is wrong with existing approaches?
- What kind of a cognitive architecture do we need?

The present work takes a shot at this target. The author's answer to the first question should be clear from the above and can be formulated concisely as follows: *the goal is to design a human-level learner*. Yet, this statement needs a further clarification. Its limited interpretation could be, e.g.: "The goal of a human-level learner is to take complex, noisy information from multiple modalities and distill this experience into a representation that supports prediction about and manipulation of the world" (Shrobe et al., 2006, p. 11). An alternative, more general interpretation (Samsonovich, 2007) is the view of a human-level learner as a computational embryo of general intelligence capable of *growing* into a human-level-intelligent artifact. This is the interpretation implied in the present work. It may still sound ambiguous, but should become clear below.

Limitations of Existing Approaches

A cognitive architecture is a computational model that describes functional components of an intelligent agent (a cognitive system) and their interactions. Best known examples include Soar (Laird et al., 1986, 1987; Laird & Rosenbloom, 1996; Laird, 2008) and ACT-R (referenced above), and related architectures: e.g., SAL (Lebiere et al.,

2008), EPIC (Meyer & Kieras, 1997), CLARION (Sun, 2004), LIDA (Franklin, 2007), Polyscheme (Cassimatis et al., 2004), etc. All these cognitive architectures as currently used are limited in their abilities to grow cognitively through natural interaction with other agents and environments. Again, the statement needs clarification. Of course, each of these architectures is Turing-complete and in this sense can be programmed to do whatever the Turing machine can do. On the other hand, it would be impossible to take, e.g., an existing ACT-R simulation that is built to solve arithmetic problems and to teach it naturally, like a human child, up to a Ph.D. level. In this sense, *all existing implementations of cognitive architectures lack general-purpose human-level teachability*. They with their current knowledge can only be taught kinds of knowledge that was conceived and potentially enabled in them by designers. In particular, they lack an unlimited ability to construct their own cognition and knowledge from metacognitive perspectives, because they lack a human-like sense of self and they lack true metacognition, as explained below.

"Metacognition" means "cognition about cognition". Nevertheless, when today engineers use this term, they frequently mean something that is external to cognition, a tool that is used to monitor and to moderate cognition and is *not* cognition per se (therefore, is not metacognition). It appears that at present virtually no mainstream cognitive architecture implements principles of a true, general-purpose metacognition, that requires certain cognitive processes to be instantiated in a metacognitive perspective of the agent, from which they would be used to operate on other cognitive processes that belong to other mental perspectives in the same system in real time, treating those processes and mental perspectives as first-class objects.

A clarification by counterexamples is necessary of what is meant by "genereal-purpose". (i) An ACT-R simulation can spawn another ACT-R simulation to simulate the mind of a human partner (Kennedy et al., 2005). In this specific implementation, however, this step was pre-engineered and pre-programmed: the given implementation cannot do a different form of metacognition. (ii) *Instructo-Soar* (Huffman & Laird, 1995) learns from instructions, yet with definite limitations on what can be learned. For example, it cannot learn to do manipulations with its own episodic memories as it does with objects in its world. (iii) Formally speaking, certain databases can be "taught" any knowledge that is expressible electronically; however, they have limited abilities in using their "knowledge". (iv) Numerous demos created for specific challenges (e.g., Breazeal et al., 2005, 2006; Haikonen, 2007) may formally pass tests for higher cognitive functions, yet are clueless outside the selected challenge paradigms.

The difference between these examples and a general-purpose learner is as big as, e.g., the difference between the universal Turing machine and a Turing machine designed to add two numbers. Interestingly, architectures of the two automata look somewhat similar to each other (Denning et al., 1978, section 11.6), and nevertheless the two automata have infinitely different abilities. One of them can only add two numbers – and can do nothing else, while another with the right software can prove all proven theorems about numbers, and do much more (of course, it can do nothing without a software). By analogy, one could say that with modern cognitive architectures we are still at a non-universal stage (even though they all are Turing-complete). Are we yet to discover a universal cognitive architecture? If yes, then what are we missing?

Designers of recent modifications of Soar and ACT-R intend to fill by demos and modules all the "cognitive gaps" that one could point in them: metacognition, theory-of-mind, episodic memory, imagery, "what if" capabilities, emotions, social cognition – in other words, most of those higher cognitive abilities that all simultaneously become unleashed in a 3-to-4 year old child during the so-called *cognitive leap*, when the child develops the familiar to adults sense of self (Moore & Lemmon, 2001). On the other hand, many designers of popular cognitive architectures would agree that their implemented agents lack any sense of self, and they do not consider this a drawback. The reason is that the notions of a *self* and self-awareness are understood in the modern artificial intelligence literature in a limited sense. "The self" may refer to the robot's body, or to the running software, or to the set of variables under homeostatic control by the agent, or to the agent as a whole contrasted with other agents or the environment (some agents have this kind of self). These basic notions of the self play the grounding role with respect to the more elaborate concepts of personhood (Damasio, 1999). The essence of the human sense of 'I' is, in fact, simple and distinct from them: it can be understood as an idealized abstraction of a subject, who is the owner of experiences and the author of volitions (*minimal self*: Gallagher, 2000; *conscious self*: Samsonovich & Nadel, 2005). Unfortunately, it has become a good form in certain scientific communities to deny the relevance of this notion to science (e.g., Sloman, 2008). Yet, it does make sense to consider implementing the same principles in an artifact.

Indeed, this notion of a subject-self plays the key role in self-regulated learning (SRL: Zimmerman, 1990, 2000; Winne & Perry, 2000). SRL is a general, ubiquitous in the human society paradigm of learning guided by meta-cognition, self-motivation and strategic action. SRL has three phases: forethought, performance and self-reflection (Zimmerman, 2000), involving elements like self-motivation, self-orientation, self-analysis, self-control, self-instruction, self-imagery, self-judgment, self-attribution, self-rewarding, etc. Principles of SRL have been studied and used in educational and psychological literature for decades and became recently supported by intelligent tutoring and diagnostic tools (see Azevedo & Witherspoon, 2008; Winne & Nesbit, 2009). SRL proves critical for student academic achievements (Pashler et al., 2007; Zimmerman, 2008), yet there is no mainstream cognitive architecture that is built on the principles of SRL, not to mention the lack of tutoring systems based on such architectures (implemented close examples are again limited: e.g., Kim & Gil, 2007, 2008).

