Computational Constructivist Model as an Anticipatory Learning Mechanism for Coupled Agent–Environment Systems

Filipo Studzinski Perotto • Constructivist AI Research Group, France • filipo.perotto/at/ufrgs.br

> Context • The advent of a general artificial intelligence mechanism that learns like humans do would represent the realization of an old and major dream of science. It could be achieved by an artifact able to develop its own cognitive structures following constructivist principles. However, there is a large distance between the descriptions of the intelligence made by constructivist theories and the mechanisms that currently exist. **> Problem** • The constructivist conception of intelligence is very powerful for explaining how cognitive development takes place. However, until now, no computational model has successfully demonstrated the underlying mechanisms necessary to realize it. In other words, the artificial intelligence (AI) community has not been able to give rise to a system that convincingly implements the principles of intelligence as postulated by constructivism, and that is also capable of dealing with complex environments. **> Results** • This paper presents the constructivist anticipatory learning mechanism (CALM), an agent learning mechanism based on the constructivist approach of AI. It is designed to deal dynamically and interactively with environments that are at the same time partially deterministic and partially observable. CALM can model the regularities experienced in the interaction with the environment, on the sensorimotor level as well, as by constructing abstract or high-level representational concepts. The created model provides the knowledge necessary to generate the agent behavior. The paper also presents the coupled agent environment system (CAES) meta-architecture, which defines a conception of an autonomous agent, situated in the environment, embodied and intrinsically motivated. **> Implications** • The paper can be seen as a step towards a computational implementation of constructivist principles, on the one hand suggesting a further perspective of this refreshing movement on the AI field (which is still too steeped in a behaviorist influence and dominated by probabilistic models and narrow applied approaches), and on the other hand bringing some abstract descriptions of the cognitive process into a more concrete dimension, in the form of algorithms. **> Constructivist content** • The connection of this paper with constructivism is the proposal of a computational and formally described mechanism that implements important aspects of the subjective process of knowledge construction based on key ideas proposed by constructivist theories. **> Key words** • **Factored partially observable Markov decision process (FPOMDP), computational constructivist learning mechanisms, anticipatory learning, model-based learning.**

Introduction

« 1 » The constructivist approach to artificial intelligence can be defined as the set of works on this science directly or indirectly inspired by ideas coming from the constructivist conception of intelligence. This conception was essentially defined by Jean Piaget (1954) and gave raise to an important school of thought that influenced many scientific fields from the second half of the twentieth century onward. The first important AI system based on constructivist concepts appeared much later, presented by Gary Drescher (1991), but even if his model had some theoretical impact on the field of AI, it could never be used to solve significant applied problems. Since then, year after year, new papers have been published that attempt to point out a way to implement such a strong mechanism (Guerin 2011). However, the constructivist approach has never thrilled most researchers in the AI community, staying in that uncomfortable position between the promise of true intelligence and the lack of impressive results.

« 2 » In this article, we present the *constructivist anticipatory learning mechanism* (CALM), an agent learning mechanism based on the constructivist approach of AI. CALM is designed to discover regularities in partially deterministic environments: it identifies the deterministic transformations present in non-deterministic situations. The mechanism operates incrementally: the agent learns at the same time as it needs to interact with the environment. CALM can also deal with partially observable environments: it is able to infer the existence of hidden or abstract properties, integrating them in its anticipatory cycle. The constructed anticipatory model can be used by the agent to optimize its action policy, improving its performance on its own activities and adapting its behavior to the experienced reality according to self-determined goals or internal motivations.

« 3 » The elementary piece of knowledge used by the mechanism is the *schema*, an anticipatory structure that can be described in the form of a conjunctive implication: $context \wedge action \rightarrow expectation$. It represents the prediction of experiencing some transformation when a given action is carried out in a given context.

« 4 » CALM is the cognitive engine embedded in an artificial agent. In order to complete the description of our constructivist computational model, in this article we present the *coupled agent environment system* (CAES), a meta-architecture that defines the agent as an autonomous entity, situated in the environment, embodied and intrinsically motivated.

« 5 » Our work aims to constitute one more brick in the effort to bridge the gap between the insightful but too abstract descriptions of intelligence made by constructivist theories and robust artificial intelligence mechanisms able to implement them.

Sensorimotor to symbolic

« 6 » The gradual development of a symbolic intelligence over a sensorimotor intelligence is an essential aspect of explaining how human beings can render intelligible their experiences, giving some sense to the world, and learning to interact with it (Piaget 1954). The challenge is the same for a situated artificial agent (such as a robot), who needs to learn incrementally the regularities observed throughout its interaction with the environment where it is inserted.

« 7 » The experienced reality is something subjective and should not be confused with an external, objective, ontological universe, which is assumed to be on the other end of the interaction interface. The world as it is cannot be apprehended outside the domain of experience; whatever may lie beyond sensorial perception is inaccessible (Glasersfeld 1974, 1979). Any representation of the outside reality will be a model necessarily based on regularities extracted from subjective sequences of observations and actions, and not from the structure of that reality, which remains unknown in its essence.

« 8 » Moreover, in complex environments, special "macroscopic" properties emerge from the functional interactions of "microscopic" elements, and such emergent characteristics are not defined in any of the sub-parts that generate them (Goldstein 1999). The salient phenomena in this kind of environment tend to be related to high-level objects and processes (Thornton 2003). In this case, if we suppose the existence of a complex universe out there, it is plainly inadequate to represent the experience only in terms of primitive sensorimotor elements (Drescher 1991).

« 9 » Considering these conditions, an intelligent agent (human or artificial) must have the capacity to overcome the limits of pure sensorial perceptions, organizing the universe in terms of more abstract concepts. The agent needs to be able to detect high-level regularities in the dynamics of the environment, but this is not possible if the agent is stuck in a rigid *representational vocabulary*.[1] In a constructivist approach, cognitive development must be a process of gradual complexification of the intelligence, where more abstract structures (symbolic) are built from simpler sensorimotor interactions, in a way that harmonizes the lived experiences with the constructed model.

« 10 » From the flat, unstructured, continuous flow of perceptions resulting from the situation of the agent in a complex universe, intelligence needs to build some organization. While the constructed internal knowledge might reflect an external reality to some degree, from the agent's perspective this remains undecidable. Importantly, though, intelligence progressively organizes knowledge in increasingly abstract structures, enriching the agent's understanding of its own experiences.

1 | The agent's representational vocabulary is the set of elements it can manipulate to create knowledge.

Situativity, embodiment and intrinsic motivation

« 11 » A given universe (natural or computationally simulated) is a whole system that can be analytically separated into two different entities: an *agent* and an *environment*. These two entities can be defined as mutually dependent *dynamical systems*,[2] partially open to each other, and continually deforming their trajectories (Beer 1995, 2004; Barandiaran & Moreno 2006; Ashby 1952).

« 12 » A situated agent (Wilson & Clark 2008) is an entity embedded in an environment. Due to the fact that the agent is only one among many forces that generate the environment dynamics, it is only partially capable of transforming the environment by its actions. In the same way, due to the fact that the agent's sensorial perception is limited in some manner, the environment becomes only partially observable and the agent can find itself unable to distinguish between differing states of the world (Suchman 1987). The same situation can be perceived in different forms, and different situations can have a similar appearance. This ambiguity in the perception of states, also referred to as *perceptual aliasing*, has serious effects on the ability of most learning algorithms to construct consistent knowledge and stable policies (Crook & Hayes 2003).

« 13 » The agent is embodied (Anderson 2003; Ziemke 2003): it presents internal states and metabolisms, elements that belong neither to the mind nor to the environment. This characteristic allows the agent to have intrinsic motivations: evaluative signals related to the internal state of the agent, and not to external environmental states to be reached.

2 | A dynamical system consists of an abstract state space evolving over time according to a rule that specifies the immediate future state given the current state.

Coupled agent–environment system

« 14 » CAES is a meta-architecture proposed in this article to define a coupled agent-environment system, respecting the notions described in the precedent section. The *universe* (U) is represented as a global system $U=\{A, E\}$, where an *agent* (A) interacts with an *environment* (E). The agent $A=\{B, M\}$ is formed by two subsystems: *body* (B) and *mind* (M). The body is the intermediary between mind and environment. Mind, body and environment can be each described by an abstract state space and an evolution function: $E=\{X_E, f_E\}$, $B=\{X_B, f_B\}$, $M=\{X_M, f_M\}$.

« 15 » These entities are interrelated dynamical systems. The environment continually imposes a *situation* (s) on the agent, which responds through an *actuation* (a). The situation is given in function of the state of the environment, $s=f_s(x_E)$, and the actuation is defined according to the state of the body, $a=f_a(x_B)$. In the same way, the mind is continually receiving a perception signal coming from the body in function of its state, $p=f_p(x_B)$, and sending to the body a *control* signal (c), decided in function of the mind's own internal state, $c=f_c(x_M)$. Part of the situation can be perceived by the mind through *external sensors* present in the body, while the mind can also control part of the actuation over the environment through *external effectors* also present in the body. The interaction of the mind with the body takes place through *internal sensors* and *effectors*. The mind does not know *a priori* what sensors and effectors are internal or external. From the point of view of the mind, both body and environment are in some way external, being part of an *exteriority* $W=\{B, E\}$, the world outside the mind. The complete CAES meta-architecture is presented in Figure 1.

« 16 » This configuration generates a kind of circularity, and defines each entity as a partially open dynamical system. The environment evolves in function of its own current state, but influenced also by the actuation coming from the agent, $x_E'=f_E(x_E, a)$. Similarly, the next body state is defined in function of the actual body state, but is influenced by both the situation coming from the environment and the control signal coming from the mind, $x_B'=f_B(x_B, s, c)$. It is the same for the mind, which continually changes its internal state (whatever that means) influenced by its perceptions, $x_M'=f_M(x_M, p)$.

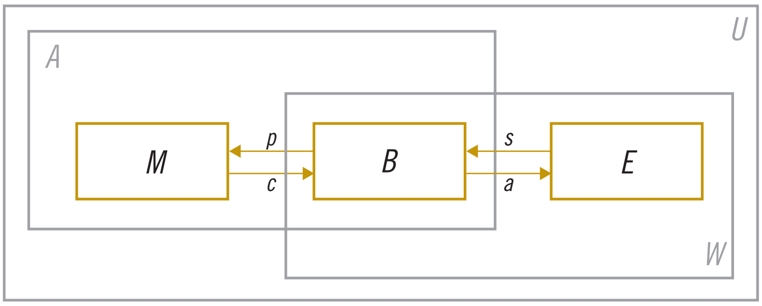

Figure 1: Body between mind and environment.

« 17 » CAES is a meta-architecture because it does not define of what or how each system is made. Moreover, it does not constraint these systems as *stationary*.[3] The environment as well as the body can change its respective set of rules and variables over time. The same applies for the mind, which needs to be non-stationary if we want to have some kind of learning or mental development. Such learning ability can be defined as a function $M'=f_\mu(M, x_M, p)$ that changes the mind's own space of states (creating new concepts or representational signs) and rules (changing the policy of actions that is responsible for determining the control signal) based on the experience (memories and immediate perceptions).

Representing ontological and experiential reality

« 18 » In our understanding, a constructivist machine learning mechanism must be made using *model-based*[4] methods. The agent constructs knowledge in order to understand its experience of interaction with the environment. Computationally, the learning problem can be divided into two parts: (a) the construction of the model, and, based on it, (b) the definition of a policy of actions, which defines the agent's subsequent behavior.

« 19 » When, for simplicity, we say that the agent constructs a *model of the world*, we need to specify that in fact it is the agent's mind that constructs a model of an exteriority (the world outside the mind) to which the mind has access only through a limited sensorial interface. A model of the world is not a reproduction of the structure of an ontological reality, but is a model of the agent's *experiential history*.[5] It is a model (and not a memory) because it does more than remember the past interactions: the model aims to generalize a complete system to represent the whole external world based on the finite set of experiences.

« 20 » Frequently in the machine learning literature, the relation between agent and environment is not clearly defined. Traditionally, computer scientists do not make an explicit difference between the world as it is ontologically and the world represented by the agent at the limits of its sensorial interface and history of interactions. They conceive the agent as acting and perceiving directly on the "real world," and this can give rise to confusing architectures, where situativity problems disappear by omission.

« 21 » We define the learning problem in the following terms: we cannot know what any "external reality" (the world outside the mind) consists of, but we suppose that it can be represented (for analytical purposes) as a *factored and partially observable Markovian decision process* (FPOMDP), where actions

3| A dynamical system is stationary if the rules that define its evolution do not change over time.

4| In opposition to *model-free* methods, where an agent can dynamically optimize its behavior only based on the immediate experience.

5| The experiential history is the sequence of interactions (perceptions and actions) realized by the mind.

and observations correspond to the control and perception signals in the CAES architecture. The world model constructed by the mind can be represented by a *factored Markovian decision process* (FMDP) that constitutes a kind of morphism of the first one, constrained by the sensorial interface limitations, as well as by the incompleteness of the experience, but possibly enriched with abstract variables created in order to make the system more structured and intelligible.

« 22 » In simulated systems, where both agent and environment are programs running in a computer, an observer can have access to the whole structure (mind, body, environment and their interfaces of interaction). In this particular case, it is possible to analyze the factors that characterize the experiential relation with that given reality. The specificities of that relation, combined with the intellectual and cognitive capacities of the agent, will determine the difficulty of learning a *successful model*,[6] and consequently the agent's possibility to become adapted to the environment.

MDP framework

« 23 » *Markovian decision processes* (MDPs) and their extensions constitute widely-used representations for modeling *decision-making* and *planning* problems (Feinberg & Shwartz 2002). An MDP is typically represented as a discrete stochastic *finite state machine* (Puterman 1994; Rivest & Schapire 1994): at each time step the machine is in some state s; the agent interacts with the process by choosing some action a to carry out; then the machine changes into a new state s' and gives the agent a corresponding reward r; a given transition function δ defines the probabilities of the state change according to s and a. The flow of an MDP (the transition between states) depends only on the system's current state and on the action taken by the agent at the time. After acting, the agent receives an evaluative reward signal (positive or negative), according to the chosen actions or the realized state transition.

6| A model can be considered successful if it allows the agent to make correct anticipations for future interactions; it is not an evaluation of the correspondence with ontological structures, which remain inaccessible.

« 24 » Solving an MDP means finding the optimal (or near-optimal) policy of actions in order to maximize the rewards received by the agent over time. When the MDP parameters are completely known, including the reward and the transition functions, it can be mathematically solved by *dynamic programming* methods. When these functions are unknown, the MDP can be solved by *reinforcement learning* methods, designed to learn a policy of actions on-line, i.e., at the same time that the agent interacts with the system, by incrementally estimating the utility of state-actions pairs and then mapping situations to actions (Sutton & Barto 1998).

« 25 » However, the MDP supposes that the agent has complete information about the state of the environment. A partially observable MDP (POMDP) (Singh et al. 2003; Cassandra, Kaelbling & Littman 1998) is an extension of the model that includes a set of observations that is different from the set of states. The underlying system state s cannot be directly perceived by the agent, which has access only to an observation o given by an observation function γ. We can represent a larger set of problems using POMDPs rather than MDPs, but the methods for solving them are computationally even more expensive (Hauskrecht 2000).

« 26 » For a situated agent, this kind of representation becomes inadequate because it requires the complete enumeration of the states, and the number of states increases exponentially according to the number of agent sensors (Bellman 1957). This is the main bottleneck in the use of MDPs or POMDPs: representing complex universes entails an exponential increase in the state space, and the problem quickly becomes intractable.

Factoring the MDP states

« 27 » When a large MDP has a significant internal structure, it can be modeled compactly; the factorization of states is an approach to exploit this characteristic (Boutilier, Dearden & Goldszmidt 2000; Jonsson & Barto 2005; Degris, Sigaud & Wuillemin 2006; Shani et al. 2008). In the factored representation, a state is implicitly described by an assignment to some set of state variables. Thus, a complete explicit state space enumeration is avoided, and the system can be described referring directly to its variables. The factorization of states enables the system to be represented in a generalized and compact way, even if the corresponding MDP is exponentially large (Guestrin et al. 2003). When the structure of the FMDP is completely known, it is possible to find good policies in an efficient way (Guestrin et al. 2003). However, the research concerning the discovery of the structure of an underlying system from incomplete observation is still incipient (Degris & Sigaud 2010).

