Chapter 7
Self-Evolvability for Cognitive Systems

Abstract. The post-formal and closure aspects for cognitive developmental stages, geometry of logic, and relational complexity theories are presented.

Conceptual and computational frameworks are presented as polytopic cognitive architectures.

Physarum computing capabilities are evaluated.

7.1 Developmental Stages

Cognitive structures are patterns of physical or mental actions that underlie specific acts of intelligence and correspond to the stages of development (Piaget 1970, 1971).

According to Piaget, there are four primary cognitive development stages: sensory-motor, preoperational, concrete operational and formal.

Fig. 7.1 shows the developmental stages hierarchy.

It was observed that restriction of cognitive capability to the formal stage may correspond to systems stagnation and unavoidable failure (Yang and Bringsjord 2005).

This refers to automata that have a code or protocol that recommend some actions for situations requiring a completely different code.

Growing complexity imposes to look for creativity and self-evolvability for automata.

Piaget's epistemology made room for cognition beyond the fourth stage. Piaget initiated the study of post-formal stages, beyond the fourth, in which agents are able to operate over logical systems. This refers to meta-processing of logics and formal theories expressed in those logics. It was considered that elaboration of axiomatic schemas may be considered as surpassing the formal stage and are to formal schemas what the latter are to concrete operations (Piaget 1973).

The post-formal stages appeared as possible candidates for the so-called 5th cognitive development stage (Bringsjord et. al 2010). They are comparable to the formal framework in which post-formal reasoning involves the Self.

Fig. 7.2 shows a polytopic presentation of the cognitive developmental stages.

The initial four stages of Piaget, associated to S, K1, K2, and K3, have been completed in Fig. 7.2 by the self-evolvability stage. This allows describing systems able to self-evolve by internal structures modification.

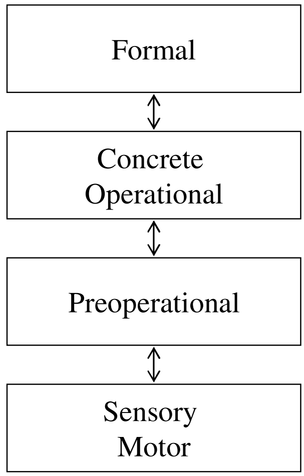

Fig. 7.1 Developmental stages hierarchy

There are four stages on the front face of the polytope. The notations are: S-Sensory Motor, K1-Preoperational, K2-Concrete Operational, and K3-Formal.

The development is considered clockwise.

Piaget considered that the sensorimotor stage differed from the latter stages in that the former was devoid of symbolic representation.

The central stage the Self may ensure the cooperation and redistribution of the four stages on another face of the polytope, with another starting stage.

Any stage embeds the previous ones. After one cycle an augmented reality may support a new cycle of development. The post-formal stage appears as a cognitive exemplar of the Self.

As shown in Fig. 7.2 two ways should be considered for development.

This means that after the integrative way S→K1→K2→K3 we need to look at the differentiation way K3′→K2′→K1′→S′.

Using the developments of the direct way may produce symmetry-breaking results for the reverse way. The swinging from direct to reverse developmental stages mediated by the Self may be a source of creativity in complex problem solving or science development.

That is because the boundaries where creative research stand out and new information is created consist of coexisting tendencies. Integration and differentiation coexists and the metastable coordination dynamics emerges as the delicate blend of integration and differentiation tendencies.

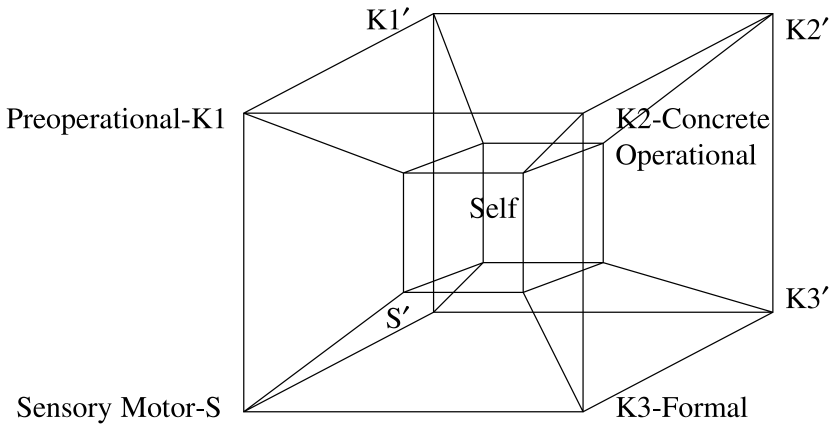

Fig. 7.2 Polytope for development stages

7.2 Logical Polytope

The Boolean logic operations may be illustrated by a polytope whose vertices represent the 16 traditional binary connectives that is, logical operations on two variables, of basic logic (Moretti 2009).