Table 1. Hierarchy of intelligent agent architectures (based on Table 1 in Samsonovich et al., 2009).

Cognitive architecture type and level	The agent is capable of
Metacognitive and self-aware (highest)	Modeling mental states of agents, including own mental states, based on the concept of a self
Reflective (high)	Modeling internally the environment and behavior of entities in it
Proactive, or deliberative (middle)	Reasoning, planning, exploration and decision making
Reactive, or adaptive (low)	Sub-cognitive forms of learning and adaptation
Reflexive (lowest)	Pre-programmed behavioral responses

The informally accepted today hierarchy of cognitive architectures is known in many variants, one of which includes five levels (Table 1). The top, fifth level is virtually empty. Approach based on the GMU-BICA cognitive architecture (Samsonovich & De Jong, 2005) is intended to fill this gap (BICA stands for "biologically inspired cognitive architecture", and reflects a parsimonious integration of computational models of cognition drawn from cognitive science, neuroscience, and artificial intelligence). However, the implemented under the DARPA BICA program rapid prototype of GMU-BICA (Samsonovich et al., 2006) did not include metacognitive mental states (they were nevertheless present in the design). After a recent adaptation of GMU-BICA for the design of an intelligent tutoring system (Samsonovich et al., 2009), it is becoming clear that the original design of GMU-BICA can be further optimized.

The Architecture We Need

This section outlines the design of *Constructor*:[1] a metacognitive architecture inspired by the conscious brain-mind that integrates multiple instances of the self with virtual reality. "The self" here refers to the notion of the minimal subject-self: the unique owner of experience and the author of voluntary actions (Gallagher, 2000; Samsonovich & Nadel, 2005). Multiple instances of the self correspond to mental perspectives (viewpoints) that may differ from each other by the time of experience ('I-Now' vs. 'I-Next'), by the status of the subject (e.g., 'I-Imagined', 'I-Goal'), by the location, the identity of the subject, etc. Constructor is based on the ideas of GMU-BICA and inherits many of its features, including the mental state framework (Samsonovich et al., 2009).

[1]The name reflects the inherent ability of this architecture to construct own cognitive and learning processes from a metacognitive perspective.

From an empirical psychological perspective, it seems plausible that representations of multiple instances of the self and attributed to them mental states co-exist and interact with each other in human working memory (German et al., 2004; Rizzolatti & Craighero, 2004; Samsonovich & Nadel, 2005). Therefore, implementing this feature in a cognitive architecture is likely to make it human-mind-compatible (Samsonovich et al., 2006, 2009). In particular, this feature makes a big difference for a human subject interacting with virtual reality, where multiple instances of the same self can be present simultaneously and may be very different from everyday experiences of self.

The Constructor architecture (Figure 1) is designed to work collaboratively with a human partner in a variety of paradigms: the partner could be a guide to the agent, a student, an instructor, a game player, a designer, etc. All these paradigms presuppose that the human and the artifact share a common task space. Constructor integrates internal (artificial) and external (human) mental states by embedding them into one and the same symbolic representation of this task space: an *intrinsic virtual environment* (Figure 1). This virtual environment provides a symbolic representation of the current cognitive paradigm and is made directly accessible to the human interacting with the artifact. Intrinsic virtual environment is a buffer that serves the functions of an interface, a simulation tool, and a medium where the construction processes take place. All operations in it can be performed cooperatively by the agent and by the human.

Representation Building Blocks

In humans, elements of subjective experiences, or instances of awareness (sometimes called *qualia*, although this term is ambiguous and may be confusing) are private and unique to each subject, and so are their representations by patterns of neuronal activity and other dynamic states in the brain. However, the form, functional characteristics and dynamics of elements of subjective experiences appear to be universal and can be described by *mental categories, mental schemas and mental states*.[2] Here and below, a *category* is understood as a functional token characterizing a certain kind of subjective experience. Examples: "red", "bicycle", "fuzziness", "the opposite one". Each category corresponds to a *schema* understood as a functional model of that kind of experience. A schema can be represented as a graph, the nodes of which represent categories (and therefore correspond to other schemas) or primitives (defined in procedural memory). Further technical details of the *schematism* of GMU-BICA and Constructor can be found in previous works (Samsonovich & De Jong 2005, Samsonovich et al. 2006) and in the Appendix below, and will be presented elsewhere.

[2]The word "mental" is added here to distinguish them from categories, concepts and schemas developed in the subject's mind (and of which the subject becomes aware as of concepts, schemas, etc. rather than feelings). The word "mental" is further omitted, except for "mental states".

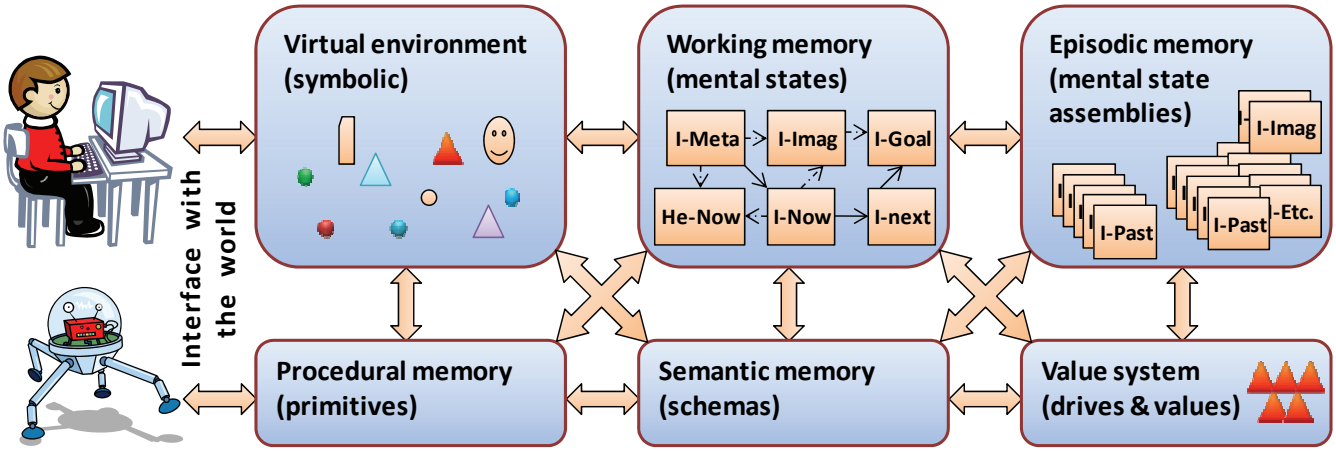


Figure 1. The Constructor architecture. Components include virtual environment, four memory systems (procedural, semantic, working and episodic), and the value system. Arrows show main interactions among components. Interface with the world is provided at the symbolic level (via virtual environment) and at the subsymbolic level (mediated by primitives in procedural memory).