« 28 » An FPOMDP is an FMDP that can represent partial observation (Guestrin, Koller & Parr 2001; Hansen & Feng 2000; Poupart & Boutilier 2004; Shani, Brafman & Shimony 2005; Sim et al. 2008). An FPOMDP can be formally defined as a 4-tuple $\{X, C, R, T\}$. The state space is factored and represented by a finite non-empty set of system properties or variables $X=\{X_1, X_2, \ldots X_n\}$, which is divided into two subsets, $X=P \cup H$, where the subset P contains the observable properties (those that can be accessed through the agent's sensory perception), and the subset H contains the hidden or non-observable properties. Each property X_i is associated to a specified domain, which defines the values the property can assume; $C=\{C_1, C_2, \ldots C_m\}$ represents the controllable variables, composing the agent actions; $R=\{R_1, R_2, \ldots R_k\}$ is a set of (factored) reward functions, in the form $R_i: P_i \rightarrow \mathbb{R}$; and $T=\{T_1, T_2, \ldots T_n\}$ is a set of transformation functions, such as $T_i : X \times C \rightarrow X_i$, defining the system dynamics. Each transformation function can be represented by a *dynamic Bayesian network*, which is an acyclic, oriented, two-layer graph. The first layer nodes represent the environment state at time t, and the second layer nodes represent the next state, at $t+1$ (Boutilier, Dearden & Goldszmidt 2000). A policy π is a mapping $X \rightarrow C$ where $\pi(x)$ defines the action to be taken in x. The agent must learn a policy that optimizes the average rewards received over time, but it never sees the ontological state x, only a perceptive situation p.

« 29 » When the agent is immersed in a system represented as an FPOMDP, the complete task for its anticipatory learning

FILIPO STUDZINSKI PEROTTO
received his doctorate degree from the University of Porto Alegre (UFRGS, Brazil) in cooperation with the University of Toulouse (INPT, France) in 2010, treating the subject of constructivist artificial intelligence models. He works with applied AI for industry, and participates in the constructivist AI research group of Toulouse.

mechanism is both to create a predictive model of the world dynamics (i.e., inducing the underlying transformation function of the system) and to define an optimal (or sufficiently good) policy of actions in order to establish a behavioral strategy. A good overview of the use of this representation in AI, referring to algorithms designed to learn and solve FMDPs and FPOMDPs, can be found in (Sigaud et al. 2009; Degris & Sigaud 2010).

Relation between ontological and experiential reality

« 30 » We distinguish four main factors that shape the relation between the agent's mind and the external world: *observability*, *complexity*, *determinism* and *controllability*.

« 31 » The *observability* factor (ω) indicates the degree of access that the agent has to the environment state through its sensorial perception. We can imagine this measure as being equivalent to the proportion of observable variables in the whole system in relation to the total number of variables. If the state of the environment can be represented by n bits of information and the state of the sensors affected by that world state can be represented by m bits, the observability factor ω is the proportion of m over n, where $0 \leq \omega \leq 1$. Considering an FPOMDP composed of binary variables, $m = |P|$ and $n = |X|$.

« 32 » If $\omega = 1$, the environment is said to be completely observable, which means that the agent has sensors to observe directly all the properties of the environment. In this case there is no perceptual confusion, and the agent always knows the current state. When $\omega < 1$, the environment is said partially observable. The lower ω is, the higher the proportion of hidden dimensions of the environment is in relation to the agent's perception. When ω is close to 0, the agent is no longer able to identify the current situation only in terms of its perception.

« 33 » The *complexity* factor (φ) is related to the rules that define the world dynamics, indicating how intelligible the environment transformations can be for the agent. The complexity can be measured as the average amount of information needed to define the evolution of one bit in the world state. In a highly structured world, it is possible to model precise causes for each transformation; in other words, the evolution of one variable of the system depends on only a few other relevant variables. In contrast, in an unstructured world there is too much interdependence between the variables to determine the evolution of the system.

« 34 » The difficulty for the agent in constructing a model is related to the complexity of the world dynamics. A less complex world can be more easily structured by intelligence. A low level of complexity means that the information about the dynamics of the environment is concentrated in the variables. It indicates the average amount of relevant variables necessary to describe each transformation. When φ is small, the rules that govern the dynamics of the whole system have few parameters. It is a kind of thermometer indicating how easy is to model causality between events. In contrast, a higher level of complexity (rising to n) indicates that the information about the dynamics is sparsely distributed over all the set of variables, and in this case the agent needs to describe the transformations in function of almost all the variables.

« 35 » The *determinism* factor (∂) is equivalent to the proportion of deterministic transformations in relation to the total number of transformations. In the completely non-deterministic case ($\partial = 0$), all transformation functions (of every property) need to be represented in terms of probabilities. On the other hand, in the completely deterministic case ($\partial = 1$), every transformation is deterministic. An environment is said partially deterministic if it is situated between these two extremities ($0 < \partial < 1$) presenting both deterministic and stochastic transformations.

« 36 » Observability and determinism are dependent factors. Partially observable environments can present some determinant variables to a good world model that cannot be directly perceived by the agent sensors. Such environments can appear arbitrarily complex and non-deterministic on the surface, but they can actually be deterministic and predictable with respect to unobservable underlying elements (Holmes & Isbell 2006). In other words, an ontologically deterministic world can be experienced as non-deterministic. The more an agent has sensors to perceive complex elements and phenomena, the more the environment will appear deterministic to it.

« 37 » Finally, the *controllability* factor (κ) represents the proportion of variables whose dynamics are influenced by the agent's actions, within the total number of variables in the system. The controllability factor affects the difficulty of learning because it determines the capacity of the agent to experiment actively.

CALM

« 38 » The *constructivist anticipatory learning mechanism* (CALM), detailed in (Perotto 2010), is a mechanism that enables an agent to learn the structure of an unknown environment in which it is situated through observation and experimentation, creating an anticipatory model of the world. CALM operates the learning process in an *active* and *incremental* way. There is no separated previous training time: the agent has a single uninterrupted interactive experience within the system; it needs to perform and learn at the same time.

« 39 » The task becomes harder because the environment is only partially observable and partially deterministic, from the point of view of the agent, constituting an FPOMDP. In this case, the agent has perceptive information from a subset of sensory variables, but the system dynamics also depends on another subset of hidden variables. To be able to create a consistent world model, the agent needs, beyond discovering the regularities of the phenomena, also to create abstract variables in order to take into account non-observable conditions that are necessary to understand the system's evolution. In other words, learning a model of the world is more than describing the environment dynamics (the rules that can explain and anticipate the observed transformations), it is also discovering the existence of hidden properties (once they influence the evolution of the observable ones) and, finally, finding a way to deduce the values of these hidden properties.

« 40 » The system as a whole is in fact an FPOMDP, but CALM is designed to discover the existence of non-observable properties, integrating them in its anticipatory model. In this way CALM can infer a structure to represent the dynamics of the system in the form of an FMDP (if the agent can successfully discover and describe the hidden properties of the FPOMDP that it is dealing with, then the world becomes treatable as an FMDP because the hidden variables become known). There are some algorithms able to calculate efficiently the optimal (or near-optimal) policy, when the FMDP is given (Guestrin et al. 2003). The algorithm to calculate the policy of actions used by CALM is similar to that presented by Degris, Sigaud & Wuillemin (2006). However, the main challenge is to discover the structure of the problem based on the on-line observation.CALM does it using representations and strategies inspired by Drescher (1991).

Representing predictive knowledge by schemas

« 41 » CALM tries to reconstruct, by experience, each system transformation function T_i, representing it by an *anticipatory tree*. Each anticipatory tree is composed of pieces of predictive knowledge called *schemas*; each *schema* represents some perceived regularity occurring in the environment by associating context (sensory and abstract), actions and expected results (anticipations).

« 42 » One important strategy for dealing with complexity is finding what is important to anticipate. At the beginning, the only interesting variables are those associated to positive or negative affective values. Staying focused on these variables avoids wasting energy by creating models that anticipate other non-important variables. Gradually, the variables needed to anticipate the evolution of some important variable (relation of causality) are also considered important, and the mechanism will seek to model their transformation function too.

« 43 » A schema is composed of three vectors, in the form

$$\Xi = \{context \wedge action \rightarrow result\}$$

denoting a kind of predictive rule. The *context* vector has their elements linked both with the agent sensors and with the abstract variables. These abstract variables are represented by (mentally created) "synthetic elements" not linked to any sensor but referring to non-sensory properties of the universe, the existence of which is inferred by the mechanism. The action vector is linked with the agent effectors. Context and action vectors can represent sets of equivalent situations or actions, by generalization. The *result* vector represents the value expected for some variable in the next time, after executing the given action in the given context. Each element vector can assume any value in a discrete interval defined by the respective variable domain.

« 44 » Some elements in these vectors can take an "undefined value." For example, an element linked with a binary sensor must have one of three values: true, false or undefined (represented, respectively, by "1", "0" and "#"). The undefined value generalizes the schema because it allows some properties to be ignored in order to represent a set of situations. The learning process happens through the refinement of the set of schemas. After each experienced situation, CALM updates a generalized episodic memory, then checks whether the result (context perceived at the instant following the action) conforms to the expected result of the activated schema. If the anticipation fails, the error between the result and the expectation serves as parameter to correct the model. The context and action vectors are gradually specialized by differentiation, adding each time a new relevant feature to identify the situation class more precisely.

« 45 » The use of undefined values makes it possible to construct an *anticipatory tree*. Each node in that tree is a schema, and relations of generalization and specialization guide its topology (quite similar to decision trees or discrimination trees). The root node represents the most generalized situation, in which the context and action vectors are completely undefined. Each level added to the tree represents the specialization of one element, where each branch replaces the undefined (generalized) value with one different possible defined value. This specialization occurs either in the context vector or in the action vector. In this way, CALM divides the state space according to the different expected results, grouping contexts and actions with their respective transformations. The tree evolves during the agent's life, and is used by the agent, even if the tree is still under construction, to take its decisions, and in consequence, to define its behavior. The structure of a schema (the elementary piece of knowledge of an anticipatory tree) is presented in Figure 2.

« 46 » The context in which the agent is at a given moment (perceived through its sensors) is applied in the tree, exciting all the schemas that have a compatible context vector. This process defines a set of excited schemas, each one suggesting a different action to take in the given situation. CALM will choose one action to activate and will perform it through the agent's effectors. The algorithm always chooses the compatible schema that has the most specific context, called decider schema, which is the leaf of a differentiated branch. This decision is taken based on the calculated utility of each possible choice. There are two kinds of utility: the first estimates the discounted sum of rewards in the future following the policy, the second measures the exploration benefits. The utility value used to take the decision depends on the circumstantial agent strategy (exploiting or exploring). The mechanism also has a kind of generalized episodic memory, which represents (in a compact form) the specific and real situations experienced in the past, preserving the information necessary to correctly construct the tree.

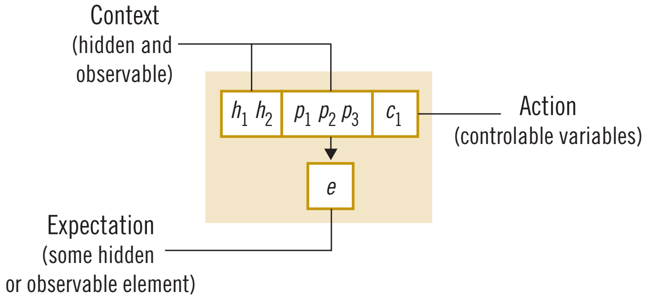

Figure 2: The anticipatory tree. Each node is a schema composed of three vectors: context, action and expected result; the leaf nodes are decider schemas.

Anticipatory tree construction

« 47 » The learning process happens through the refinement of the set of schemas. At each given moment in the time, the set of schemas of our agent, gradually constructed by the mechanism, is assumed to be coherent with all the past experience, describing in an organized way the regular phenomena observed during the interaction with the universe. To do so, the mechanism must have a memory of the past situations, but this memory can be neither too precise (because remembering all the experienced episodes would require a nonviable amount of space) nor too simple (because the lack of information would make it impossible to revise the model if there was contradiction with new disequilibrating observations). The implementation of a feasible episodic memory is not evident; it can be very expensive if we try to stock too much information coming from the sensory flow. However, using some strong but well-chosen restrictions (such as limiting the dependency analysis between variables), and using a generalized and structured representation of the past experience, it becomes computationally viable.

« 48 » After each experienced situation, CALM actualizes the generalized episodic memory and checks whether the result (context perceived at the instant following the action) is in conformity to the expectation of the activated schema in the anticipatory tree. If the anticipation fails, the error between the result and the expectation serves as a parameter for correcting the model. In the anticipatory tree topology, the context and action vectors are taken together. This concatenated vector identifies the node in the tree. It can be expanded following a top-down strategy: the initial tree contains a unique schema, with completely generalized context and action, and it is gradually specialized by differentiation, adding new relevant features to identify more precisely the category of equivalent situations, which entails the creation of new branches in the tree where the context and action vectors are each time more defined. In well-structured universes, the shorter way is starting with an empty vector and searching for the probably small set of features relevant to distinguish the important situations, rather than starting with a full vector and having to waste energy eliminating a lot of useless elements. Selecting the right set of relevant features to represent some given concept is a well-known problem in AI, and the solution is not easy, even using approximated approaches. To do this, CALM adopts a forward greedy selection (Blum & Langley 1997), using the data registered in the generalized episodic memory.

« 49 » The *expected result* vector can be seen as a label in each decider schema, anticipating how the world changes when the schema is activated. Initially, all different expectations are considered as different classes, and they are gradually generalized and integrated with others. The agent has two alternatives when the expectation fails. In a way that makes the knowledge compatible with the experience, the first alternative is to try to divide the scope of the schema, creating new schemas with more specialized contexts. Sometimes this is not possible and then the schema's expectation is reduced. In the expected result vector, "#" means that the element is not deterministically predictable. Another symbol can be used to represent some special situations, in order to reduce the number of schemas; this is the symbol "=", used to indicate that the value of the expected element will not be changed.

« 50 » Three basic methods compose the CALM learning function, namely: *differentiation*, *adjustment*, and *integration*. Differentiation is a necessary mechanism because a schema responsible for a too general context cannot often make precise anticipations. If a general schema does not work well, the mechanism divides it into new schemas, differentiating them by some element of the context or action vector. In fact, the differentiation method takes an unstable decider schema and changes it into a two level sub-tree. The parent schema in this sub-tree preserves the context of the original schema. The children, which are the new decider schemas, have context vectors that are a little more specialized than those of their parent. They attribute a value to some undefined element, dividing the scope of the original schema. Each one of these new deciders engages itself in a part of the domain. In this way, the previous correct knowledge remains preserved, distributed in the new schemas, and the discordant situation is isolated and treated only in its specific context. Differentiation is the method responsible for making the anticipatory tree expand. Each level of the tree represents the introduction of some new constraint. The algorithm needs to choose what will be the differentiator element, which could be from either the context vector or the action vec-

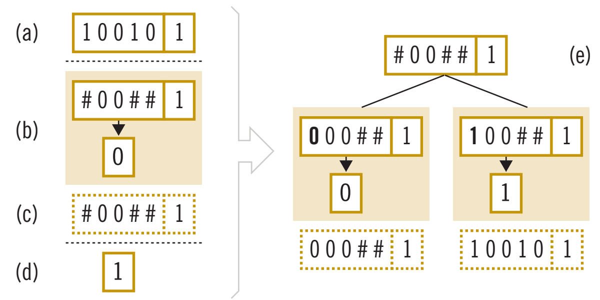

Figure 3: Differentiation method example: (a) a real experimented situation (with five variables) and executed action (one variable); (b) activated schema (with compatible context, action, and expectation); (c) associated episodic memory (representation of real situations where the scheme has been activated, in this case representing no interdependencies between variables); (d) real observed result, after the execution of the action; (e) sub-tree generated by differentiation in order to compensate the divergence observed between expectation and result.