Table 7.1 shows the binary propositional connectives.

Table 7.1 Binary propositional connectives

⊤	∨	←	**P**	→	**q**	↔	∧	**NAND**	**XOR**	**¬q**	**N→**	**¬p**	**N←**	**NOR**	⊥
T	T	T	T	T	T	T	T	F	F	F	F	F	F	F	F
T	T	T	T	F	F	F	F	T	T	T	T	F	F	F	F
T	T	F	F	T	T	F	F	T	T	F	F	T	T	F	F
T	F	T	F	T	F	T	F	T	F	T	F	T	F	T	F

The binary-connective labels in Table 7.1 correspond to the digital labels shown in Fig. 7.3. Thus the binary-connective labels and the digital labels provide different ways of looking at the same abstract structure, which can itself be interpreted either as a Hasse diagram of a Boolean lattice or as a polytope. Table 7.1 shows the 16 connectives. We associate T to the digit “1” and F to the digit “0”.

Fig. 7.4 shows a different presentation of the logical polytope.

A projection of the 4-cube is retained.

Specific forms of the logical polytope have been applied to substantiate the steps of the drug discovery processes (Afshar et al. 2007, Luzeaux et al. 2008). The polytope describes in a general way a rational agent and enables the supervision of the computing process.

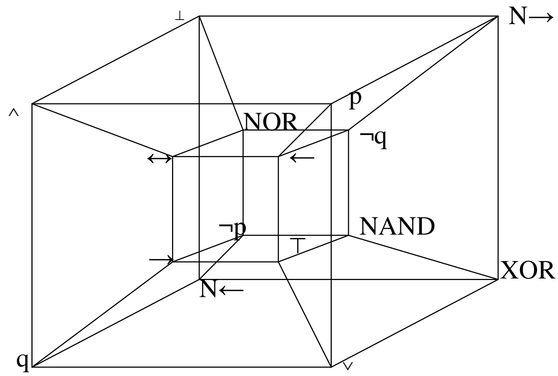

Fig. 7.3 Logical polytope

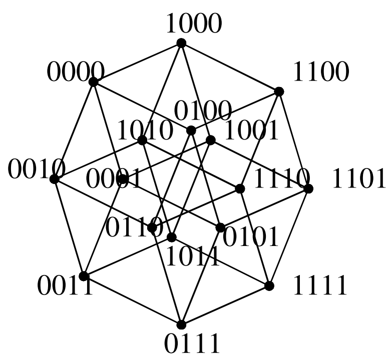

Fig. 7.4 Logical polytope sequence

7.3 Relational Complexity

A theory capable to analyze the processing demands of problems, to explain the main components of understanding and problem-solving methods was proposed by Halford (Halford 1993).

Structure mapping is the analogical reasoning that cognitive systems use to give meaning to problems by translating the given meaning of a problem into a representation or mental model that they already have and which allows them to understand the problem. The structure mappings that can be constructed depending upon the relational complexity of the structures they involve. The relational complexity of structures depends on the number of entities or the number of dimensions that are involved in the structure. The processing load of a task corresponds to the number of dimensions, which must be simultaneously represented, if their relations are to be understood. For example, to understand any comparison between two entities one must be able to represent two entities and one relation between them.

To understand a transitive relation, one must be able to represent at least three entities: otherwise it would not be possible to mentally arrange the entities in the right order that would reveal the relations between all entities involved.

Halford identified four levels of dimensionality for cognitive processes. The first is the level of element mappings. Mappings at this level are constructed on the basis of a single attribute. The second is the level of binary relations or relational mappings. At this level two-dimensional concepts can be constructed. Thus, two elements connected by a given relation can be considered at this level. The next is the level of system mappings, which requires that three elements or two relations must be considered simultaneously. At this level ternary relations or binary operations can be represented.