It is interesting to observe that from this point of view, brain representations of schemas and neural correlates of the elements of subjective experiences are the same things. Another representation building block is a *mental state*, which is "a set of instances of schemas attributed as the content of awareness to a unique mental perspective of a subject" (Samsonovich et al., 2009, p. 115). In Constructor, a mental state is interpreted as a functional model of an instance of the self per se rather than its experience or its footprint. Technical details of mental state dynamics are also inherited from GMU-BICA, and their description can be found in the corresponding works (Samsonovich et al., 2006, 2009). One new element is that active mental states are represented by handles in virtual environment.

Architecture Components and Their Functions

In addition to the virtual environment described above, the architecture has five other components, or memory systems: procedural, working, semantic, episodic, and value system, that are briefly described below.

Semantic Memory consists of schemas organized into a semantic net. In addition to this organization, schemas may be allocated as points in an abstract semantic space based on their semantics (e.g., Gardenforth, 2004). All symbolic representations in the virtual environment, in working and episodic memory systems are based on schemas.

Procedural Memory consists of primitives (functions) that are considered sub-symbolic. One set of primitives connect symbolic representations in virtual environment to analog, graphical and textual input-output channels, subserving the corresponding input-output functions. Another (possibly overlapping) set of primitives are used inside schemas as *nodes* (see Appendix): they subserve standard operations (e.g., arithmetic or logical), primitives of control, etc., making schemas similar to LISP functions.

Working Memory consists of instances of schemas that are bound to each other and attributed to particular instances of the self, i.e., organized into mental states, as in GMU-BICA (Samsonovich et al., 2009). In Constructor, elements of working memory can be represented as first-class objects in virtual environment, where they can be analyzed and used by metacognitive mental states to deliberately construct cognitive and learning processes.

Episodic Memory consists of groups of "frozen" mental states that were previously active (i.e., were present in working memory) and may become active again, although in a new for them status of the past. The notion of episodic memory includes not only retrospective memories of actual experiences attributed to the self (Tulving, 1983), but also prospective memories, including plans (Zimmer et al., 2001), and more generally, memories of any imaginary experiences (dreams), not all of which, however, may be remembered.

Value System includes drives and values. A drive is an internal stimulus represented by a number that may reflect certain resources of the system, global and specific measures of the system activities. A drive can cause or facilitate activation of the associated with it schema (e.g., hunger should increase activity of the schema of eating, at least in imagery). Values are associated with dimensions of the semantic space mentioned above.

Illustrative Paradigms

Elements of Self-Regulated Learning. A paradigm of learning how to solve problems that requires metacognition can be as simple as a "yes-no" game. The task for the agent in this game is to explain a given narrative: i.e., to map the narrative by schemas (presumably the agent has a large pool of real-life schemas stored in its semantic memory). The only available actions for the agent are questions, e.g.,

whether a certain schema applies to the narrative. To these questions, only three possible answers can be given by the human participant who knows the story: "yes", "no", and "irrelevant". Of course, there is a potential difficulty with answering arbitrary questions correctly given this constraint, but it can be assumed that the human participant always makes the choice that is closest to the truth.

Given these settings, consider the following example narrative:

> *A man cannot sleep. He gets up, makes a phone call, and when the recipient answers, he hangs up without saying anything. Then he sleeps well.*

The story is incomplete, because there causal relations among its parts are missing. A straightforward, naive strategy in solving the problem is to try one-by-one all potentially applicable schemas, which is exactly what a naive human participant typically does, asking questions like these:

– Did the man worry about his wife? – No.

– Did the man worry about his son? – No.

And the process continues.

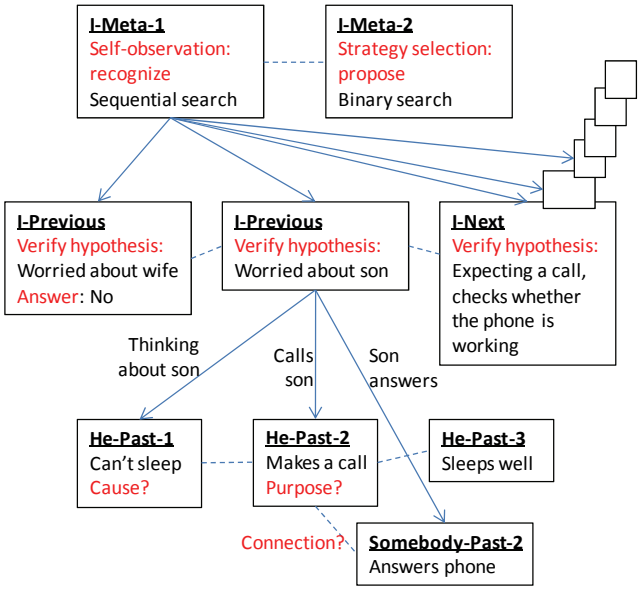


Figure 2. Graphical representation of an expected snapshot of virtual environment containing iconic representations of the metacognitive processes involved in solution of the telephone problem.

The first step at a metacognitive level is to observe what is going on at the cognitive level, by mapping a schema onto the current cognitive process. The answer is that a search is being performed, with elements of the search space being explored one by one. This observation would lead to a prediction that the process may take forever, given the number of potentially applicable schemas and their combinations. Then, e.g., an idea (a schema) pops into the metacognitive mental state: "*A binary search is more efficient than a sequential search*". Applying the schema of a binary search to the given situation means that the agent should use a different strategy at the cognitive level: ask questions that cut the search space nearly in half. These must be very general questions discriminating big categories of logical possibilities, for example:

– Was the factor that kept the man alert external?

– Does the identity/location of the caller/recipient matter?

Given a sufficient set of schemas and the ability to translate between schemas and text, the agent should be able to find the (very short and simple, yet nontrivial) solution of this problem in a reasonable time, using the above strategy.