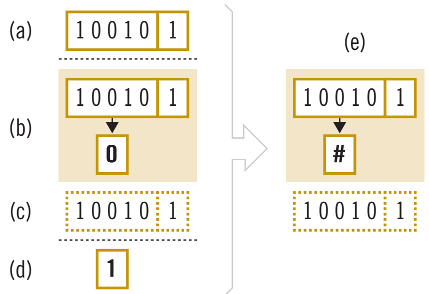

Figure 4: Adjustment method example: (a) a real experimented situation and action; (b) activated schema; (c) associated episodic memory; (d) real observed result; (e) schema expectation reduction after adjustment.

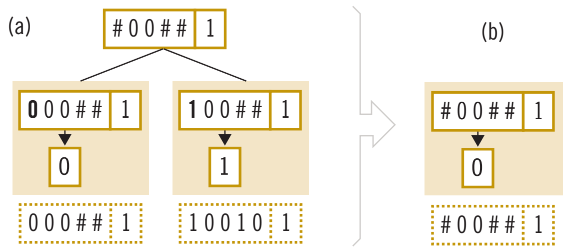

Figure 5: Integration method: (a) sub-tree after an adjustment; (b) an integrated schema substitutes the sub-tree.

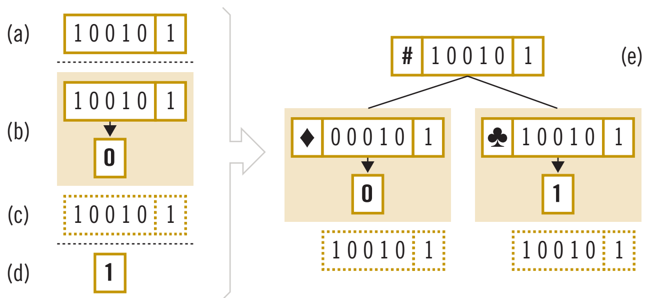

Figure 6: Synthetic element creation method: (e) incremented context and expectation vectors, and differentiation using a synthetic element.

tor. This differentiator needs to separate the situation responsible for the disequilibrium from the others, and the algorithm chooses it by calculating the information gain, and considering a limited (parametrized) range of interdependencies between variables. Figure 3 illustrates the differentiation process.

« 51 » When some schema fails and it is not possible to differentiate it in any way, then CALM executes the adjustment method. This method reduces the expectations of an unstable decider schema in order to make it reliable again. The algorithm simply compares the activated schema's expectation and the real result perceived by the agent after the application of the schema, setting the incompatible expectation elements to the undefined value ("#"). The adjustment method changes the schema's expectation (and consequently the anticipation predicted by the schema). Figure 4 illustrates this.

« 52 » In this way, the schema expectation can change (and consequently the class of the situation represented by the schema), and the tree maintenance mechanism needs to be able to reorganize the tree when this change occurs. Therefore, successive adjustments in the expectations of various schemas can reveal unnecessary differentiations. When CALM finds a group of schemas with similar expectations for approaching different contexts, the integration method comes into action, trying to join these schemas by searching for some unnecessary common differentiator element and eliminating it. The method operates as shown in Figure 5.

Dealing with the unobservable

« 53 » When CALM reduces the expectation of a given schema by adjustment, it assumes that there is no deterministic regularity following the represented situation in relation to these incoherent elements, and that the related transformation is unpredictable. However, sometimes a prediction error can be explained by considering the existence of some abstract or hidden property in the environment, which could be useful to differentiate an ambiguous situation but which is not directly perceived by the agent sensors. So, before adjusting, CALM assumes the existence of a non-sensory property in the environment, which will be represented as a synthetic element. When a new synthetic element is created, it is included as a new term in the context and expectation vectors of the schemas. The use of synthetic elements assumes the existence of something beyond the sensory perception, which can be useful to explain non-equilibrated situations. They have the function of amplifying the differentiation possibilities.

« 54 » In this way, when dealing with partially observable environments, CALM has two additional challenges: (a) inferring the existence of unobservable properties, which it will represent by synthetic elements, and (b) including these new elements into its predictive model. A good strategy for doing this is to look at the historical information.

« 55 » CALM introduces a method called *abstract differentiation*. When a schema fails in its prediction, and when it is not possible to differentiate it by the current set of considered properties, then a new Boolean synthetic element is created, enlarging the context and expectation vectors. Immediately, this element is used to differentiate the incoherent situation from the others. The method attributes arbitrary values to this element in each differentiated schema. These values represent the presence or absence of some non-observable condition, necessary to determine the correct prediction in the given situation. The method is illustrated in Figure 6, where the new elements are represented by card suits.

« 56 » Once a synthetic element is created, it can be used in subsequent differentiations. A new synthetic element will be created only if the existing ones are already saturated. To avoid the problem of creating infinite new synthetic elements, CALM can do this only up to a determined limit, after which it considers that the problematic anticipation is not deterministically predictable, undefining the expectation in the related schemas by adjustment. Figure 7 illustrates the idea behind synthetic element creation.

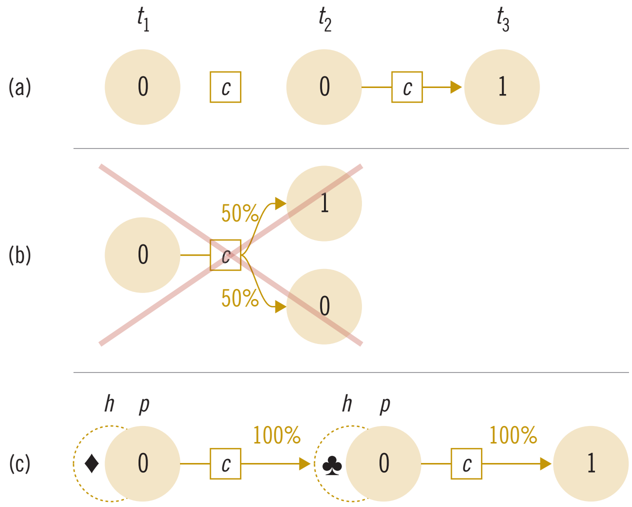

Figure 7: Discovering the existence of non-observable properties in: (a) a real experienced sequence; (b) what CALM does not do (the attribution of a probability); (c) the creation of a synthetic element in order to explain the observed difference

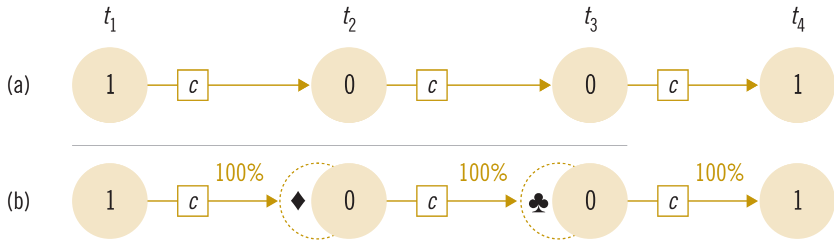

Figure 8: Predicting the dynamics of a non-observable property in: (a) a real experienced sequence; (b) the use of a synthetic element to explain the logic behind the observed transformations.

« 57 » The synthetic element is not associated to any sensory perception. Consequently, its value cannot be observed. This fact can place the agent in ambiguous situations, where it does not know whether some relevant but non-observable condition (represented by this element) is present or absent. Initially, the value of a synthetic element is verified *a posteriori* (i.e., after the execution of the action in an ambiguous situation). Once the action is executed and the following result is verified, then the agent can rewind and deduce the situation really faced in the past instant (disambiguated). Discovering the value of a synthetic element after the circumstance where this information was needed can seem useless, but in fact this delayed deduction gives information to another method called *abstract anticipation*. If the non-observable property represented by this synthetic element has a regular dynamics, then the mechanism can propagate the deduced value back to the schema activated in the immediately previous instant. The deduced synthetic element value will be included as a new anticipation in the previous activated schema. Figure 8 shows how this new element can be included in the predictive model.

« 58 » For example (complementing Figure 8), in time t_1 CALM activates a schema $\Xi_1 = \{\#1 \wedge c \rightarrow \#0\}$, where the context and expectation are composed of two elements (the first one synthetic and the second one perceptive) and one action. Suppose that the schema succeeds and, as predicted, the next observation is "0". The problem is that the next situation "#0" is ambiguous because it excites both the schemas, $\Xi_2 = \{\blacklozenge 0 \wedge c \rightarrow \#0\}$ and $\Xi_3 = \{\clubsuit 0 \wedge c \rightarrow \#1\}$. At this time, the mechanism cannot know the value of the synthetic element, crucial to determining the real situation. Suppose that, anyway, the

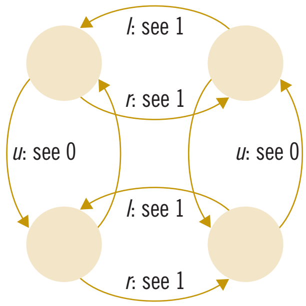

Figure 9: The hyper-flip problem.

mechanism decides to execute the action "*c*" in time t_2, and this is followed by the sensory perception "0" in t_3. Only now, in t_3, the agent can deduce that the situation really dealt with in t_2 was "♦0", and it can include this information in the schema activated in t_0, in the form $\Xi_1 = \{\#1 \wedge c \rightarrow \blacklozenge 0\}$.

Experimental results

« 59 » To exemplify the functioning of the proposed method, we will use the hyper-flip problem, and extension of the problem used by Satinder Singh et al. (2003) and Michael Holmes & Charles Isbell (2006). It consists of an agent who lives in a two-state universe. It has 3 actions (*l*, *r*, *u*) and 2 perceptions (0, 1). The agent does not have any direct perception of the underlying current state. It sees "1" when the state changes horizontally, and "0" otherwise. Action "u" changes the state vertically, action "l" causes the deterministic transition to the left state, and action "r" causes the deterministic transition to the right state. The flip problem is showed as a state machine in Figure 9.

« 60 » CALM is able to solve this problem. First, the mechanism tries to predict the next observation in function of its action and current observation. However, it quickly discovers that the perceptive observation is not useful to the model, and that there is insufficient information to make correct anticipations. So, it creates a new synthetic element that will be able to represent the underlying left (♣) and right (♦) states. Figure 10 shows the final solution. It is interesting to note that the constructed world model (with its 3 variables) is not a copy of the ontological structure of the problem (a machine with 4 states).

« 61 » In order to test the robustness of the mechanism, a hundred new observable variables have been inserted in the hyper-flip problem for a second scenario. These new variables present random transformation functions and do not influence the evolution of the original observation. The result is that the mechanism is not affected in its capacity to solve the problem (it finds the same solution as that previously indicated). The time of learning increases in a linear order with the addition of irrelevant variables.

Related work

« 62 » CALM is an original mechanism that enables an agent to create incrementally a model of an experience during the course of its interaction with the universe. The pioneer work on constructivist AI was presented by Drescher (1991). He proposed the first constructivist agent architecture, which learns a world model by an exhaustive statistical analysis of the correlation between all the context elements observed before each action, combined with all resulting transformations. Drescher has also suggested the need to discover hidden properties by creating "synthetic items."

« 63 » The schema mechanism represents a strongly coherent model. However, there are no theoretical guarantees of convergence. Another restriction is the computational cost of the kind of operations used in the algorithm. The need for space and time resources increases exponentially with the problem size. Nevertheless, some other researchers have presented alternative models inspired by Drescher, such as Yavuz & Davenport (1997), Morrison, Oates & King (2001), Chaput (2004), and Holmes & Isbell (2005), always based on the search for statistically observed regularities.

« 64 » CALM differs from these previous works because we limit the problem to the discovery of deterministic regularities (even in partially deterministic environments). In this way, we can implement direct induction methods in the agent learning mechanism. This approach presents a low computational cost, and it allows the agent to learn incrementally and find high-level regularities. For that, we have been inspired by Holmes & Isbell (2006), who used the notion of the state signature as a historical identifier of the states to develop the idea of learning anticipations through the analysis of relevant pieces of history.

« 65 » With the emergence of the factored MDP model, some important works have been realized to create algorithms designed to discover the structure of the system (Degris, Sigaud & Wuillemin 2006; Degris & Sigaud 2010; Strehl, Diuk & Littman 2007; Jonsson & Barto 2005). However CALM, as far as we know, is the only one to merge the induction of synthetic elements to represent the non-observable variables in an FPOMDP.

« 66 » Another originality of CALM is the use, in such learning problems, of a generalized episodic memory associated to the search for important variables (related to affective values or relevant to anticipate the evolution of other important variables).

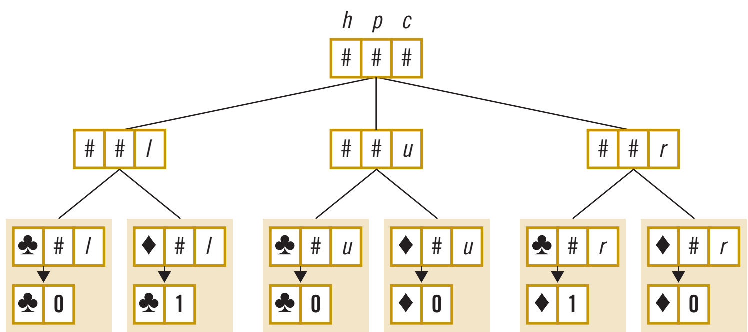

Figure 10: Final schematic tree for solving the flip problem. The vector represents synthetic elements (h), perceptible elements (p) and actions (c). The decider schemas show the expectations.

Conclusion

« 67 » The CALM mechanism can provide autonomous adaptive capabilities to an agent because it is able to construct knowledge incrementally to represent the deterministic regularities observed during its interaction with the environment, even in partially deterministic universes.

« 68 » CALM is able to deal with partially observable environments, detecting high-level regularities. The strategy is the induction and prediction of unobservable properties, represented by synthetic elements.

« 69 » Synthetic elements enable the agent to step beyond the limit of instantaneous and sensorimotor regularities. In the agent's mind, synthetic elements can represent three kinds of "unobservable things": (a) hidden properties in partially observed worlds, or sub-environment identifiers in discrete non-stationary worlds; (b) markers to necessary steps in a sequence of actions, or to different possible agent points of view; and (c), abstract properties, which do not exist properly, but which are powerful and useful tools for the agent, enabling it to organize the universe into higher levels.

« 70 » With these capabilities, CALM is able to step beyond sensorial perception, constructing more abstract terms to represent the universe and to "understand" its own reality in more complex levels. CALM can be very effective for constructing models in partially but highly deterministic ($1 > \partial \gg 0$) and partially but highly observable ($1 > \omega \gg 0$) environments, and when the transformation functions have well-structured causal dependencies ($0 < \varphi \ll n$).

« 71 » Currently, we are improving CALM to enable it to form action sequences by chaining schemas. It will allow the creation of composed actions and plans. The next research steps include: formally demonstrating the mechanism's robustness and correctness; making comparisons between CALM and related solutions proposed by other researchers; and analyzing the mechanism's performance when facing more complex problems. Future works could include the extension of CALM to deal with non-deterministic regularities, noisy environments and continuous domains.

Received: 12 July 2013
Accepted: 19 September 2013

Open Peer Commentaries

on Filipo Studzinski Perotto's "Computational Constructivist Model"

To Bridge the Gap between Sensorimotor and Higher Levels, AI Will Need Help from Psychology

Frank Guerin
University of Aberdeen, UK
f.guerin/at/abdn.ac.uk

> Upshot • Constructivist theory gives a nice high-level account of how knowledge can be autonomously developed by an agent interacting with an environment, but it fails to detail the mechanisms needed to bridge the gap between low levels of sensorimotor data and higher levels of cognition. AI workers are trying to bridge this gap, using task-specific engineering approaches, without any principled theory to guide them; they could use help from psychologists.