At the final level multiple-system mappings can be constructed. At this level quaternary relations or relations between binary operations can be constructed and four dimensions can be considered at once. The four levels of structure mappings correspond, in the theory of cognitive development of Piaget, to the sensorimotor, the preoperational, the concrete operational, and the formal stage. The four levels may be linked to the sensorimotor, interrelational, dimensional, and vectorial stages as described by Case (Case 1992).

In an overall sense there is a clear correspondence between Piaget's four major stages and the levels defined by Halford or by Case.

Fig. 7.5 shows the development stages-relational complexity polytope.

The elements of the front face of the polytope are presented in Table 7.2.

Table 7.2 outlines some categorification aspects for development stages.

Table 7.2 Categorification for development stages

Author\ Stage	K0	K1	K2	K3	Self
-	n=0	n=1	n=2	n=3	n≥4
Piaget (1971)	Sensori-motor	Preconceptual	Concrete Operational	Formal	Post-Formal
Halford (1993)	Elemental Association	Relational Mapping	Binary Operations	Quaternary Relations	-
Case (1992)	Sensori-motor	Interrelational	Dimensional	Vectorial	-

A challenge is the study of development stages for self-integrative closure, connecting levels n=0, sensory-motor and n=3, formal, and the emergence of the Self, corresponding to the levels n≥4 and to post-formal stages.

The notations for Fig. 7.5 are: K0-Elemental Association, K1-Relational Mapping, K2-Binary Operations, and K3-Quaternary Relation.

Fig. 7.5 outlines the direct integrative way S→K1→K2→K3 and the reverse way of differentiation K3′→K2′→K1′→S′.

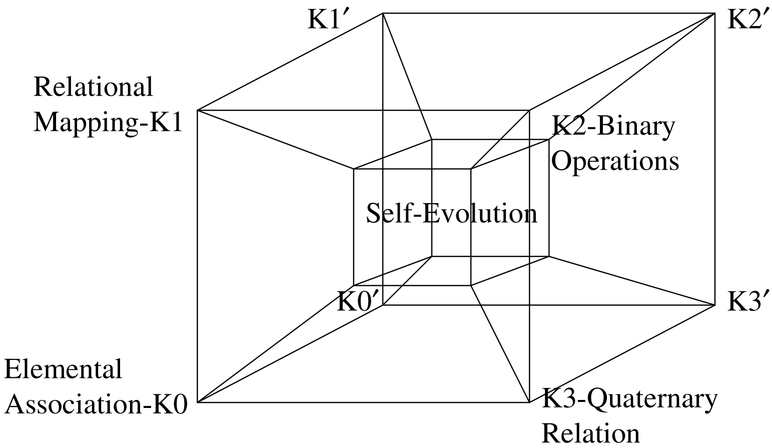

Fig. 7.5 Development stages: relational complexity polytope

A useful heuristic is that relational complexity cannot be reduced if the variables interact. This is analogous to analysis of variance method since interacting variables must be interpreted jointly. A procedure for determining effective relational complexity was described by Halford (Halford et al. 1998b). If a relation can be decomposed into simpler relations, then recomposed without loss of information, effective complexity is equivalent to the less complex relation.

The frontiers where new information is created consist of synchronized integrative and derivative ways. This explains why complex problem solving needs both integrative and derivative ways and the right rhythm of swinging between them.

The development stage theory of Piaget and the relational complexity theory open the problem of the level attained by different systems that learns and evolves.

The properties of higher cognitive processes and how they can be modeled by neural networks have been extensively studied by Halford and collaborators (Wilson and Halford 1994, Halford et al. 1998a, 1998b). They proposed and evaluated the so-called STAR (Structured Tensor Analogical Reasoning) model for problem solving.

The rank of tensor used in STAR is linked to the arity of relation, that is, to the number of attributes to the relation, and in the end, to the Piaget stages of cognitive development. The STAR model uses a tensor of rank-3 to represent a predicate of two arguments.