Cognitive Growth Example. Another paradigm intended to illustrate the constructive cognitive growth ability may consist in exploration, habituation and exploitation of a given nontrivial environment. The agent will use its own intrinsic virtual reality to simulate the given environment together with the agent's own knowledge, skills, values and memories, etc. represented as first-class objects. This unified model of the environment and the agent's mind will be subject to exploration, experimentation and optimization using metacognitive processes.

Validation and Expected Outcomes

The Constructor architecture described above belongs to the top category in Table 1: "metacognitive and self-aware". From the human point of view, this means that the settings described above will result in an illusion of a genuine agency present in the artifact. This expected feature can be validated. There is a close relation between the feeling of self-presence in virtual reality (agency, ownership) and the feeling of another agency present in virtual reality (Herrera et al., 2006). Both phenomena can be measured using "breaks of presence" and "surprise mirror tests" sensitive to key aspects of the minimal self, as explained below.

The feeling of presence in virtual reality is a well-documented phenomenon that is indirectly objectively measurable: by psychometrics, behavioral assessment, etc. A measurement paradigm typically involves detection of *breaks of presence* (Slater & Usoh 1994; Usoh et al. 2000): i.e., experiences of a sudden loss of the illusion of virtual reality. The notion of breaks of presence extends to perception of artificial entities embedded in the environment that pose as "alive conscious beings" (Herrera et al. 2006).

The mirror test (Gallup, 1995) is considered a test for the sense of self in a naive agent. There are, however, multiple possible mechanisms of passing the test, with only few of them involving the notion of a self as an actor. The test may be even much easier to pass with a robot design, when the test paradigm is known in advance (Haikonen, 2007); therefore, this test was not very useful in the past as

a test for artificial human-level intelligence. A mirror test becomes nontrivial and human-hard when unexpected, unusual settings are used in virtual reality: the mirror may show a modified image of the agent's body (or a totally unrelated image, or not an image but some other form of activity); the image or activity may be displaced in space and time, etc. *Surprise mirror tests* of this flavor can be designed to specifically address the notion of the minimal self, and therefore can be used to validate the sense of self in an artifact.

While there is a hope that these tests and measures will establish the superiority of Constructor with respect to modern state-of-the-art cognitive architectures, the big question remains: what is next? The main motivation and the main expected outcome is the practical impact of the new architecture. The hope is that Constructor will enable intelligent agents capable of human-like intellectual and cultural development, because it will clarify and utilize fundamental mechanisms underlying the functioning of the human mind.

Epilogue

This work outlined general principles that can be used to design and build a cognitive architecture of a new kind: with potentially unlimited metacognitive abilities. There is a hope that this relatively small step can lead to a big outcome. To understand why or why not the time now may be favorable to a big change, one needs to take a metascientific perspective and observe that at many levels of natural organization and evolution, one and the same cycle consisting of two alternating modes repeats itself.

Mode 1 (incremental evolution): An open system that is far from equilibrium develops a certain internal ordering by disposing entropy in the environment. This process usually involves selection of the elements of the ordered state. Over time, this process approaches a plateau of organization and entropy production. While the ordering of the system approaches a maximum, the rate of entropy production approaches a minimum (Prigogine, 1955; Prigogine & Nicolis, 1977; Nicolis & Prigogine, 1989).

Mode 2 (leap): The system makes a leap to a new stage of evolution, such that: (i) a qualitatively new internal order emerges starting from parts of the system where the current order does not dominate; (ii) this new order rapidly takes over the old one by re-using and transforming its elements; (iii) the entropy production, the level of evolution and capabilities of the system all rapidly increase, until the process enters Mode 1 and starts approaching a new plateau. This leap phenomenon is called, depending on the domain and the medium in which it occurs, by many names: a phase transition, a catastrophe, an avalanche, a breakthrough, a revolution. Leaps are ubiquitous and can be found at all levels in nature. Examples include phase transitions in disordered systems (e.g., Ziman, 1979), alteration of the species hierarchy (Teilhard de Chardin, 1955), and scientific revolutions (Kuhn, 1962). In the latter example, elements of the ordered state are scientific concepts labeled by the associated with them terms.

The common *leapfrog pattern* observed in evolution of different kinds of systems at different scales of organization can be described by one general principle grounded in statistical physics: *the leap occurs when the system approaches a minimum of entropy production*.

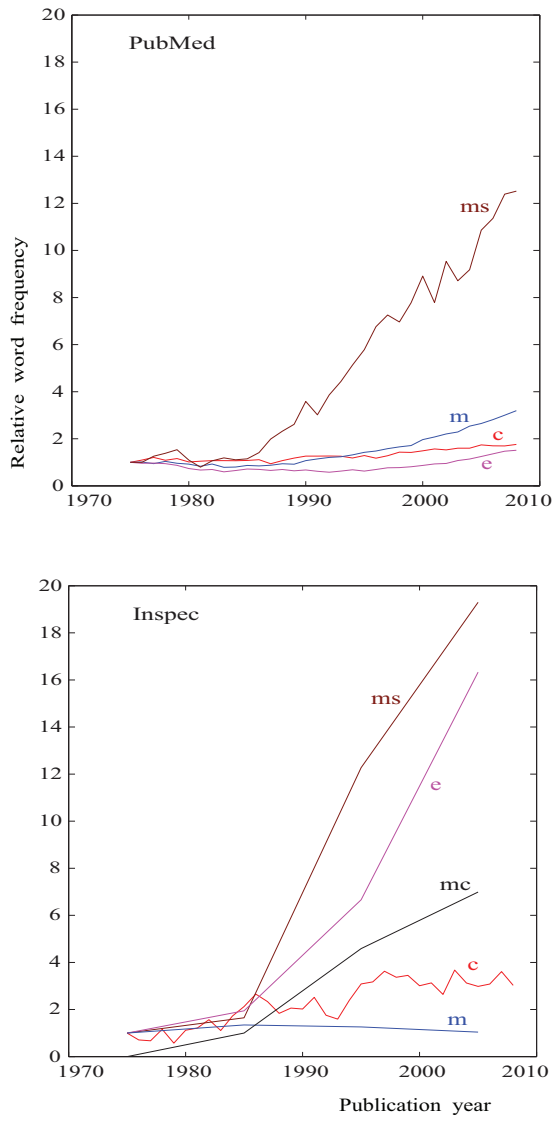
For example, in development of science, the entropy production can be associated with disposal of fallacious concepts and conversion of the corresponding terms into "bad words" that label "bogus notions" (Sloman, 2008, 2009). At the same time, new concepts emerge and new terms enter the lexicon when a leap occurs. Therefore, it seems possible to detect and predict leaps in development of science by measuring the time course of the relative frequencies of scientific terms.