« 1 » The formulation of the problem as it appears in the abstract of Filipo Perotto's article packs in a lot of information that merits discussion:

> "The constructivist conception of intelligence is very powerful for explaining how cognitive development takes place. However, until now, no computational model has successfully demonstrated the underlying mechanisms necessary to realize it. In other words, the artificial intelligence (AI) community has not been able to give rise to a system that convincingly implements the principles of intelligence as postulated by constructivism, and that is also capable of dealing with complex environments."

« 2 » This suggests that the psychologists have succeeded in explaining how cognitive development takes place and that the AI community has failed in its job to implement these "principles of intelligence." However, I would throw the problem back at the psychologists. I think that significant work is still needed at the level of theoretical psychology before we have something close to a proper explanation of how cognitive development takes place. Psychological explanations are for the most part vague and woolly; they do not elucidate the mechanisms underlying development (Jean Piaget's theory being a good example). Furthermore, Piaget's theory is at times even at odds with experimental psychology. It may be many, many years before we have a suitably detailed theory from the psychologists that is consistent with the evidence from experiments. Until that time, one could argue that the "principles of intelligence as postulated by constructivism" are implemented very well by existing AI systems. Gary Drescher, for example, did implement the basic principles of constructivism, but "it could never be used to solve significant applied problems," because the techniques do not scale up to systems with large numbers of inputs and degrees of freedom. However, Piaget did not give us any idea of how to deal with these issues, so one could lay the blame on him.

« 3 » To quote from the abstract again, "there is a large distance between the descriptions of the intelligence made by constructivist theories and the mechanisms that currently exist." If we consider Piaget's theory, and Drescher's system or the CALM system, I am not sure that there is such a large distance. Piaget's descriptions of assimilation and accommodation are so all encompassing and so lacking in detail that it seems to me that Drescher's system or the CALM system constitute perfectly good implementations. Psychology tends to leave mechanisms very underspecified.

« 4 » To quote again from the article's abstract: "…and that is also capable of dealing with complex environments." Here is perhaps the essence of the problem. When you start building an actual AI system that has to interact with the world, you face a daunting task of dealing with a complex environment. It seems that AI is being saddled with the burden of not only implementing the high-level theory, but also making sure it can deal with complex environments. The "complex environments" problem needs to be thrown back at the psychologists. The history of AI has shown that a theory of cognition that works at a high abstract level but cannot account for the interface to the sensorimotor level is not much of a theory of cognition at all. The devil is in the detail. There are many writers who convincingly show how high-level cognition is very much grounded in our sensorimotor intelligence (e.g., Barsalou 2008; Byrne 2005; Bril, Roux & Dietrich 2005). Psychological theories tend to overlook the need for complex mechanisms to bridge the gap between the sensorimotor level and high-level cognition. Psychologists may need to become computer scientists to some extent, so that they have an appreciation of the computational problems involved and the need for them to describe mechanisms to account for how humans successfully solve these.

« 5 » On the positive side, there are some works in cognitive science that are beginning to attempt to address the issue of providing some theoretical framework to account for how a sensorimotor level can connect with higher levels of cognition: for example, the multi-layered cognitive system of Bipin Indurkhya (1992, Chapter 5).

« 6 » For the CALM system itself, I feel the article has all the correct ideas from a philosophical and psychological point of view, e.g., about the agent constructing its own symbolic structures and not having access to the "ontological reality." However, if we are to evaluate it as a candidate for a "general artificial intelligence mechanism that learns like humans do" (first sentence of abstract), then it might suffer the same shortcomings as Drescher's work, i.e., "it could never be used to solve significant applied problems." For example, if the context were to be the visual input from two stereo cameras delivering a few million pixels in 24 bit colour at thirty frames per second and the system is trying to predict the consequences of actions, in the complexity of an everyday setting, in this visual stream, it might not be feasible to use each bit of input as a CALM variable. One could, of course, propose to hook the CALM system up to a higher-level abstracted version of the visual input, but then one runs into the issues of where to make the cut-off between what the core CALM system sees and what is the responsibility of other abstraction mechanisms. If the cut-off is at the wrong place, then one runs into classical AI problems of (a) having a core cognition that makes unreasonable assumptions about how accurately it can interface with the world or (b) having a prespecified worldview imposed by the provided abstractions (see Brooks 1991 or Stoytchev 2009 for problems with this). There does not seem to be any clear theory from psychology to guide us on how to connect the sensorimotor level with some higher levels. AI does have various different applied systems that successfully make a connection from high-level symbols to perception and action in complex settings: for example, robots that perform everyday tasks (Beetz et al. 2010). However, each applied AI system tends to be specialised and optimised for one particular task. None could claim to be a reasonable model of general human cognition, nor do they attempt to be. This is really a job for the psychologists.

Frank Guerin obtained his Ph.D. degree from Imperial College, London, in 2002. Since August 2003, he has been a Lecturer in Computing Science at the University of Aberdeen. He is interested in understanding the core of cognition in computational terms. He has focused on understanding infant cognitive development, as a first step to understanding later cognition.

Received: 11 October 2013
Accepted: 18 October 2013

Environments Are Typically Continuous and Noisy

Martin V. Butz
University of Tübingen, Germany
martin.butz/at/uni-tuebingen.de

> **Upshot** • The schema system presented in the target article suffers from problems that had been acknowledged more than ten years ago. The main point is that our world is neither deterministic nor symbolic. Sensory as well as motor noise is ubiquitous in our environment. Symbols do not exist a priori but need to be grounded within our continuous world. In conclusion, I suggest that research on schema-learning systems should tackle small but real-world, continuous, and noisy problem domains.

Heuristic learning principles are not enough

« 1 » About 15 years ago, I began working together with Wolfgang Stolzmann and Joachim Hoffmann on the development of anticipatory classifier systems (Stolzmann 2000). We attempted to tackle the fundamental problems of learning a cognitive model in well-structured environments, implementing contextual rule differentiation, rule adjustment, and rule integration mechanisms. With iterative improvements and additions, the ACS2 system was developed. ACS2 combines a heuristic rule differentiation and specialization mechanism that is based on Hoffmann's cognitive learning principle, termed "anticipatory behavioral control" (Hoffmann 2003), with a generalization mechanism that is implemented by a steady-state evolutionary algorithm in ACS2. In my book on *Anticipatory Learning Classifier Systems* (Butz 2002), I summarized the capabilities of the developed system as well as the fundamental challenges.

« 2 » While the fundamental challenges included the problem of *partially observable Markov decision processes* (POMDPs), I had also acknowledged that "essentially any characteristic in an environment that causes the deterministic perceptual causality to become probabilistic or noisy causes difficulties" (Butz 2002: 127). I fear that the algorithm presented in the target article suffers similar difficulties. That is, while it may be able to solve the tackled, small POMDP problem, it is very doubtful that the heuristic learning mechanism put forward is able to produce similarly good solutions in noisy, continuous environments.

« 3 » Is this a concern for the constructivist community? In the following I will argue that it is indeed a severe concern and I propose that the community should focus their research efforts on working with systems that experience noisy, continuous environments rather than symbolic ones.

Noisy experiences

« 4 » Our Western world seems dominated by symbolic knowledge, and so we tend to forget our actual, natural environment. In this environment, our perceptions are typically continuous and noisy, and manipulations of and interactions with the environment sometimes fail by nearly pure chance. How can we live in this messy, nondeterministic environment with all its caveats? How can we learn a useful world model with which we can manipulate and interact with the environment purposefully?

« 5 » Various evidences suggest that our mind constructs predictive models about the consequences of body-environment interactions (Butz 2008). Even in the simplest cases, a certain form of causality is present during such interactions. Thus, learning about condition-action-effect contingencies is possible and such knowledge is useful when striving for a particular effect. However, a full-blown model of the environment with all its inherent contingencies is too large to grasp. Thus, as the author also suggests in §42, most likely the models we learn need to focus on those aspects that are relevant for us, that is, those that are associated with positive or negative affective values.

« 6 » However, these models need to be functional in noisy, continuous environments. Thus, conditions, actions, and effects are initially not symbolic but consist of contextual subspaces, motor primitives, and local perceptual changes. According to Lawrence Barsalou (1999) and others, we learn our symbol processing capabilities during our lives, grounding these capabilities in our perceptual, noisy, and continuous experiences.

« 7 » Most schema-oriented learning systems, such as the one proposed in the target article, have not managed to develop symbol systems in a noisy, continuous realm. Schema learning systems up until now have stuck to symbol manipulation problems, such as the admittedly tricky hyper-flip problem. But are these problems constructive? Can they lead to a system that may convincingly develop a constructivist system that becomes *cognitive*? I doubt it.

Natural environments

« 8 » What can be done about it? I believe that the constructivist community should focus on the question of how symbol processing capabilities can develop in noisy, continuous environments – where experiences are grounded and embodied in an actual bodily perception-action system. Evidence has been accumulating over recent years that this is not an insurmountable endeavor. The *theory of event coding* (Hommel et al. 2001) postulates that events may be a highly important cognitive concept for structuring experiences and thus for perceiving the environment in chunks that may be symbolizable. Also, in the cognitive robotics literature, the registration of events – such as when touching an object – has been acknowledged as one key mechanism for segmenting the environment into meaningful interaction components (Wörgötter et al. 2013). Bodily interactions with the environment were structured into a natural action grammar with properties that are strongly related to Noam Chomsky's universal grammar (Pastra & Aloimonos 2012). Research from my own group suggests that goal-oriented representations should be separated from representations of spatial interaction for setting the stage to develop compositional concept structures, which are necessary for language development (Butz 2013).

« 9 » In conclusion, I agree with the authors that schema learning approaches should be re-considered and revived. Starting with a symbolic world and facing one particular, partially-observable toy problem, however, will not advance schema learning mechanisms. Rather, these mechanisms need to be implemented in environments within which interactions are continuous, state transitions are stochastic to a certain degree, and perceptions are noisy. Tools and mechanisms are currently being developed that can segregate these continuous realms into meaningful and purposeful symbol systems. Key components of such mechanisms are anticipations, modularizations, and event-based separations. Measures of valence and resulting purposeful, goal-oriented interactions are most likely additional key concepts. A learning system that builds schemas based on these principles may indeed be the way forward towards scalable cognitive systems that develop in complex environments, effectively implementing constructivist theories of cognition.

Martin V. Butz is Full Professor at the Eberhard Karls University of Tübingen, in the Faculty of Science, Department of Computer Science, and Department of Psychology. His research group works on cognitive modeling, focusing on how the brain develops representations of the body and the surrounding space and how these representations are used to manipulate the environment goal-directedly.

Received: 16 October 2013
Accepted: 21 October 2013

The Power of Constructivist Ideas in Artificial Intelligence

Kristinn R. Thórisson
Reykjavik University, Iceland
thorisson/at/gmail.com

> Upshot • Mainstream AI research largely addresses cognitive features as separate and unconnected. Instead of addressing cognitive growth in this same way – modeling it simply as one more such isolated feature and continuing to uphold a wrong-headed divide-and-conquer tradition – a constructivist approach should help unify many key phenomena such as anticipation, self-modeling, life-long learning, and recursive self-improvement. Since this is likely to result in complex systems with unanticipated properties, all cognitive architecture researchers should aim to implement their ideas in full as running systems to be verified by experiment. Perotto's paper falls short on both these points.

«1» Cognitive growth, self-inspection, anticipation (prediction based on partial observation), self-organization – what do these have in common? They are all part of a growing set of concepts from biology, cognitive science, artificial intelligence, and psychology that must be related to one another if we are ever to produce a coherent theory of intelligence, whether in machines, animals, or humans. And if our aim is to build working systems – if our stance is a software engineering one with an end-goal of building deployable systems that can operate in real-world environments, whether it be space probes, housecleaning robots, deep-sea explorers, or stock-market investment programs – then our methodological approach must embody principles that are useful for steering our efforts when designing, architecting, implementing, and testing our systems.

«2» Filipo Perotto presents in his paper a model of an anticipatory learning mechanism, CALM, which is based on constructivist principles. His high-level model of agent-environment coupling, CAES, seems a reasonable one. Both models are based on the fundamental assumptions, which I agree with, that: (a) to understand intelligent behavior we must include in our analysis the context in which it operates; and (b) most environments of any interest to intelligent beings contain a mixture of deterministic and non-deterministic causal connections, with many of the former remaining invisible. In my view, and it would seem Perotto's as well, an environment with complex causal relationships (e.g., our everyday world) gives rise to a vast number of potentially observable phenomena, many of which do not clearly or readily convey their underlying causes; this set of potential observable and inspectable phenomena is nevertheless the only information that an intelligent system has access to, via their sensory apparatuses, for anticipating how their external environment behaves so as to efficiently and effectively achieve its goals within it.

«3» Before continuing with direct commentary, some points are in order so as to elucidate the context in which I look at systems engineering, architecture, and constructivism. Due to the high number of combinatorics that a complex environment will produce, through countless interactions between its numerous elements, an agent must create *models* that isolate and capture some essence of underlying causes (invariants or partial invariants) in this environment (Conant & Ashby 1970). Such models must be capable of capturing abstract levels of detail that can be used to steer the operations of a system towards efficient expenditure of computational resources –any thought spent on details completely unrelated to goals (future and present) would be a waste of the agent's time. Thus, the partial models of the environment that an intelligent agent creates will likely form some sort of a cognitive "random-access" abstraction hierarchy. Depending on the type of current goal and situation, the agent can then choose models at a particular level of abstraction at any time to help it exclude irrelevant issues from consideration when decisions are being made about how to achieve the goal in that situation. A coherent, unifying model of cognition following constructivist principles must explain how this works, in particular how goals, models, experiences, and iterative knowledge acquisition and improvement operate in concert to achieve cognitive growth in an agent. An engineering methodology for how to build artificial systems implementing such functions must go further, by helping with defining specifications for an implementable architecture, and providing guidelines on how to implement them in a way that allows experimental evaluation.

«4» An artificial system built to achieve general intelligence must be able to deal with novel situations – situations not foreseen by its programmers. Instead of being given pre-programmed algorithms by its designers, known to be applicable to particular and specific problems, tasks, situations, or environments, the AI itself must be imbued with the ability to generate algorithms (or, compute a control function – I do not distinguish between the two here). For this to be possible, the system must furthermore be equipped with the ability to (re-) program itself, otherwise it cannot sensibly change its own operation in any meaningful way based on acquired experience. And to be able to do so, the system must be reflective – that is, the system's architecture and operational semantics must be captured in a way that enables it to read and interpret its own structure and operation. This is what I consider the essence of a constructivist AI methodology: specifications for how to imbue machines with the capability to make informed changes (whether slowly or quickly) to their own operation, via the runtime principles embodied in their architecture. I do not believe constructivist AI can be done without some form of self-programming on the part of the machine, which in turn cannot be achieved without transparency of its operational semantics. In fact, even more radically, I suspect artificial general intelligence cannot be achieved *at all* without such capabilities; higher levels of cognitive operation in the context of novel or unanticipated tasks, situations, and environments must require some sort of cognitive growth – namely, some form of re-programming of the cognitive system's operation. Conversely, constructivist views on cognition are so different and incompatible with standard software engineering methodologies, especially with its tradition of manual software creation, that they cannot be used at all for engineering such systems. To address constructivist principles head on in a computational framework will require a new *constructivist AI methodology* (CAIM; Thórisson 2012).

« 5 » Whether or not Perotto agrees with my views on the nature and need for constructivist development principles thus outlined, he does make some claims to taking steps toward computational implementations of constructivist principles. In this context, many important questions come to mind – chief among them being how effective the ideas are for explaining cognitive growth in nature, and how useful might they be for helping implement artificial general intelligence. As Perotto's paper seems to be aimed more at the second topic, we can ask, firstly, do the ideas presented in his paper help with – or are they likely to lead us to – better software engineering methods for implementing constructivist learning in deployed systems? Secondly, we can ask, if they do in fact offer some new insights to this end, how much still remains to be explained for such systems to spring forth as a result – or conversely, how big a part of the constructivist puzzle does the work attempt to address? Let us look at these in order.