Halford studies suggest that for early Piaget stages in cognitive development, the categorical coproduct, "$\cup$", prevails allowing the associative knowledge. This is a fast and parallel process. During the higher Piaget stages the categorical product, "$\times$," seems preponderant, allowing the relational knowledge. It is a slow, sequential, effortful, higher cognitive process. The categorical product is naturally adapted to represent relations because its structure is analogous to the Cartesian product space in which relations are defined. The study of tensor product networks using distributed representations outlined the significant role of Hadamard matrices (Wilson and Halford 1994).

These matrices are special solutions of the wave equations.

The significance of Klein-4 group and of Latin squares for learning transfer in neural networks and in cognitive systems was also evaluated (Birney et al. 2006). Such structures are linked to the INRC group studied by Piaget (Inhelder and Piaget, 1958) as well as to standard solutions of the wave equation, WE model.

7.4 Explanatory Levels with n-Categories

Human inferential abilities like transitive inference and class inclusion, involve the dual category theory concepts, product and coproduct, respectively (Philips et al. 2009). Children around five years of age develop what is called transitive inference which is, for example, given that A is larger than B, and B is larger than C, one may infer that A is also larger than C. Class inclusion develops later in children and consists of the ability to discern between the cardinality of classes and subclasses.

Category theory shows that these abilities can be formally connected.

Transitive inference can be modeled with product, and class inclusion with its dual, the coproduct. This fact would explain that these two reasoning abilities have similar profiles of development, because they involve related sorts of processes, namely product and coproduct.

The n-category theory is useful to formally contrast category theory explanation against classical and connectionist approaches (Philips and Wilson 2010). Observe that the definitions of functor and natural transformation are very similar. In fact, they are morphisms at different levels of analysis. For n-category theory, a category such as Set is a 1-category, with 0-objects, that is, sets, for objects and 1-morphisms, that is, functions for arrows. A functor is morphism between categories. The category of categories, Cat, has categories for objects and functors for arrows. Thus, a functor is a 2-morphism between 1-objects, that is 1-categories, in a 2-category. A natural transformation is a morphism between functors. The functor category, Fun, has functors for objects and natural transformations for arrows. Thus, a natural transformation is a 3-morphism between 2-objects, that is functors, in a 3-category. A 0-category is just a discrete category, where the only arrows are identities, which are 0-morphisms. In this way, the order n of the category provides a formal notion of explanatory level (Phillips and Wilson 2010). Classical or connectionist compositionality is essentially a lower-level attempt to account for systematicity. That level is best described in terms of a 1-category. Indeed, a context-free grammar defined by a graph is modeled as the free category on that graph containing sets of terminal and non-terminal symbols for objects and productions for morphisms. By contrast, the category theory explanation involves higher levels of analysis, specifically functors and natural transformations, which live in 2-categories and 3-categories, respectively. Of course, one can also develop higher-order grammars that take as input or return as output other grammars. Similarly, one can develop higher-order networks that take as input or return as output other networks. The problem is that neither classical nor connectionist compositionality delineates those higher-order grammars or networks that have the systematicity property from those that do not.

Fig. 7.6 outlines the polytope for explanatory levels.

A decategorification way should be considered too. This means that after the integration way S→K1→K2→K3 we need to look at the differentiation way K3′→K2′→K1′→S′.

The differentiation is a kind of reverse epistemology. Observe that making use of the developments of the direct way, the reverse way may offer a symmetry-breaking results.

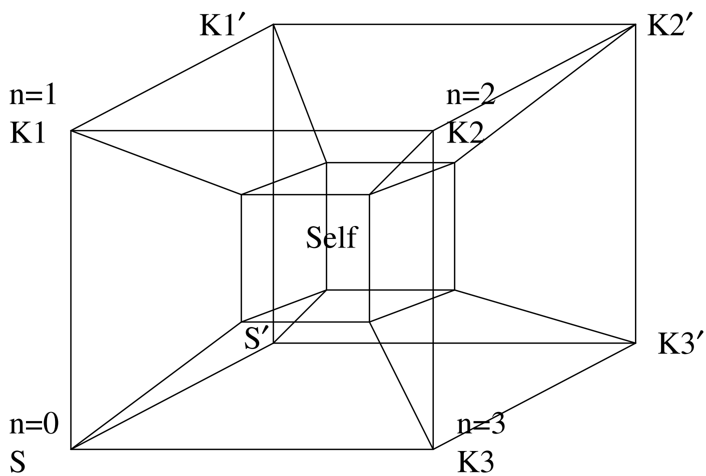

Fig. 7.6 Polytope for explanatory levels

In such cases, the swinging from direct to reverse epistemology is beneficial. Swinging methods based on direct and reverse epistemology have been applied in knowledge evaluation and development because the boundaries where new information is created consist of simultaneous tendencies. Tendencies to integrate should coexist with tendencies to differentiate and it is the intermixing of both that matters for self-evovability.