Consider the following example. Cognitive revolution of the middle of the previous century occurred as a leap from behaviorism to cognitive psychology. The essence of the leap was an expansion of the scientific framework by including in it new concepts and new terminology. The notion of internal brain representations was taken seriously and admitted to science, together with words like "mind", "perception", "expectation" and "memory" (Miller, 2003).

In spite of the rapid progress in cognitive psychology, the problem of interpretation of semantics of brain representations was not resolved. A plateau was reached that is characterized by a conceptual vacuum in cognitive sciences and at the same time by the inability to account for phenomena like subjective experience (Chalmers, 2006). Today the connectionist paradigm is exhausted by cognitive modelers, while symbolic cognitive modeling approaches are bound by decades-old limitations of frameworks like ACT-R and Soar. New concepts and new theoretical paradigms are necessary in order to make further progress in the computational science of mind.

It is interesting in this context to look at the frequencies of word usage in the scientific literature over decades (Figure 3). Here, in order to discriminate among scientific domains, two sources were used: PubMed (http://www.ncbi.nlm.nih.gov/pubmed) and Inspec (http://www.engineeringvillage2.com). PubMed is an interface to MEDLINE: a database of academic journals covering life sciences, medicine, biological and biomedical research. In contrast, Inspec covers scientific and technical journals and conference proceedings in physics, electrical engineering and electronics, computing and control, and information technology. The total number of citations from 1970 to present available via PubMed is 16,197,975, which is about two or three times more than with Inspec.

Contrary to some expectations (e.g., Bernard Baars, private communication), results (Figure 3) show that the relative frequency of words like "consciousness" that label hot topics of recent debates is remarkably stable, with only a 75% increase over the last three decades in the biomedical literature. In the engineering literature, the frequency of "consciousness" increased nearly three times since 1975, yet compared to neologisms (not shown in



.

Figure 3. Alteration with time of relative word frequencies in scientific publications. Results are derived from database searches (left: PubMed, right: Inspec) for selected words occurring in any field, performed on September 5, 2009. Curves are labeled according to the search words: c, "consciousness"; m, "memory"; e, "emotion"; ms, "mental state"; and mc, "machine consciousness", for which there are only 3 entries in PubMed (2005, 2008 and 2009). Each data point represents one year in the left panel and one decade (all curves except c) in the right panel. All word frequencies were normalized by the frequencies of the word "normal" (n) at each data point and then uniformly scaled to set the leftmost computed nonzero value to one (each curve was scaled separately). The total search counts for the selected time interval and selected words are, for PubMed: n=1,026,200, e=109,930, c=22,004, m=128,337, ms=7,256, and for Inspec: n=377,931, e=7,474, c=2,250, m=199,629, ms=330, mc=25.

Figure 3) this behavior can be interpreted as approaching a plateau. Similarly, relative frequencies of words "memory" and "emotion" in the biomedical literature demonstrate remarkable stability, and the relative frequency of "memory" in the technical literature did not significantly change at all since the 70s.

There are, however, words and phrases (aside from neologisms) the frequencies of which grow dramatically. E.g., Figure 3 shows a rapid growth of the relative frequency of "emotion" in the technical literature, but not in the biomedical literature, and a similarly rapid growth for "mental state" in both databases. The phrase "machine consciousness" appears likely to gain its popularity in both domains, but the counts are too low to make a judgment (see Figure 3 caption)

Could these emerging popular terms be precursors for the missing concepts in cognitive sciences and in artificial intelligence? Are we witnessing the beginning of a new leap in cognitive and computational sciences? The time will answer these questions. As we can see today, there is a possibility and a necessity for the leap to systems with better learning and cognitive abilities, achievable by enabling general metacognitive capabilities in artifacts.

Conclusions

This work focused on the goal of designing a general-purpose human-level learner. From this point of view, essential limitations of existing approaches based on state-of-the-art cognitive architectures and associated with them paradigms were discussed. The conclusion is that none of the current mainstream cognitive architectures appears to be suitable for designing a true human-level learner, and an architecture of a new kind is necessary.

The Constructor architecture was described in this work as a candidate potentially suitable for implementation of a general-purpose human-level learner. Constructor is a descendant of GMU-BICA developed at GMU under the DARPA IPTO BICA program (terminated in 2006). The key distinguishing features of Constructor include metacognition and self-awareness, which put it on top of the cognitive architecture hierarchy (Table 1). Other distinguishing features include an intrinsic virtual environment which is used by the agent for interface with other agents and for active construction of its own cognitive and learning processes. It is expected that, as a result, the Constructor agent can be teachable at a human level, without obvious intrinsic limitations determined by the engineered paradigms and infrastructure of learning that are characteristic of modern approaches.

Acknowledgments

I am grateful to Drs. Shane Mueller and Frank Ritter for commenting on early versions of the manuscript. I am grateful to Drs. Christian Lebiere, David Noelle, John Laird, and many others for enlightening discussions of BICA. I am grateful to my collaborators in the projects based on GMU-BICA: Drs. Kenneth A. De Jong and

Anastasia Kitsantas. This research is supported by the Center for Consciousness and Transformation, George Mason University.

Appendix: Schematism

The object-oriented paradigm (OOP) is the basis framework for modern software development (e.g., Meyer, 1988), and mainstream cognitive architectures do not make an exception. In general, many frameworks of knowledge representation conceived independently fall into the same paradigm (e.g., frames: Minsky, 1981). OOP is based on a key set of concepts, including notions like object, class, hierarchy, abstraction, encapsulation, inheritance, and modularity (Tracy & Bouthoorn, 1997). At the very core of OOP is the primitive: *Object* → *attribute* → *value*. An object is augmented with a set of attributes to each of which a value can be assigned (a value can be another object). It seems like there is no alternative to this core primitive. The same primitive lies at the basis of representation structures in cognitive architectures, including Soar and GMU-BICA or Constructor. There is, however, a difference between its usage in these cases. Figure 4 illustrates this difference between Soar and Constructor by examples of working memory snapshots.