« 6 » The aim of AI is not just to speculate but to build working, implemented systems. In AI, any theoretical construct aimed at advancing our understanding of how to implement cognitive functions should ultimately be judged on whether actual implementation can conclusively, or partially, allow us to conclude through reliable means (i.e., scientific experimentation), that the ideas, when operating in a relatively complete AI architecture situated in a complex world (Perotto's target environments), are capable of *scaling up*. By "scaling up" I mean the ability of a system to grow in a way that supports recursive self-improvement in complex environments (e.g., the physical world), with respect to its top-level goals. This question is of course difficult to answer, whether experimentally or analytically. A quick walk down memory lane reminds us, however, that the history of AI is replete with examples of proposals that looked great on paper but completely failed such scaling up when implemented in a running system, or when attempts were made to expand the models the ideas embodied to include more of the many functional characteristics that they originally left untouched. Unfortunately, experimental evaluation of Perotto's proposed ideas is touched on only briefly in the paper, and the support provided to answer this question is inconclusive at best. On this count, therefore, not much can be said about the scalability of Perotto's ideas. This is disappointing because a fundamental feature of known constructivist systems in nature is their capability to grow cognitively with experience – itself a form of scaling-up. Other phenomena, such as the power of the CALM schema formalism to produce new knowledge of complex environments, to support models of self (required for any system capable of self-directed cognitive growth), and their ability to support self-inspection, are also not addressed to any sufficient extent in the work. Since these issues are briefly touched on or left unmentioned, we can only assume that they remain unaccounted for by the present work.

« 7 » My second question regards the "size of the intelligence puzzle" addressed. An artificial cognitive system must, to have a chance at becoming a comprehensive theory of the major facets of intelligence, include a large number of functions that allow the system to operate relatively autonomously in complex environments. This *theoretical scalability* of an isolated mechanism is its perseverance and robustness when included in a better (larger, more comprehensive) model/theory, which can in turn serve as the foundation for building systems with increased operating power, including an increased capacity for cognitive growth and architectural complexity. If Perotto's work turns out to be correct, if it indeed offers, as Perotto claims in the abstract, "a step towards computational implementation of constructivist principles," how much of the phenomenon in question – cognitive growth – remains to be explained? The lack of a clear connection between his CALM and CAES models is already a sign that some amount of work remains to be done in this direction. My own list of candidate principles and features (cf. some already mentioned above) that should be accounted for in any reasonable theory of cognitive growth is, unfortunately, quite a bit longer than that addressed in Perotto's paper. Firstly, as described above, cognitive growth requires some kind of autonomic, recursive self-improvement. Although my team has made some progress on this front recently (Nivel & Thórisson 2013, Nivel et al. 2013), research on the topic is still in its infancy, with a host of unanswered practical and theoretical questions. Such questions include: What kind of representations[1] are amenable to automatic self-programming for cognitive growth (existing programming languages and paradigms created for humans require human-level intelligence to be used – which calls for the very phenomenon we are striving to understand how to implement); how can the transparent operational semantics needed for automatic programming be achieved? Related to that are the questions: How can a system's operational semantics be measured; what kind of meta-level control structures can be used to steer cognitive growth; what kinds of control architectures can serve as host architectures for the proposed (or any other) constructivist principles? Questions regarding theoretical scalability issues loom large.

« 8 » These are, of course, not simple topics. Quite the contrary, they are deep and challenging. But they are central to constructivist approaches, developmental robotics, and principles of cognitive growth, and it is precisely for that reason that they must not be left unaddressed, lest our efforts become victims to the same oversimplification and incorrect application of divide-and-conquer methodology that has plagued much of AI research in the past half century (cf. Thórisson 2013). Unlike so many other phenomena in AI, e.g., planning, vision, reasoning, and learning, that have been largely addressed by calling them "computational" and studying them in isolation through the same strictly allonomic methodologies as used for banking systems, word processors, and Web page construction, a constructivist methodology holds a promise – a potential – to unify a host of complex cognitive mechanisms, most of which have eluded scientific explanation so far. A holistic stance is by far the most likely to lead to an understanding of the phenomenon of intelligence, and anyone with a constructivist mindset has already taken an important step in that direction. But for this to pave the way towards a bet-

1 | My use of "representations" implies a larger scope than models, capturing virtually anything that might be needed to be encoded in a particular runtime medium for a running ("live") intelligent system.

ter theory, a genuine attempt must be made to weave as many key cognitive phenomena into the account as possible, to attempt to provide a unifying account. And for any engineering effort to be taken seriously, the requirement for experimental evaluations of (physical and/or virtual) running software systems cannot go ignored. Perrotto's stance on these pressing issues remains for the time being largely unknown; we can only hope that he addresses them in the future.

Kristinn R. Thórisson has been doing research in artificial general intelligence and real-time interaction for over two decades in academia and industry. His AERA constructivist cognitive architecture is the world's first system that can learn complex skills by observation in largely underspecified circumstances. He is a two-time recipient of the Kurzweil Award and has a Ph.D. from Massachusetts Institute of Technology.

Received: 16 October 2013
Accepted: 24 October 2013

Anticipatory? Yes. Constructivist? Maybe

Georgi Stojanov
The American University of Paris, France
gstojanov/at/aup.edu

Upshot: The CALM cognitive agent with its learning mechanism, as presented by the author, can be described as "trivially constructivist." Probably, at best, it can be seen as a model of the empirical abstraction but not of the reflective abstraction. The "intrinsic motivations" in the simulated agent presented as "evaluative signals" sent from the agent's "body" to its "mind" can be seen as low-level physiological drives. They cannot account for far more sophisticated intrinsic motivations such as curiosity.

« 1 » In the opening sections and in §1, Filipo Perotto sets up a formidable challenge for himself by promising a step toward an artificial general intelligence (AGI) that follows the constructivist approach of Piagetian flavor. There is also an explicit critique of this constructivist AI, in which despite all the promises made, there has been a "lack of concrete results." The critique is justified if the expected concrete result was to build an artifact that would exhibit the behavior of a three-year-old infant. We are certainly not there yet. On the other hand, the constructivist AI approach certainly made huge theoretical advances by demonstrating the inappropriateness of the traditional software methodology to deal with the design of self-constructive autonomous intelligent agents (e.g., Thórisson 2012), or shifting the research focus to issues neglected in traditional AI: sensorimotor interaction, intrinsic motivation, complete cognitive architectures (e.g., Stojanov, Kulakov & Clauzel 2006; Stojanov & Kulakov 2011).

« 2 » In §2 and §3, Perotto introduces the conceptual structure of a schema: context ∧ action → expectation, which he also calls an "elementary piece of knowledge." The "context" vector represents the readings of all external and internal sensors, and when some "action" is executed, the agent anticipates the outcome in terms of the "expectation" vector. Thus, throughout its lifetime, the agent put in particular environment should learn to predict the outcomes of its actions ("to adapt itself"), even if the environment is partially observable. Many researchers have used this "context ∧ action → expectation" construct (Drescher 1991; Schachner 1996; Schachner, Real del Sarte & León 1999; Tani 1996; Stojanov, Bozinovski & Trajkovski 1997; Chaput 2004; see Stojanov 2009 for an overview of computational models of Piagetian schemas) in the task of learning forward-models (or anticipative models) of the environment. The simulated environment is represented via a FPOMDP (§28). The states of the environment are represented with a set of properties X, and among those properties there are some that cannot be perceived by the agent's perceptual apparatus. This leads to perceptual aliasing and makes the problem of learning effects of actions in given contexts much more difficult. CALM (§38) is the learning mechanism designed to learn the dynamics of the underlying FPOMDP through execution of agent's actions (which, from the point of view of the FPOMDP are controllable variables) and construction of reliable predictive schemas, described above. §50, §51 and §52 describe the three basic methods for schema construction in CALM: differentiation, adjustment, and integration. As there are unobservable properties of the environment, sometimes CALM will fail to predict accurately the effect of some action, and in some cases, the situation can be remedied by abstract differentiation (§55). Essentially, this means that the context and expectation vectors are arbitrary values that are enlarged and attributed in a way to make them distinct from existing schemas. The new schemas are called synthetic elements as they cannot be directly perceived. The method of propagation of the value of the synthetic elements is called abstract anticipation (§57). Once (if the complexity/observability ratio allows) the agent using CALM learns the environment model perfectly, it can always predict the effect of its action in a given context.

« 3 » My condensed (and, I hope, not too simplistic) description of CALM in the previous paragraph is to show that (although an original and efficient solution) it is *constructivist* only in a trivial way: it learns a model of its environment incrementally. Perotto appears to be like many developmental psychologists in the 1970s:

> "What they [developmental psychologists] called construction seemed to refer to the fact that children acquire adult knowledge not all at once, but in small pieces. I did not think that this was a revelation and therefore called their approach 'trivial constructivism'." (Glasersfeld 2005: 10).

The monolithic single-thread algorithm is completely deterministic and will (eventually) come up with the same result, given the same learnable environment. There is no learning-to-learn (i.e., change of the learning trajectory) nor ability for reconceptualization of a given situation, or evolution of more sophisticated intrinsic motivations (more about motivations below). Moreover, as Perotto notes in §9, "The agent needs to be able to detect high-level regularities in the dynamics of the environment, but this is not possible if the agent is stuck in a rigid *representational vocabulary*." The representational vocabulary of a CALM-driven agent is rigid: all of the possible different schemas. Synthetic items definitely enlarge it, but only up to a certain predefined limit. There is no way in which (in a genuinely constructivist spirit) the agent can impose some organization on the sensed environment. In continuation of §9 we can read

> “In a constructivist approach, cognitive development must be a process of gradual complexification of the intelligence, where more abstract structures (symbolic) are built from simpler sensorimotor interactions, in a way that harmonizes the lived experiences with the constructed model.”

CALM does not provide a way to build “more abstract structures… from simpler sensorimotor interactions.” At best, there are the synthetic elements that contain abstract properties in the sense that they do not correspond to any sensory inputs. Given that those abstract properties are added to schemas whose context and action vectors are equal, it is impossible to understand them as abstract/symbolic structures in the sense given in §9. In Piagetian parlance, the learning exhibited by CALM could be seen as model of the *empirical abstraction* but not of the *reflective abstraction* that is crucial for cognitive development and creative behavior. Briefly, via *empirical abstraction,* some quality (e.g., weight or color or contingency among actions and qualities) is abstracted from an object/situation. On the other hand, reflective abstraction is about reorganization of existing schemas and their projection onto a higher plane. (See Kitchener 1986: 61–65 for an informative discussion of empirical and reflective abstraction, as well as the discussion in Campbell & Bickhard 1993 on *the knowing levels*).

« 4 » In §13, one can read that the agent’s body with its “internal states and metabolisms, elements that belong neither to the mind nor the environment… allow the agent to have intrinsic motivations…” I believe that the decision to introduce the two entities (“body” and “mind”) is somewhat arbitrary, given that it is barely mentioned in the rest of the paper. It appears that the body is introduced only to have the above-mentioned possibility to have “intrinsic motivations.” If this is the case, then the intrinsic motivations can be related to low-level physiological drives (hunger, pain-avoidance) with no possibility for development of more sophisticated forms of motivations such as curiosity. If, on the other hand, the intrinsic motivations can be placed in the “mind” of the agent, I see no reason to draw the arbitrary body-mind distinction.

Georgi Stojanov obtained his PhD degree in computer science and AI from UKIM University in Skopje, Macedonia in 1997 on “Anticipation theory and electro-expectograms in the context of biological and machine intelligence.” His research interests include: interactivism and constructivism; cognitive development and creativity; analogy and metaphor. He was a visiting scholar at the University of Trieste, Italy, Les Archives Jean Piaget in Geneva, Université de Versailles-Saint-Quentin-en-Yvelines in Paris, and at the Institute for Non-linear Science, UCSD. As of 2005, he is an associate professor at the American University of Paris.

Received: 13 October 2013
Accepted: 18 October 2013

Action, Anticipation, and Construction: The Cognitive Core

Mark H. Bickhard
Lehigh University, USA
mhbo/at/lehigh.edu

> Upshot • Interaction-based models of cognition force anticipatory and constructivist models. The CALM model offers significant development of such models within a machine learning framework. It is suggested that moving to an entirely interactive-based model offers still further advantages.

« 1 » Charles Sanders Peirce introduced action and interaction as the proper loci for understanding the mind well over a century ago (Joas 1993). An interaction-based model of cognition, in turn, is intrinsically anticipatory – i.e., anticipations of potential actions and interactions (Bickhard 2009b; Buisson 2004; Piaget 1954). And an action and interaction-based model of cognition forces a constructivism: it is not feasible for the world to impress competent interactive system organization into a passive mind; it must be constructed. For yet another step, given that prescience does not exist, such a constructivism must be a variation and selection constructivism, an evolutionary epistemology (Campbell 1974). These characteristics, thus, form a coherent framework for understanding cognition, and, more generally, mind (Bickhard 2009b).

« 2 » Classic passive mind models, however, descending from the ancient Greeks, still dominate the scene, currently in their “recent” incarnations of symbolic computationalism and connectionism. Machine learning is an interesting combination of perspectives: learning about the environment requires checking what is tentatively learned against that environment, which requires action and anticipation and construction of what is checked. Most cleanly, what is checked are those anticipations per se. But there is still also a reliance on passive models of perception (generally based on sensations) and restricted models of action and construction.

« 3 » Filipo Perotto’s CALM is a significant advance within this framework, especially in its ability to extract anticipatory information from an only partially observable and not fully deterministic world, and to use synthetic elements in doing so. It is important to demonstrate that these more realistic framework assumptions can be handled, and to show how they can be handled.

« 4 » But CALM, too, is built on sensation models of perceiving and on singleton actions. One of the current foci for development of the CALM model is to develop possibilities of chaining schemas – again, I would agree that this is exactly the right direction. I would like to comment, however, on an even more general approach that might be considered – a fully interactive approach.

« 5 » Consider that passive sensations, insofar as they exist at all, functionally serve to detect properties of the environment, and that such detection – as a strictly factual matter – is all that is functionally relevant to the system. In particular, such detections need not be understood to *represent* that which is detected in order to account for their influences on system processing. Still further, such detections can also be realized by fully interactive processes, not just by passive receptive processes (Bickhard & Richie 1983). On the other hand, anticipations concerning possible interactions with the environment also can, and arguably do, occur with respect to *whole patterns* of interaction, not just singular actions. Chaining of schemas is precisely a step in this direction, but it requires more than "chains" to be able to model general interactive patterns.

« 6 » So, I would suggest that patterns of interaction can serve:

1 | detection functions, rather than sensations and perceptions interpreted as representational (with all of the classic problems that that interpretation entails: Bickhard 2009b),[2] *and*

2 | as patterns of interaction that are anticipated as possible in the future, *and*

3 | as patterns that can be tentatively constructed in learning more about the environment – learning more about what patterns of interaction can be anticipated as possible, given what prior interactive detections have already occurred.

« 7 » Such shifts generate a dynamic systems model, more than a classic computational model, but one in which representation is not absent:. truth value emerges in anticipations that are capable of being true or false, and cognitive representation more generally can be built from organizations of such anticipations (Piaget 1954; Bickhard 2009b). In such a model, representation is not built on or out of presumed sensations as representations.

« 8 » Modeling the dynamics of such dynamic systems is difficult. For one class of problems, there are no topological or metric spaces built in to serve as spaces for generalization. On the other hand, if the construction of such topological spaces can itself be constructed, then we can model the cognitive development and organization and re-organization of such spaces in children and adults – a higher level of learning and development than is usually addressed (Bickhard & Campbell 1996). For another class of problems, cognitive representations of, for example, objects or numbers, cannot be simply presupposed, but must themselves be constructed. However, that was among the basic insights of Piaget some time ago (Piaget 1954; Allen & Bickhard 2011, 2013a, 2013b, 2013c).