Figure 4 A shows a semantic-net representation of a snapshot of working memory in Soar with the following content of awareness: "A large orange box (O53) contains a big red ball (O87) and a small red apple (O43) that weighs 200 grams (X44)". Soar represents objects (box, ball, apple) by OOP-objects, their general properties (color, size, etc.) by OOP-attributes, and specific properties of objects (red, big, small, etc.) by OOP-values. There is an alternative to this scheme.

Figure 4 B shows a graph representing a snapshot of a mental state content in Constructor that corresponds to the same content as in Figure 4 A, only without the ball, to simplify the picture: "A large orange box (O53) contains a small red apple (O43) that weighs 200 grams (X44)". As the figure illustrates, Constructor represents all elements – objects, general properties and specific properties – by one and the same class of OOP-objects called categories, or nodes (small circles in Figure 4 B). Each category is associated with its unique schema in semantic memory (not shown), and vice versa: each schema represents a functional model of the category of its head node (red circles in Figure 4 B). A schema is a set of categories, or nodes, possibly linked to each other (there are no examples of internal links in Figure 4 B). Schemas constitute another class of OOP-objects. As an OOP-object, each category (node) in Constructor has a standard set of OOP-attributes. This set is domain-independent, one and the same for all nodes, and includes attributes like category (e.g., "apple"), name (e.g., "apple1"), Id (e.g., "O43"), supercategory list (e.g., "fruit, object"), bindings (represented by arrows in Figure 4 B), and value (to which optionally a scalar or a vector can be assigned).

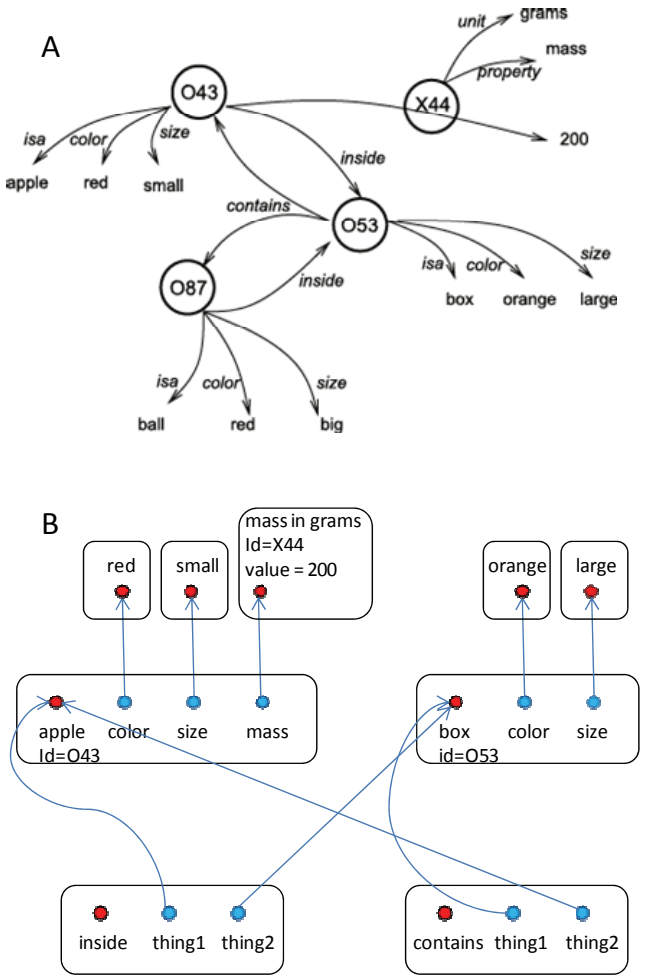


Figure 4. Snapshots of working memory representations in Soar (A) and in Constructor (B). A: a fragment of Figure 3.1 from the Soar manual (Laird & Congdon, 2009), representing a semantic net illustration of four objects in working memory. The state identifier is not shown. Each circle corresponds to an object, each link corresponds to an attribute, and each tip of the arrow points to a value. B: a snapshot of a mental state content in Constructor representing a part of the same content (one object is deleted to simplify the figure). The mental state label and attributes are not shown. Each rounded rectangle represents an instance of a schema, each circle represents a node (a category), and each line represents a binding. The head node in each schema is represented by a red circle, terminal nodes are light-blue.

It is important to notice the difference with Soar. Attributes of objects in Soar do not correspond to attributes of nodes in Constructor, except one case: "isa" in Soar corresponds to "category" in Constructor. Otherwise, in Soar, attributes are domain-specific and represent properties of physical objects, like color, size, mass, and also specific relations among physical objects, like "contains" or "inside". In Constructor, every node regardless of the domain of simulation has one and the

same, standard set of attributes that includes category (isa), name, Id, supercategory list, bindings, value, and many others like perspective, attitude, mode of binding, etc. that are not explained here. The complete set of attributes and their functions in Constructor will be described elsewhere.

Schemas and corresponding to them categories are used in Constructor as universal building blocks to represent all kinds of elements, including those that in Soar correspond to rules and operators. Mental states in Constructor also formally fall into the category of schemas (they are schemas of animate entities, although they are treated differently from the rest of schemas).

References

Anderson, J.R., & Lebiere, C. (1998). The Atomic Components of Thought. Mahwah: Lawrence Erlbaum Associates.

Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. Psychological Review 111 (4): 1036-1060.

Azevedo, R., and Witherspoon, A. M. (2008). Detecting, tracking, and modeling selfregulatory processes during complex learning with hypermedia. In Samsonovich, A. V. (Ed.). *Biologically Inspired Cognitive Architectures: Papers from the AAAI Fall Symposium. AAAI Technical Report* FS-08-04 (pp. 16-26). Menlo Park, CA: AAAI Press.

Breazeal, C., Buchsbaum, D., Gray, J., Gatenby, D., & Blumberg B. (2005). Learning from and about others: Towards using imitation to bootstrap the social understanding of others by robots. In Rocha, L., & Almedia e Costa, F. (Eds.). *Artificial Life* 11 (1-2): 1-32.