« 9 » Overall, then, moving to a fully interactive dynamic systems framework makes a number of modeling problems much more difficult. But it offers advantages of avoiding classic problems concerning, for example, the nature of representation (Bickhard 2009b), and offers direct approaches to modeling phenomena that are very difficult to approach within standard frameworks (e.g., re-organizing the topology of representational spaces in response to understanding an analogy; Bickhard & Campbell 1996).

2 | Note that this also frees the model from being able to generalize only along the dimensional variables that are built into the sensation apparatus, and from the related built-in metric spaces for error, etc. Of course, it also makes the dynamics of such generalization more difficult to model.

Mark Bickhard is the Henry R. Luce Professor in Cognitive Robotics and the Philosophy of Knowledge at Lehigh University. He is affiliated with the Departments of Philosophy and Psychology, and is Director of the Institute for Interactivist Studies. His work focuses on the nature and development of persons, as biological, psychological, and social beings.

Received: 11 October 2013
Accepted: 21 October 2013

Representing Knowledge in a Computational Constructivist Agent

Thomas Degris
ENSTA ParisTech-INRIA, France
thomas.degris/at/inria.fr

> Upshot • The aim of this commentary is to relate the target article to recent work about how to represent the knowledge acquired from experience by a constructivist agent.

« 1 » Constructivist agents acquire new knowledge and maintain existing knowledge by experimenting with their environment. A key question is then how to represent knowledge for such an agent.[3] In the target paper, knowledge that can be created and updated from data is emphasized, but a different mathematical framework and a different architecture, namely the Horde architecture (Sutton et al. 2011), is used. This commentary presents the similarities and differences between the target paper and the Horde architecture.

« 2 » Both papers focus on a situated agent embedded in its environment. The agent does not have access to the full state of the environment. To be able to understand better its interaction with the environment, the agent needs to construct abstract internal structures from a low level sensorimotor loop. In both papers, the internal representation built by the agent comes from its own experience and does not need to match an arbitrary absolute representation of its environment.

« 3 » The target paper has chosen to represent the knowledge of the agent with tree-structured representations. While trees can, in principle, take advantage of specific structures in the data, they also have issues that can make them impractical to use as a lifelong constructivist agent in the actual world. More specifically, as mentioned in the target paper, the main idea of structured representations is that the system dynamics can be factored to save memory and computational time. But such structure just may not be in the data. For instance, if one would like to predict the next value of a bump sensor on a small mobile robot, it is likely that information from all the sensors on the robot, as well as many abstract representations, may help in one way or another to make a better prediction. Thus, a prediction as simple as the value of a bump sensor may simply not be factorable. Moreover, even when some of the system dynamic may be factorable, there is no guarantee that other representations, such as value functions or policies, will be factorable (Boutilier, Dearden & Goldszmidt 2000). For an agent to take complex decision or to understand a complex environment, perhaps it is unavoidable to consider a large number of variables or signals. In comparison, the Horde architecture focuses on algorithms with a complexity in computation

3 | Joseph Modayil and I proposed an answer to that question in their presentation "Scaling-up knowledge for a cognizant robot" at the AAAI Spring Symposium on Designing Intelligent Robots: Reintegrating AI, Stanford University, 2012.

and memory that is linear to the number of parameters to learn. In practice, thousands of predictions depending on thousands of features can be learned online in real time on an actual robot (Modayil, White & Sutton 2012).

« 4 » The agent in the target paper constructs a set of schemas to build a predictive representation. A schema takes a context and an action to make a prediction about the next time step. A context can be seen as a set of conditions on the agent state; that is, conditions on internal variables and the last observation from the sensors. The action in the schema describes what the agent will do to go to the next time step. Thus, knowledge constructed by the agent answers questions such as: "Am I going to be connected to my docking station at the next time step if I do this action?" In comparison, demons in the Horde architecture can represent knowledge similar to schemas but also more general knowledge: for a given agent state and a policy – that is, a sequence of (stochastically chosen) actions – a demon makes a temporally abstract prediction. For instance, an agent can construct the answer to questions such as: "What is the probability of being connected if I follow the policy going back to my docking station?" or "How much energy will I use to go back to my docking station?" Moreover, there are two additional advantages with temporal abstractions. First, demons can be used to build predictive features in the agent internal state. Predictive state representations (PSRs) are known to be more general than POMDPs or n^{th}-order Markov models – representations based on history (Singh, James & Rudary 2004). Second, it becomes possible to consider high-level planning on temporal abstractions, as has been proposed with the option framework (Sutton, Precup & Singh 1999).

« 5 » Overall, the Horde architecture has two key features compared to the representation used in the target paper. First, Horde can learn and maintain predictive knowledge online and in real time. Second, Horde can learn answers to temporally-abstract questions. Thus, the Horde architecture is a direct possible answer to two of the questions mentioned at the end of the target paper: how to extend the work to stochastic and continuous environments and how to consider action sequences. Of course, the Horde architecture asks its own set of questions: What are the criteria to create or delete demons based on data and experience? What should be the behavior of an agent in its environment to optimize learning in demons (intrinsic motivations)? The path to a constructivist agent for a general artificial intelligence remains uncertain.

Thomas Degris is a postdoctoral fellow in the Flowers team from ENSTA ParisTech-INRIA. He did his PhD at University Pierre et Marie Curie on factored Markov decision processes. He also worked as a postdoctoral fellow in the RLAI group at the University of Alberta.

Received: 13 October 2013
Accepted: 18 October 2013

Some Comments on the Relationship Between Artificial Intelligence and Human Cognition

Bernard Scott
Centre for Sociocybernetics Research, Bonn, Germany
bernces1/at/gmail.com

> Upshot • In making a contribution to artificial intelligence research, Perotto has taken note of work on human cognition. However, there are certain aspects of human cognition that are not taken into account by the author's model and that, generally, are overlooked or ignored by the artificial intelligence research community at large.

« 1 » In his paper, Filipo Perotto has taken note of work on human cognition. In particular, he references Jean Piaget (§6) and Ernst von Glasersfeld (§7). The former developed his "genetic epistemology" by studying the development of human children. The latter, using Piaget as one of his main sources, developed "radical constructivism," a philosophical treatise about how humans come to know. Rather than attempt to position the author's work in the broad field of artificial intelligence research, something I do not feel confident to do without further reading, I wish to note aspects of human cognition that the author's model does not take into account and that, generally, are overlooked or ignored by the artificial intelligence research community at large.

« 2 » First, I note that humans, like other biological organisms, are dynamical systems (§11), far from equilibrium, whose structures are continually being formed and reformed by the dissipation of energy. The author does state that humans are dynamical systems; however, his account is limited to the statement (in footnote 3) that "A dynamical system consists of an abstract state space evolving in time according to a rule that specifies the immediate future state given the current state." Far richer concepts of what dynamical systems are and the challenges of modeling them are to be found, for example, in the writings of Heinz von Foerster (2003: chapter 1) and Ilya Prigogine (1981) on self-organisation. Humans are also organisationally closed, autopoietic systems, endowed with an operationally closed nervous system. Using these foundational ideas, Humberto Maturana and Francisco Varela (1980) developed a "biology of cognition." This work adds considerably to our understanding of constructive cognitive processes. Related ideas are to be found in chapters 8, 10 and 11 of von Foerster (2003), where there is discussion of how sensorimotor activity leads to the computation of "objects" as an invariant of an organism's constructed reality. These works are seminal accounts of what is referred to in later literature as "enactive cognition."

« 3 » The author uses the term "symbolic" in §9. The author's model is presented as a general mechanism for learning. It, like much other work in artificial intelligence research, ignores or takes for granted that which is, with few exceptions, peculiarly human in human cognition: the ability to communicate and compute using what George Herbert Mead refers to as "significant symbols" – gestures, icons and utterances that call forth in the sender similar response to those elicited in the receiver. Humans converse with each other and converse with themselves. This truth falsifies the claim made by many in the artificial intelligence community that brains and computers are both "physical symbol systems."

Other critics (e.g., Searle 1980) have also challenged this claim. Scott & Shurville (2011) provide an extended discussion of the topic and propose its falsification based on their analysis that a "symbol" is a second-order "object" that two or more interacting organizationally closed systems compute as standing for a given first-order "object" and compute that they are both doing so.

« 4 » In his model, the author of the target paper refers to his simulated agent as having a "mind" (§13). If we take "mind" to refer to the conceptual processes that constitute humans as individual selves, then it is possible find in the literature more elaborated understandings of "mind" as an embodied, organizationally closed, self-reproducing system of concepts that arises as a consequence not only of ongoing cognitive constructions but also of social interaction (Pask, Scott & Kallikourdis 1975; Pask 1981; Scott 2007).

Bernard Scott completed a Ph.D. in cybernetics from Brunel University, UK, in 1976. His supervisor was Gordon Pask, with whom he worked between 1967 and 1978. Among other positions, Bernard is a Fellow of the UK's Cybernetics Society and Past President of Research Committee 51 (on sociocybernetics) of the International Sociological Association.

Received: 14 October 2013
Accepted: 31 October 2013

Author's Response: Evaluating CALM

Filipo Studzinski Perotto

> **Upshot** • In this response, I address the points raised in the commentaries, in particular those related to the scalability and robustness of the mechanism CALM, to its relation with the CAES architecture, and to the transition from sensorimotor to symbolic.

General claims

« 1 » The commentators have touched important points in the ideas presented in the article. Some of the criticisms made might appear heavy, but this is due to the nature of this research, not limited to technical applied AI questions, which aims to address challenging philosophical and scientific problems.

« 2 » Since I declared in the beginning of the paper that until now, Constructivist AI has not been able to present "impressive results," I led the readers to expect some spectacular results. However, the stated experimental outcomes (with the hyper-flip problem) can rather disappoint such expectations. The assertion might also give the wrong impression that I considered Constructivist AI stagnant until the arrival of this article. I am in complete agreement with **Georgi Stojanov** when he says that constructivist AI "certainly made huge theoretical advances" (§1), and I would add that AGI has incorporated several concepts from the constructivist approach, even if those researchers do not necessarily call themselves constructivists. **Stojanov** says that such critique would be justified "if the expected concrete result was to build an artifact that would exhibit the behavior of a three-year-old infant" (§1). To date, we are not able to do so, neither within the constructivist approach, nor with any other form of AI.

« 3 » In the long road towards constructivist artificial general intelligence, my article aims to be just a step forward, but it is still far from the finishing line. The ideas presented make up just a further brick for constructing the bridge, and not a complete definitive answer. As **Stojanov** says, this is already "a formidable challenge."

« 4 » As is often the case with most of these ambitious investigations, the work done until now left more open hypotheses, unanswered questions, ideas and promises, than actually determined conclusions or remarkable results. Nevertheless, thanks to that ambition, it is possible to believe that the work done, albeit quite modest, points in the right direction.

« 5 » Despite all the efforts, we still find ourselves stuck between two steep challenges. On the one hand, there is the complexity of the sensorimotor problems, which require computationally viable models capable of treating large continuous domains and realizing cybernetic adaptation, interactive processing of imprecision, refinement of skills, etc. On the other hand, there is the problem of constructing symbols to represent abstract entities and processes, which could lead the agent to a kind of higher level of thought in which the experience is organized in terms of intelligible concepts. The ideas presented in my article do not solve either of these challenges but could eventually help AI to get a foothold in both.

Scalability and robustness

« 6 » The first important question cited many times in the commentaries can be summarized like this: can the mechanism scale up well to complex, continuous, large-order, real-time, noisy, non-deterministic environments? In other words: can the viability of the model be convincingly demonstrated in an experimental way? As claimed in the introduction to my article, so far nobody has been able to do this in constructivist general artificial intelligence.

« 7 » CALM, too, suffers from scalability difficulties. It can scale up well on highly structured environments, where the agent deals with a large number of variables but where causal links are very precise, where relevant variables in function of the agent's goals are easy to identify, and where non-observable variables exist on a very small scale. I agree that it is easy to be robust in such environments. Since CALM was designed to work in discrete symbolic environments, it is not adapted to be directly applicable to large sensorimotor problems.

« 8 » **Frank Guerin** (§6) suggests the example where two stereo cameras deliver a few million pixels in 24 bit color at thirty frames per second and CALM tries to predict the consequences of actions in the complexity of an everyday setting. He wonders whether each bit of input could be used as a CALM variable. Admittedly, CALM is not adapted to face this kind of problem.

« 9 » I believe that for CALM to deal with sensorimotor problems of larger magnitudes, a bigger architecture must be developed. This includes tools that can segregate "continuous realms into meaningful and purposeful symbol systems" (Martin Butz §9) processing sensorimotor signals before linking them with other CALM modules. In humans, what is conveyed from the eyes to the association areas is more than a matrix of pixels, and is rather information about lines, contours, contrasts, movements, basic forms, etc.

« 10 » The human brain is not a flat system processing all signals at once, but is divided in several zones and layers that are more or less specialized. Sensory and motor data are processed in primary areas before being integrated in the association zones of the neural cortex (Tortora & Derrickson 2012). Roughly speaking, perhaps CALM can be related to the association cortex rather than to the sensorimotor cortex. Guerin puts the right question when asking "where to make the cut-off between what the core CALM system sees and what is the responsibility of other abstraction mechanisms" (§6). For now, this question must remain unresolved.

Experimental scenario

« 11 » The experimental problem used in my article (hyper-flip), although adequate for illustrating the mechanism's capabilities, is admittedly too simplistic. It remains a tricky toy problem, which can demonstrate neither the algorithm's robustness nor how to solve concrete problems. It would be necessary to conduct more sophisticated experiments to show that CALM could be able to discover unobservable and relevant environmental properties, representing and using them efficiently as synthetic elements in its world model when facing more complex problems.

« 12 » I can only agree with Kristinn Thórisson when he says that "the aim of AI is not just to speculate but to build working, implemented systems" (§6) and that "for any engineering effort to be taken seriously, the requirement for experimental evaluations of (physical and/or virtual) running software systems cannot go ignored" (§8).

« 13 » I am also in agreement with Butz when he declares in the upshot that "sensory as well as motor noise is ubiquitous in our environment" and that "symbols do not exist a priori but need to be grounded within our continuous world." Simple high-level symbol manipulation problems that ignore the low-level sensorimotor challenges can hardly lead to a system that may convincingly become cognitive.

« 14 » Like several other researchers, I believe that the domain *par excellence* for testing machine learning models is robotics. A simple robot with continuous noisy sensors in real-time action into the physical world is a fantastic challenge for such general AI systems. In its current stage, CALM is not yet ready to face that kind of problem successfully. In the following paragraphs, I would like to address some of the directions I can envisage it taking in order to move forward.

From continuous signals to discrete representations

« 15 » One of the main limitations of CALM is the need for a predefined discrete representation of both the signals received and those transmitted by the agent. Also, time is considered as a discrete succession of cycles. However, many problems in complex environments can only be properly addressed through continuous representations, which enable an agent to face problems on the sensorimotor level.

« 16 » It seems more natural to start with continuous signals and gradually construct discrete states as a sort of abstraction. This is the first step to going beyond sensorimotor primitives and arriving in a symbolic dimension. It also applies to temporal abstraction because intelligence needs to slice the continuous flow into relevant pieces of time in order to recognize events or cycles.

« 17 » In any case, schema learning mechanisms are not necessarily incompatible with continuous environments. An extension of the schema used by CALM can be used to represent changes in continuous variables. Very basically, we can represent the anticipation of an increase or decrease in the value of a certain variable, or the tendency to converge towards some specific value, given some action. In this way, each schema realizes a kind of simplified regression, where in function of some conditions (context and action), the schema can anticipate a continuous variation of some variable.

Noise and non-determinism

« 18 » The definition and exploration of environments that I called "partially deterministic" (e.g., §2) should be considered worthwhile. The methods behind CALM were defined to focus on the discovery of deterministic regularities in an environment composed of deterministic and non-deterministic phenomena.