Breazeal, C., Berlin, M., Brooks, A., Gray, J., & Thomaz, A.L. (2006). Using perspective taking to learn from ambiguous demonstrations. R*obotics and Autonomous Systems (RAS) Special Issue on The Social Mechanisms of Robot Programming by Demonstration*, 54 (5): 385-393.

Cassimatis, N.L., Trafton, J.G., Bugajska, M.D., & Schultz, A.C. (2004). Integrating cognition, perception and action through mental simulation in robots. Journal of Robotics and Autonomous Systems 49 (1-2): 13-23.

Chalmers, D.J. (1996). *The Conscious Mind: In Search of a Fundamental Theory.* New York: Oxford University Press.

Damasio, A.R. (1999). *The Feeling of What Happens.* New York: Harcourt.

Denning, P.G., Dennis, J.B., & Qualitz, J.E. (1978). Machines, Languages, and Computation. Englewood Cliffs, NJ: Prentice Hall.

Franklin, S., (2007). A foundational architecture for artificial general intelligence. In B. Goertzel & P. Wang (Eds.). Advances in Artificial General Intelligence: Concepts, Architectures and Algorithms. Proceedings of the AGI Workshop 2006. Frontiers in Artificial Intelligence and Applications, vol. 157, pp. 36-54. IOS Press: Amsterdam, The Netherlands.

Gallagher, S. (2000). Philosophical conceptions of the self: implications for cognitive science. *Trends in Cognitive Science* 4: 14-21.

Gallup, G.G., Povinelli, D.J., Suarez, S.D., Anderson, J.R., Lethmate, J., & Menzel, E.W. (1995) Further reflections on self-recognition in primates. *Animal Behavior* 50: 1525-1532.

Gärdenfors, P. (2004). *Conceptual Spaces.* Cambridge, MA: MIT Press.

German, T. P., Niehaus, J. L., Roarty, M. P., Giesbrecht, B., & Miller, M. B. (2004). Neural correlates of detecting pretense: Automatic engagement of the intentional stance under covert conditions. *Journal of Cognitive Neuroscience* 16 (10): 1805-1817.

Gluck, K.A., & Pew, R.W. (Eds.). (2005). *Modeling Human Behavior with Integrated Cognitive Architectures: Comparison, Evaluation, and Validation.* Mahwah, NJ: Erlbaum.

Gray, W. D. (Ed.) (2007). *Integrated Models of Cognitive Systems. Series on Cognitive Models and Architectures.* Oxford, UK: Oxford University Press.

Haikonen, P. O. A. (2007). Reflections of Consciousness: The Mirror Test, in *Proceedings of the 2007 AAAI Fall Symposium on Consciousness*, pp. 67 – 71.

Herrera, G., Jordan, R., & Vera, L. (2006). Agency and presence: A common dependence on subjectivity? *Presence: Teleoperators and Virtual Environments* 15 (5): 539-552.

Huffman, S.B., & Laird, J.E. (1995). Flexibly instructable agents. *Journal Of Artificial Intelligence Research* 3: 271-324.

Jones, R.M., Laird, J.E., Nielsen, P.E., Coulter, K.J., Kenny, P.G., & Koss, F. (1999). Automated intelligent pilots for combat flight simulation. *AI Magazline* 20 (1): 27-41.

Kennedy, W.G., Bugajska, M.D., Adams, W., Schultz, A.C., & Trafton, J.G. (2008). Incorporating mental simulation for a more effective robotic teammate. In Fox, D. & Gomes, C.P. (Eds.). *Proceedings of the Twenty-Third Conference on Artificial Intelligence,* pp. 1300-1305. Chicago, IL: AAAI Press.

Kim, J., & Gil, Y. (2007). Incorporating tutoring principles into interactive knowledge acquisition. *International Journal of Human-Computer Studies* 65 (10): 852-872.

Kim, J., & Gil, Y. (2008). Developing a meta-level problem solver for integrated learners. In Cox, M. T., & Raja, A. (Eds.). *Metareasoning: Thinking about Thinking.*

Papers from the 2008 AAAI Workshop. AAAI Technical Report, vol. WS-08-07, pp. 136-142. Menlo Park, CA: AAAI Press.

Kuhn, T.S. (1962). *The Structure of Scientific Revolutions.* Chicago, IL: University of Chicago Press.

Laird, J. (2008). Extending the Soar cognitive architecture. In Wang, P., Goertzel, B., & Franklin, S. (Eds.). Artificial General Intelligence 2008: Proceedings of the First AGI Conference. Frontiers in Artificial Intelligence and Applications, vol. 171, pp. 224-235. IOS Press: Amsterdam, The Netherlands. ISBN 978-1-58603-833-5.

Laird, J.E. & Congdon, C.B. (2009). *The Soar User's Manual: Version 9.1.* University of Michigan.

Laird, J.E., Newell, A., & Rosenbloom, P.S., (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence* 33: 1-64.

Laird, J.E., & Rosenbloom, P.S. (1996). The evolution of Soar cognitive architecture. In Steier, D.M., & Mitchell, T.M. (Eds.). Mind Matters: A Tribute to Allen Newell. (pp. 1-50). Erlbaum: Mahwah, NJ.

Laird, J.E., Rosenbloom, P.S., & Newell, A. (1986). Universal Subgoaling and Chunking: The Automatic Generation and Learning of Goal Hierarchies. Boston: Kluwer.

Lebiere, C., O'Reilly, R., Jilk, D.J., Taatgen, N., and Anderson, J.R. (2008). The SAL Integrated Cognitive Architecture. In Samsonovich, A. V. (Ed.). Biologically Inspired Cognitive Architectures: Papers from the AAAI Fall Symposium. AAAI Technical Report FS-08-04 (pp. 98-104). Menlo Park, CA: AAAI Press.

McCarthy, J., Minsky, M.L., Rochester, N., & Shannon, C.E. (1955/2000). A proposal for the Dartmouth summer research project on artificial intelligence. In Chrisley, R., & Begeer, S. (Eds.). *Artificial Intelligence: Critical Concepts.* Vol. 2, pp. 44-53. London: Routledge.

Meyer, B. (1988). *Object-Oriented Software Construction.* New York: Prentice Hall.

Meyer, D.E., & Kieras, D.E. (1997). A computational theory of executive cognitive processes and multiple task performance: Part I. Basic mechanisms. Psychological Review 63: 81-97.

Miller, G.A. (2003). The cognitive revolution: a historical perspective. *Trends in Cognitive Sciences* 7: 141-144.