« 19 » For an agent, a complex environment can appear non-deterministic because its perception, control and understanding are limited in some way (partial observability, noisy sensors, imprecise effectors, other entities acting in the same environment, too many complex causal relations, etc.). Apparent non-determinism can be modeled either by creating stochastic rules, or by continuing to search for causes.

« 20 » Every roboticist knows the importance of taking the noise and imprecision inherent in sensory and motor apparatus seriously. So far, CALM has not been equipped with any mechanism to treat noise explicitly. Even so, the presence or absence of noise could be represented as a cause of some perturbed anticipations, as in: $a \wedge \sim noise \rightarrow b$.

« 21 » That said, the possibility of representing certain situations as stochastic regularities could be incorporated into CALM, working as a complementary method for situations where deterministic assumptions are not possible. Such a method can allow the mechanism to search for probabilities in order to anticipate which is most likely to happen. Nevertheless, that kind of search for statistical regularities should not interfere with the search for causal relations.

Robustness and parallelization

« 22 » Thomas Degris says that for "an agent to take complex decisions or to understand a complex environment, perhaps it is unavoidable to consider a large number of variables or signals" (§3). This is certainly correct, especially when the problem is close to the sensorimotor level.

« 23 » As Stojanov claims, CALM resembles a completely deterministic "monolithic single-thread algorithm" (§3). In nature, animal as well as human brains do not operate as a centralized hierarchy, but more like cooperative and concurrent modules working simultaneously in several levels, and not necessarily in complete, harmonious coherence. The power of intelligence stems from the diversity of many effective imperfect methods, and decisions emerge from conflicts and negotiations among them (Minsky1988: 308).

« 24 » I believe that any good constructivist AI program will have to end up being more or less in accordance with that systemic modular perspective of the mind, where learning will appear as a continual construction and reconstruction of modules, each one working in a specific level and domain, but in constant interaction with other modules. Once within this conjuncture, CALM could be imagined as the engine inside some modules, under the baton of some principle responsible for coordinating the modules in the whole system.

« 25 » Moreover, concerning robustness, parallelization is a very powerful means to break complexity and to deal with complex environments. The neural organization and functioning of the brain is highly parallelized. Although it was not mentioned in my article, CALM can implement a kind of parallelization, since the construction of each anticipatory tree (that models the dynamics of one single variable) can be realized independently from the other trees, i.e., in different separated threads.

« 26 » Another way to be robust is to pay attention to what is important (Foner & Maes 1994). The problem of indistinctly correlating actions with changes in sensor data is computationally unfeasible for any non-trivial application. This problem becomes more manageable by restricting the set of sensor data the agent attends to, or the set of internal structures that is updated, at particular instants. In the same vein, CALM implements a focus of attention related to the affectively important variables.

« 27 » **Degris** writes that "while trees can, in principle, take advantage of specific structures in the data, they also have issues that can make them impractical to use as a life-long constructivist agent in the actual world" (§3) and that "even when some of the system dynamic may be factorable, there is no guarantee that other representations, such as value functions or policies, will be factorable" (ibid). Some technical choices with regard to CALM's methods should be revised, especially with regard to the management of episodic memory and anticipatory trees. It is evident that a robust algorithm for such general purposes must be carefully studied. In other words, the algorithms in CALM will probably need certain improvements.

« 28 » **Degris** cites the "Horde" architecture, suggesting that it can "represent knowledge similar to schemas but also more general knowledge" (§4). Horde can construct "demons," which are generalized value-functions for given partial policies. Those demons can be learned in parallel by an efficient extended reinforcement learning method during the actuation of the agent. However, I think that the knowledge represented by Horde is not that similar to what is represented by CALM.

« 29 » Space does not allow for a more detailed comparison between CALM and Horde. However, it is evident that architectures like Horde will be a precious source of good strategies for dealing with large real-time sensorimotor problems, translating them, when necessary, into symbolic terms. I believe that, the crucial problem of using efficient forms of representation aside, the most important challenge is to find a way to connect consistently the sensorimotor (continuous, noisy, real-time, large scale) domains to basic symbolic domains, and the latter to more abstract ones.

From lower to higher levels

« 30 » Another major question repeatedly mentioned in the commentaries is the passage from lower levels of interaction, based on sensorimotor primitives, to higher levels, based on abstract concepts. The question can be formulated like this: Is CALM able gradually to construct successive layers of abstraction in order to represent its knowledge?

« 31 » According to Jean Piaget (1957), from a fragmented sensorimotor universe, intelligence builds elementary notions, defines relations, finds regularities and eventually constructs an objective, substantial, spatial, temporal, regular and external universe, independent of the subject itself. A subjective "reality" will emerge from the increasing coherence between schemas in the course of these adaptions.

« 32 » In §3, **Stojanov** claims that CALM does not provide a way to build more abstract structures from simpler sensorimotor interactions. At least he recognizes that CALM is able to create synthetic elements that enlarge the sensorial context with something that is beyond perception. Even if this is simple, the synthetic elements are certainly a form of abstraction since they do not correspond to any sensory input. However, once CALM places the synthetic elements side by side with the sensorial ones, it does not create layers. The context is represented as a single flattened array. Evidently, we cannot go too far without some kind of robust structuring mechanism in order to organize knowledge into different levels.

« 33 » In human beings, cognition is in some way the construction of several layers of abstraction in order to understand and interpret experiences. If this process is compared with flying from the Earth to the Moon, the inference of synthetic elements would correspond to the takeoff. It does not give us too much altitude but it is crucial to start the voyage.

« 34 » Building synthetic elements does not constitute a form of abstract or symbolic thought by itself, but such a process contains the basic insight of what we could call "concept invention." Synthetic elements allow the designation of entities that cannot be represented from combinations of direct sensory perceptions. Thus, the possibility of representing unobservable conditions is a breakthrough along the road from mere direct perception to more abstract forms of understanding.

Grounding symbolic concepts on sensorimotor flows

« 35 » **Guerin** correctly claims that high-level cognition is very much grounded in sensorimotor intelligence (§4). I believe that extracting significant symbolic concepts from interactive sensorimotor flows is one of the key challenges for AI today. The robotics community and the symbolic AI community can be seen as digging tunnels on the opposite sides of a mountain. Despite a lot of progress, a consistent integration of contributions from the two sides is still incipient. The same metaphor can be used to refer to the relation between neuroscientists and psychologists. As pointed out by **Guerin**,

> "there are some works in cognitive science that are beginning to attempt to address the issue of providing some theoretical framework to account for how a sensorimotor level can connect with higher levels of cognition." (§5)

In fact, the search for mechanisms capable of doing so will draw on the findings from all these fronts: high- and low-level, computational and cognitive sciences.

« 36 » Moreover, as **Thórisson** says,

> "due to the high number of combinatorics that a complex environment will produce, through countless interactions between its numerous elements, an agent must create *models* that isolate and capture some essence of underlying causes." (§3)

Following constructivist principles, I would suggest that the passage from the sensorimotor to the conceptual domain is possible through a series of abstractions where, at each step, a large number of localized, context-dependent, quick and small elements are coordinated in more general, independent elements. Because CALM is not able to do so, **Stojanov**'s claim in his upshot is correct: it can be seen, at best, "as a model of the empirical abstraction but not of the reflective abstraction."

« 37 » In the same vein, **Mark Bickhard** says that "CALM… is built on sensation models of perceiving and on singleton actions." (§4). I am in agreement with him when he claims that "anticipations concerning possible interactions with the environment… occur with respect to *whole patterns* of interaction, not just singular actions" (§5).

« 38 » Enabling the mind of an agent to learn and think in terms of "whole patterns of interactions" is another major challenge in AI. These patterns must be related in two ways:

1 | spatially, to high-level constructed objects, and
2 | temporally, to abstract events, i.e., "cognitive concepts for structuring experiences and thus for perceiving the environment in chunks that may be symbolizable" (**Butz** §8) or "temporally abstract prediction[s]" (**Degris** §4).

CALM and CAES

« 39 » **Thórisson** points to the lack of a clear connection between CALM and CAES models. **Stojanov** claims that the decision to introduce the two entities (body and mind) was somewhat arbitrary. Furthermore, **Bernard Scott** expressed his disappointment with the way the term "mind" was employed in my article. I agree that the relation between CALM and CAES was not developed in the paper, and that the hyperflip experiment does not illustrate that relation. So let me try to make up for this omission.

« 40 » CAES is an architecture that connects concepts from cybernetics, the theory of autopoiesis, dynamical systems, and affective AI. It is based on the definition of three entities: environment, body, and mind. CALM is the engine that plays the role of the cognitive system in the mind. Besides a cognitive system, the mind includes an affective system (responsible for evaluating the perceived situations), an emotional system (directed to the internal body states), and a reactive system (directed to the body effectors).

« 41 » Ross Ashby (1952) defined the organism (or the agent) as a system composed of a set of essential variables that must stay within a certain physiological normality (limits of viability) in order to preserve the system's integrity and, consequently, the organism's survival. A given behavior contributes to the agent's adaption if it ensures the persistency of these essential variables within its viable limits. The presence of essential variables assumes that the agent has something like an internal environment. That is the body (Parisi 2004).

« 42 » In nature, organism and environment can exert opposing forces with respect to the global system's flow. However, only the organism is at risk of disintegration, of disappearing as unity. A nondestructive dynamical coupling is reached in the relation between the two systems when the organism interacts with the environment in order to ensure its self-preservation.

« 43 » Randall Beer (1995) integrated the cybernetic concept of organism and autopoiesis using dynamical systems. The adaption criterion is abstractly represented as a zone in the space where the flow of the system must remain. The limits of adaption are the frontiers of that region within the global system space (composed by agent and environment), and the agent is considered adapted to the environment if its activity drives the global system's trajectory in such a way that it never escapes from those frontiers.

« 44 » I agree with **Scott** when he says that "humans, like other biological organisms, are dynamical systems, far from equilibrium." Even if my article does not address this issue, CAES architecture was imagined to correspond to a definition of an agent as a system far from equilibrium, in the sense proposed by Bickhard (2009a) or Xabier Barandiaran and Alvaro Moreno (2008).

Intrinsic motivations and curiosity

« 45 » **Stojanov** observes that the intrinsic motivations exhibited by CALM "can be related to low-level physiological drives (hunger, pain-avoidance) with no possibility for development of more sophisticated forms of motivations such as curiosity" (§4). I agree that the motivation system of my model is still far too utilitarian, even though some effort has been made to build an intrinsically motivated agent, which is consistent from the perspective of an embodied AI. It is evident that motivation is also linked to the subject's activity itself. Drinking water because we are thirsty is a kind of behavior that can be easily anchored in a biologically-driven explanation. Other behaviors, such as playing checkers, listening to music, or writing scientific papers, can hardly be explained by simply referring to physiological needs.

« 46 » Nevertheless, CALM implements the notion of curiosity for exploratory behavior. In naive AI approaches, curiosity usually means doing random actions from time to time. In CALM, there is a measure of exploratory utility that allows the agent to plan actions that may lead to new discoveries, or new knowledge that would enhance its world model. The mechanism follows two behavioral policies: one to optimize the affective signals, and another to optimize the gain of knowledge related to relevant variables. The choice of what action to do depends on the weighting of these two policies.

Language and reflectivity

« 47 » Thórisson claims that artificial general intelligence cannot be done "without some form of self-programming on the part of the machine, which in turn cannot be achieved without transparency of its operational semantics" (§4) and finishes by questioning the power of the CALM schema formalism "to support models of self… and their ability to support self-inspection" (§6). Similarly, Scott says that "CALM ignores or takes for granted that which is… peculiarly human in human cognition: the ability to communicate and compute using… 'significant symbols'" (§3).

« 48 » It is evident that those capacities are the notable characteristics of high-level intelligence. But to stay in accordance with constructivist principles, I believe that those abilities emerge as a result of the process of learning and interpreting the experiences on abstract levels. Piaget (1953) suggested that the basic principles regulating intellectual functioning remain unchanged over a lifetime, and that increasingly refined skills and knowledge result from the gradual complexification of the underlying constructed knowledge structures.

« 49 » To summarize my point, I do not believe that the absence of language or self-investigation represents a particular lack in the mechanism. In fact, the simplicity of the problems faced by CALM for now, as well as its non-existent capacity for creating different layers of abstraction to interpret its experiences, do not allow the agent to have the faculty for doing language or self-awareness.

Conclusion

« 50 » The commentaries pointed out many aspects of my model that can be improved, such as the lack of sensorimotor grounding of the symbolic elements manipulated by CALM and the impossibility to create more abstract levels of knowledge to represent the agent's experience. I am confident that improved versions of my model will be able to deal with these problems, in particular extending it to be modular. In this way, the agent will be equipped with more refined sensorimotor apparatus, capable of realizing some pre-processing of signals, coupled with other modules capable of doing some preliminary computing in order to solve some basic sensorimotor problems at a low level, filtering the data that must be sent to the first symbolic modules. Finally, a more sophisticated form of abstraction needs to be incorporated into the algorithm in order to allow the construction of an organized structure of anticipatory modules, acting at different levels of abstraction.

Acknowledgements

I would like to thank all of the commentators for this very enriching discussion, and also thank the editors for this valuable opportunity.

Received: 5 November 2013
Accepted: 8 November 2013

Combined References

Allen J. W. P. & Bickhard M. H. (2011) Normativity: A crucial kind of emergence. Human Development 54: 106–112.
Allen J. W. P. & Bickhard M. H. (2013a) Stepping off the pendulum: Why only an action-based approach can transcend the nativist-empiricist debate. Cognitive Development 28: 96–133.
Allen J. W. P. & Bickhard M. H. (2013b) The pendulum still swings. Cognitive Development 28: 164–174.
Allen J. W. P. & Bickhard M. H. (2013c) Beyond principles and programs: An action framework for modeling development. Human Development 56: 171–177.
Anderson M. (2003) Embodied cognition: A field guide. Artificial Intelligence 149(1): 91–130.
Ashby W. R. (1952) Design for a brain. First Edition. Chapman & Hall, London.
Barandiaran X. & Moreno A. (2006) On what makes certain dynamical systems cognitive. Adaptive Behavior 14(2): 171–185.
Barandiaran X. & Moreno A. (2008) Adaptivity: From metabolism to behavior. Adaptive Behavior 16(5): 325–344.
Barsalou L. W. (1999) Perceptual symbol systems. Behavioral and Brain Sciences 22: 577–600.
Barsalou L. W. (2008) Grounded cognition. Annual Review of Psychology 59: 617–645.
Beer R. D. (1995) A dynamical systems perspective on agent-environment interactions. Artificial Intelligence 72: 173–215.
Beer R. D. (2004) Autopoiesis and cognition in the game of life. Artificial Life 10(3): 309–326.
Beetz M., Jain D., Mösenlechner L. & Tenorth M. (2010) Towards performing everyday manipulation activities. Robotics and Autonomous System 58(9): 1085–1095.
Bellman R. A. (1957) Markovian decision process. Journal of Mathematics and Mechanics 6: 679–684.
Bickhard M. H. (2009a) Interactivism: A manifesto. New Ideas in Psychology 27: 85–95.
Bickhard M. H. (2009b) The interactivist model. Synthese 166(3), 547–591.
Bickhard M. H. & Campbell R. L. (1996) Topologies of learning and development. New Ideas in Psychology 14(2): 111–156.
Bickhard M. H. & Richie D. M. (1983) On the nature of representation: A case study of James Gibson's theory of perception. Praeger Publishers, New York.
Blum A. L. & Langley P. (1997) Selection of relevant features and examples in machine learning. Artificial Intelligence 97: 245–271.
Boutilier C., Dearden R. & Goldszmidt M. (2000) Stochastic dynamic programming with factored representations. Artificial Intelligence 121(1–2): 49–107.
Bril B., Roux V., Dietrich G. (2005) Stone knapping: Khambhat (India), a unique opportunity? In: Roux V. & Bril B. (eds.) Stone Knapping: The necessary conditions for a uniquely hominid behaviour. McDonald Press, Cambridge MA: 53–72.
Brooks R. A. (1991) Intelligence without representation. Artificial Intelligence 47: 139–159.
Buisson J.-C. (2004) A rhythm recognition computer program to advocate interactivist perception. Cognitive Science 28(1): 75–87.