Minsky, M. (1981). A framework for representing knowledge. In Haugeland, J. (Ed.). *Mind Design*, pp. 95-128. Cambridge, MA: The MIT Press.

Moore, C., & Lemmon, K. (Eds.) (2001). *The Self in Time: Developmental Perspectives.* Mahwah, NJ: Lawrence Erlbaum Associates.

Newell, A. (1990). Unified theories of cognition. Cambridge, MA: Harward University Press.

Nicolis, G. & Prigogine, I. (1989). *Exploring Complexity: An Introduction.* New York: Freeman.

O'Reilly, R.C., and Munakata, Y. (2000). Computational Explorations in Cognitive Neuroscience: Understanding the Mind by Simulating the Brain. Cambridge, Massachusetts: MIT Press.

Pashler, H., Bain, P., Bottge, B., Graesser, A., Koedinger, K., McDaniel, M., & Metcalfe, J. (2007). *Organizing instruction and study to improve student learning (NCER 2007-2004).* Washington, DC: National Center for Education Research, Institute of Education Sciences, U.S. Department of Education.

Prigogine I., Nicolis G. (1977). *Self-Organization In Non-Equilibrium Systems.* Wiley.

Prigogine, I. (1955). *Introduction to Thermodynamics of Irreversible Processes.* New York: Interscience Publishers.

Pew, R.W., & Mavor, A.S. (Eds.). (1998). *Modeling Human and Organizational Behavior: Application to Military Simulations.* Washington, DC: National Academy Press. books.nap.edu/catalog/6173.html.

Ritter, F.E., Shadbolt, N.R., Elliman, D., Young, R.M., Gobet, F., & Baxter, G.D. (2003). *Techniques for Modeling Human Performance in Synthetic Environments: A Supplementary Review.* Wright-Patterson Air Force Base, OH: Human Systems Information Analysis Center (HSIAC).

Rizzolatti, G., & Craighero, L. (2004). The mirror-neuron system. *Annual Reviews in Neuroscience* 27: 169-192.

Samsonovich, A.V. & De Jong, K.A. (2005). Designing a self-aware neuromorphic hybrid. In K.R. Thorisson, H. Vilhjalmsson, & S. Marsela (Eds.). AAAI-05 *Workshop on Modular Construction of Human-Like Intelligence. AAAI Technical Report* WS-05-08, pp. 71–78. Menlo Park, CA: AAAI Press.

Samsonovich, A.V. & Nadel, L. (2005). Fundamental principles and mechanisms of the conscious self. *Cortex* 41 (5): 669–689.

Samsonovich, A.V., Ascoli, G.A., De Jong, K.A., and Coletti, M.A. (2006). Integrated hybrid cognitive architecture for a virtual roboscout. In M. Beetz, K. Rajan, M. Thielscher and R. B. Rusu (Eds.). *Cognitive Robotics: Papers from the AAAI Workshop, AAAI Technical Report WS-06-03*, pp. 129-134. Menlo Park, CA: AAAI Press.

Samsonovich, A.V., De Jong, K.A., & Kitsantas, A. (2009). The mental state formalism of GMU-BICA. *International Journal of Machine Consciousness* 1 (1): 111-130.

Shrobe, H.E., Wilson, P.H., et al. (2006). *CHIP: A Cognitive Architecture for Comprehensive Human Intelligence and Performance.* MIT CSAIL Technical Report(http://www.darpa.mil/ipto/programs/bica/docs/MIT-CSAIL.pdf).

SIGArt, (1991). Special section on integrated cognitive architectures. *Sigart Bulletin*, 2(4).

Slater, M., and Usoh, M. (1994). Depth of presence in virtual environments. *Presence-Teleoperators and Virtual Environments* 3 (2): 130–144.

Sloman, A. (2008). "The Self": A bogus concept. Manuscript published online at http://www.cs.bham.ac.uk/research/projects/cogaff/misc/the-self.html

Sloman, A. (2009). An alternative to working on machine consciousness. *International Journal of Machine Consciousness*, in press.

Sun, R. (2004). The CLARION cognitive architecture: Extending cognitive modeling to social simulation. In: Ron Sun (Ed.), Cognition and Multi-Agent Interaction. Cambridge University Press: New York.

Teilhard de Chardin, P. (1955). *Le Phénomène Humain (The Phenomenon of Man).* Paris: Editions du Seuil.

Tracy, K.W. & Bouthoorn, P. (1997). *Object-Oriented Artificial Intelligence Using C++*. New York: Freeman.

Tulving, E. (1983). *Elements of Episodic Memory.* New York: Clarendon Press.

Usoh, M., Catena, E., Arman, S., & Slater, M. (2000). Using presence questionnaires in reality. *Presence-Teleoperators and Virtual Environments* 9 (5): 497-503.

Winne, P.H. & Perry, N.E. (2000). Measuring self-regulated learning. In P. Pintrich, M. Boekaerts, & M. Seidner (Eds.), Handbook of self-regulation (p. 531-566). Orlando, FL: Academic Press.

Winne, P. H., & Nesbit, J. C. (2009). Supporting self-regulated learning with cognitive tools. In Hacker, D.J., Dunlosky, J., and Graesser, A.C (Eds.). *Handbook of Metacognition in Education*. Mahwah, NJ: Erlbaum.

Ziman, J. M. (1979). *Models of Disorder: The Theoretical Physics of Homogeneously Disordered Systems.* Cambridge, UK: Cambridge UP.

Zimmer, H.D., Cohen, R.L.,Guynn, M.J., Engelkamp, J., Kormi-Nouri, R., & Foley, M.A. (Eds.). (2001) *Memory for Action: A Distinct Form of Episodic Memory?* Oxford, UK: Oxford University Press.

Zimmerman, B. J. (1990). Self-regulated learning and academic achievement: An overview. *Educational Psychologist* 25: 3-17.

Zimmerman, B. J. (2000). Attaining self-regulation: A social cognitive perspective. In Boekaerts, M., Pintrich, P. R., and Zeidner, M. (Eds.). *Handbook of Self-Regulation.* pp. 13-39. San Diego, CA: Academic Press.

Zimmerman, B.J. (2008). Investigating self-regulation and motivation: Historical background, methodological developments, and future prospects. *American Educational Research Journal* 45 (1): 166-183.