Butz M. V. (2002) Anticipatory learning classifier systems. Kluwer, Boston MA.

Butz M. V. (2008) How and why the brain lays the foundations for a conscious self. Constructivist Foundations 4(1): 1–14 & 32–37. Available at http://www.univie.ac.at/constructivism/journal/4/1/001.butz

Butz M. V. (2013) Separating goals from behavioral control: Implications from learning predictive modularizations. New Ideas in Psychology 31: 302–312.

Byrne R. W. (2005) The maker not the tool: The cognitive significance of great ape manual skills. In: Roux V. & Bril B. (eds.) Stone Knapping: The necessary conditions for a uniquely hominid behaviour. McDonald Press, Cambridge MA: 159–169.

Campbell D. T. (1974) Evolutionary epistemology. In: Schilpp P. A. (ed.) The philosophy of Karl Popper. Open Court, LaSalle IL: 413–463.

Campbell R. L. & Bickhard M. H. (1993) Knowing levels and the child's understanding of mind. Behavioral and Brain Sciences 16(1): 33–34.

Cassandra A. R., Kaelbling L. P. & Littman M. L. (1998) Planning and acting in partially observable stochastic domains. Artificial Intelligence 101: 99–134.

Chaput H. (2004) The constructivist learning architecture. Unpublished PhD Thesis, University of Texas.

Conant R. C. & Ashby W. R. (1970) Every good regulator of a system must be a model of that system. International Journal of Systems Science 1(2): 89–97.

Crook P. & Hayes G. (2003) Could active perception aid navigation of partially observable grid worlds? In: Proceedings of the 14th European Conference on Machine Learning (ECML). Lecture Notes in Artificial Intelligence 2837. Springer-Verlag, Berlin: 72–83.

Degris T. & Sigaud O. (2010) Factored markov decision processes. In: Buffet O. & Sigaud O. (eds.). Markov decision processes in artificial intelligence. Wiley-ISTE, London: 99–125.

Degris T., Sigaud O. & Wuillemin P.-H. (2006) Learning the structure of factored markov decision processes in reinforcement learning problems. In: Proceedings of the 23rd International Conference on Machine Learning. ACM Press: 257–264.

Drescher G. (1991) Made-up minds: A constructivist approach to artificial intelligence. MIT Press, Cambridge MA.

Feinberg E. A. & Shwartz A. (2002) Handbook of Markov decision processes: Methods and applications. Kluwer, Norwell MA.

Foerster H. von (2003) Understanding understanding: Essays on cybernetics and cognition. Springer, New York.

Foner L. N. & Maes P. (1994) Paying attention to what's important: Using focus of attention to improve unsupervised learning. In: Cliff D., Husbands P., Meyer J. & Wilson S. (eds.) From animals to animats 3. Proceedings of the Third International Conference on Simulation of Adaptive Behavior. MIT Press, Cambridge MA: 256–265.

Glasersfeld E. von (1974) Piaget and the radical constructivist epistemology. In: Smock C. D. & Glasersfeld E. von (eds.) Epistemology and education. Follow Through Publications, Athens GA: 1–24. Available at http://www.vonglasersfeld.com/034

Glasersfeld E. von (1979) Cybernetics, experience, and the concept of self. In: Ozer M. N. (ed.) A cybernetic approach to the assessment of children: Toward a more humane use of human beings. Westview Press: Boulder, CO, pp. 67–113. Available at http://www.vonglasersfeld.com/056/

Glasersfeld E. von (2005) 30 years of radical constructivism. Constructivist Foundations 1(1): 9–12. Available at http://www.univie.ac.at/constructivism/journal/1/1/009.glasersfeld

Goldstein J. (1999) Emergence as a construct: History and issues. Emergence: Complexity and Organization 1: 49–72.

Guerin F. (2011) Learning like baby: A survey of AI approaches. Knowledge Engineering Review 26(2): 209–236.

Guestrin C., Koller D. & Parr R. (2001) Solving factored POMDPs with linear value functions. In: Proceedings of the Workshop on Planning under Uncertainty and Incomplete Information. Morgan Kaufmann, San Francisco CA: 67–75.

Guestrin C., Koller D., Parr R. & Venkataraman S. (2003) Efficient solution algorithms for factored MDPs. Journal of Artificial Intelligence Research 19: 399–468.

Hansen E. A. & Feng Z. (2000) Dynamic programming for POMDPs using a factored state representation. In: Proceedings of the Fifth International Conference on Artificial Intelligence, Planning and Scheduling. AAAI Press, Menlo Park CA:130–139.

Hauskrecht M. (2000) Value-function approximations for partially observable Markov decision processes. Journal of Artificial Intelligence Research 13: 33–94.

Hoffmann J. (2003) Anticipatory behavioral control. In: Butz M. V., Sigaud O. & Gérard P. (eds.) Anticipatory behavior in adaptive learning systems: Foundations, theories, and systems. Springer, Berlin: 44–65.

Holmes M. & Isbell C. (2005) Schema learning: Experience-based construction of predictive action models. Advances in Neural Information Processing Systems 17: 562–585.

Holmes M. & Isbell C. (2006) Looping suffix tree-based inference of partially observable hidden state. In: Cohen W. W. & Moore A. (eds.) Proceedings of the 23rd International Conference on Machine Learning (ICML 2006). ACM Press, New York: 409–416.

Hommel B., Müsseler J., Aschersleben G. & Prinz W. (2001) The theory of event coding (TEC): A framework for perception and action planning. Behavioral and Brain Sciences 24: 849–878.

Indurkhya B. (1992) Metaphor and cognition: An interactionist approach. Kluwer, Dordrecht.

Joas H. (1993) American pragmatism and German thought: A history of misunderstandings. In: Joas H., Pragmatism and social theory. University of Chicago Press, Chicago: 94–121.

Jonsson A. & Barto A. (2005) A causal approach to hierarchical decomposition of factored MDPs. In: De Raedt L. & Wrobel S.(eds.) Proceedings of the 22nd International Conference on Machine Learning (ICML 2005). ACM Press, New York: 401–408.

Kitchener F. K. (1986) Piaget's theory of knowledge. Yale University Press, New Haven.

Maturana H. R. & Varela F. J. (1980) Autopoiesis and cognition. Reidel, Dordrecht.

Minsky M. (1988) The society of mind. Simon & Schuster, New York.

Modayil J., White A. & Sutton R. S. (2012) Multi-timescale nexting in a reinforcement learning robot. In: Ziemke T., Balkenius C. & Hallam J. (eds.) Proceedings of the Conference on the Simulation of Adaptive behavior (SAB'12). Springer, Heidelberg: 299–309.

Morrison C., Oates T. & King G. (2001) Grounding the unobservable in the observable: The role and representation of hidden state in concept formation and refinement.

In: Working notes of AAAI Spring Symposium. AAAI Press, Menlo Park CA: 45–49.

Nivel E. & Thórisson K. R. (2013) Towards a programming paradigm for control systems with high levels of existential autonomy. In: Kühnberger K.-U., Rudolph S. & Wang P. (eds.) Artificial General Intelligence. Proceedings of the Sixth International Conference, AGI-13, Beijing, China, July/August 2013. Springer, Berlin: 78–87.

Nivel E., Thórisson K. R., Dindo H., Pezzulo G., Rodriguez M., Corbato C., Steunebrink B., Ognibene D., Chella A., Schmidhuber J., Sanz R. & Helgason H. P. (2013) Autocatalytic endogenous reflective architecture. Reykjavik University School of Computer Science Technical Report, RUTR-SCS13002.

Parisi D. (2004) Internal robotics. Connection Science 16(4): 325–338.

Pask G. (1981) Organisational closure of potentially conscious systems In: Zelany M. (ed.) Autopoiesis. North Holland Elsevier: New York: 265–307.

Pask G., Scott B. & Kallikourdis D. (1973) A theory of conversations and individuals (exemplified by the learning process on CASTE). International Journal of Man-Machine Studies 5: 443–566.

Pastra K. & Aloimonos Y. (2012) The minimalist grammar of action. Philosophical Transactions of the Royal Society B: Biological Sciences 367: 103–117.

Perotto F. S. (2010) Un mécanisme constructiviste d'apprentissage automatique d'anticipations pour des agents artificiels situés. Unpublished Ph.D Thesis at the Institut National Polytechnique de Toulouse, France.

Piaget J. (1953) The origin of intelligence in the child. Routledge & Kegan Paul. London.

Piaget J. (1954) The construction of reality in the child. Routledge & Kegan Paul, London. Originally published as: Piaget J. (1937) La construction du réel chez l'enfant. Delachaux et Niestlé, Neuchâtel.

Poupart P. & Boutilier C. (2004) VDCBPI: An approximate scalable algorithm for large scale POMDPs. In: Proceedings of the 17th Advances in Neural Information Processing Systems (NIPS). MIT Press, Cambridge MA: 1081–1088.

Prigogine I. (1980) From being to becoming. Freeman, San Francisco CA.

Puterman M. L. (1994) Markov decision processes: Discrete stochastic dynamic programming. Wiley, New York.

Rivest R. & Schapire R. (1994) Diversity-based inference of finite automata. ACM Journal 43(3): 555–589.

Schachner W. (ed.) (1996) Constructivism, interactionism and their applications. CEPIAG's Symposium Proceeding. CEPIAG, Geneva.

Schachner W., Real del Sarte O. & León C. (1999) Valore, indice e rappresentazione nella guida dei valori elementari. Metis 1: 7–22.

Scott B. (2007) The co-emergence of parts and wholes in psychological individuation. Constructivist Foundations 2(2–3): 65–71. Available at http://www.univie.ac.at/constructivism/journal/2/2-3/065.scott

Scott B. & Shurville S. (2011) What is a symbol? Kybernetes 48(1/2): 12–22.

Searle J. (1980) Minds, brains, and programs. Behavioural and Brain Sciences 3: 417–424.

Shani G., Brafman R. I. & Shimony S. E. (2005) Model-based online learning of POMDPs. In: Proceedings of the 16th European Conference on Machine Learning (ECML). Springer, Berlin: 353–364.

Shani G., Poupart P., Brafman R. I. & Shimony S. E. (2008) Efficient ADD operations for point-based algorithms. In: Proceedings of the 8th International Conference on Automated Planning and Scheduling (ICAPS). AAAI Press, Menlo Park CA: 330–337.

Sigaud O., Butz M. V., Kozlova O. & Meyer C. (2009) Anticipatory learning classifier systems and factored reinforcement learning. In: Pezzulo G., Butz M. V., Sigaud O. & Baldassarre G. (eds.). Anticipatory behavior in adaptive learning systems: From psychological theories to artificial cognitive systems. Springer, Berlin: 321–333.

Sim H. S., Kim K.-E., Kim J. H., Chang D.-S. & Koo M.-W. (2008) Symbolic heuristic search value iteration for factored POMDPs. In: Proceedings of the 23rd National Conference on Artificial Intelligence. AAAI Press, Menlo Park CA: 1088–1093.

Singh S., James M. & Rudary M. (2004) Predictive state representations: A new theory for modeling dynamical systems. In Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence. AUAI Press, Banff: 512–519.

Singh S., Littman M., Jong N., Pardoe D & Stone P. (2003) Learning predictive state representations. In: Proceedings of the 20th International Conference on Machine Learning (ICML). AAAI Press, Menlo Park CA: 712–719.

Stojanov G. (2009) History of usage of Piaget's theory of cognitive development in AI and robotics: A look backwards for a step forwards. In: Cañamero L., Oudeyer P.-Y. & Balkenius C. (eds.) Proceedings of the Ninth International Conference on Epigenetic Robotics. LUCS, Lund: 243–245. Available at http://www.lucs.lu.se/LUCS/146/

Stojanov G., Bozinovski S. & Trajkovski G. (1997) Interactionist-expectative view on agency and learning. IMACS Journal for Mathematics and Computers in Simulation 44: 295–310.

Stojanov G. & Kulakov A. (2011) Modeling attention within a complete cognitive architecture. In: Roda C. (ed.) Attention support in digital environments. Cambridge University Press, Cambridge: 210–241.

Stojanov G., Kulakov A. & Clauzel D. (2006) On curiosity in intelligent robotic systems. In: Proceedings AAAI 2006 Fall Symposium on Interaction and Emergent Phenomena in Societies of Agents, Arlington, Virginia. AAAI Press, Cambridge: 183–189.

Stolzmann W. (2000) An Introduction to anticipatory classifier systems. In: Lanzi P. L., Stolzmann W. & Wilson S. W. (eds.) Learning classifier systems: From foundations to applications. Springer, Berlin: 175–194.

Stoytchev A. (2009) Some basic principles of developmental robotics. IEEE Transactions on Autonomous Mental Development 1(2): 122–130.

Strehl A. L., Diuk C. & Littman M. L. (2007) Efficient structure learning in factored-state MDPs. In: Proceedings of the 22nd National Conference on Artificial Intelligence, AAAI Press, Menlo Park CA: 645–650.

Suchman L. A. (1987) Plans and situated actions: The problem of human-machine communication. Cambridge University Press, Cambridge.

Sutton R. S. & Barto A. G. (1998) Reinforcement learning: An introduction. MIT Press, Cambridge MA.

Sutton R. S., Modayil J., Delp M., Degris T., Pilarski P. M. White A. & Precup D. (2011) Horde: A scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In: Proceedings of the Tenth International Conference on

Autonomous Agents and Multiagent Systems (AAMAS'11), Volume 2. IFAAMAS, Taipei: 761–776.

Sutton R. S., Precup D. & Singh S. (1999) Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. Artificial Intelligence 112: 181–211.

Tani J. (1996) Model-based learning for mobile robot navigation from the dynamical systems. Perspective. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Special Issue on learning autonomous robots 26(3): 421–436.

Thórisson K. R. (2012) A new constructivist AI: From manual construction to self-constructive systems. In: Wang P. & Goertzel B. (eds.) Theoretical foundations of artificial general intelligence. Atlantis Press, Paris: 145–171.

Thórisson K. R. (2013) Reductio ad absurdum: On oversimplification in computer science and its pernicious effect on artificial intelligence research. In: Abdel-Fattah A. H. M. & Kühnberger K.-U. (eds.) Proceedings of the Workshop Formalizing Mechanisms for Artificial General Intelligence and Cognition (Formal MAGiC). Institute of Cognitive Science, Osnabrück: 31–35.

Thornton C. (2003) Indirect sensing through abstractive learning. Intelligent Data Analysis 7(3): 1–16.

Tortora G. J. & Derrickson B. H. (2012) Principles of anatomy and physiology. 13th Edition. Wiley, New York.

Varela F., Thompson E. & Rosch E. (1991) The embodied mind: Cognitive science and human experience. MIT Press, Cambridge MA.

Wilson R. & Clark A. (2008) How to situate cognition: Letting nature take its course. In: Aydene M. & Robbins P. (eds.). Cambridge handbook of situated cognition. Cambridge University Press, New York.

Wörgötter F., Aksoy E. E., Krüger N., Piater J., Ude A. & Tamosiunaite M. (2013) A simple ontology of manipulation actions based on hand-object relations. IEEE Transactions on Autonomous Mental Development 5: 117–134.

Yavuz A. & Davenport D. (1997) PAL: A constructivist model of cognitive activity. In: Riegler A., Peschl M. F. & von Stein A. (eds.) Proceedings of the international conference "New Trends in Cognitive Science: Does Representation Need Reality?" Vienna, Austria.

Ziemke T. (2003) What's that thing called embodiment? In: Alterman R. & Kirsh D. (eds.) Proceedings of the 25th Annual Meeting of the Cognitive Science Society. Lawrence Erlbaum, Mahwah NJ: 1305–1310.