Table 7.3 outlines the categorification aspects for explanatory levels.

Table 7.3 Categorification for explanatory levels

Level	K0 (S)	K1	K2	K3	Self
-	n=0	n=1	n=2	n=3	n≥4
Categories	0-category	1-category	2-category	3-category	4-category
Example	sets	Set	Cat	Fun	-

The study of exploratory levels for self-integrative closure and the emergence of the Self corresponding to n≥4 are necessary.

7.5 LISA

LISA (Learning and Inference with Schemas and Analogies) is a system used in the synchronous activation approach to model analogical inference (Hummel and Holyoak 1997, Hummel and Choplin 2000). It demonstrates that temporal synchrony in conjunction with structured neural representations suffices to support complex forms of relational information processing specific to cognitive systems.

The problem for such systems is their suitability for reflexive or reflective cognitive processes. Reflexive processes are linked to categorical coproduct while reflective processes are linked to the categorical product. While reflexive and reflective processes follow different kinds of computational constraints, in most cases, the two types of processes interact and need to be integrated in the performance of a single task.

LISA is a computational model based on temporal synchrony and designed for analogical inference and for schemas induction.

LISA system is illustrated in Fig. 7.7. The basic level includes semantic units, s, the next includes the so-called localist units, L, (predicate/object or object/roles), the next level includes the sub-problems and the higher level the problems.

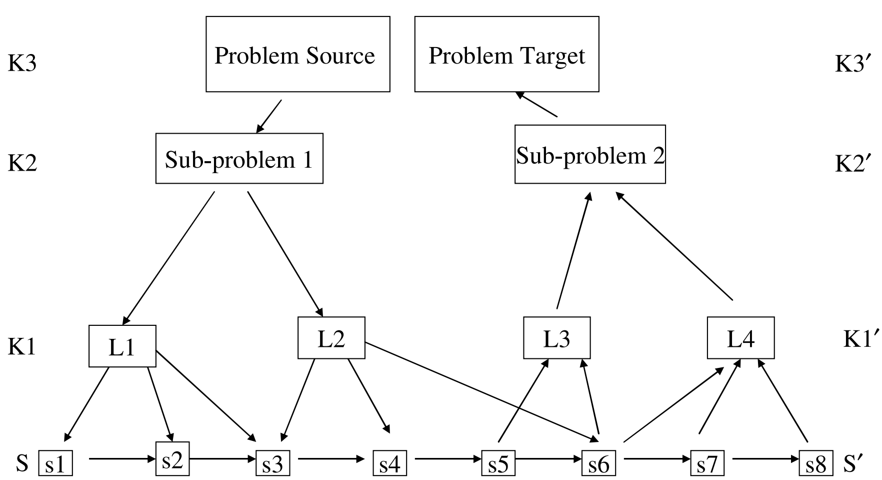

Fig. 7.7 LISA

LISA is a computational model based on temporal synchrony and designed for analogical inference and for schemas induction. The data for LISA network consists of a collection of trees and a representation that is a pattern of "0", "1" and so on for each terminal symbol occurring in those trees. The tree contains a hierarchy of entities: problem, sub-problems, roles, objects and semantics.

The task for the LISA network is to provide a means of compressing each tree into a representation, the so-called activation vector, and reconstructing the tree from its representation. The SKUP elements are naturally associated to the LISA elements. The problems to solve may be associated to the hierachy of conditions K1, K2 and K3. LISA contains a driver network associated to operators U, and to the reflective reasoning.

The representational structure of LISA provides at least a starting point for reflexive reasoning capabilities. LISA propositions are retrieved into memory via guided pattern matching. During retrieval and comparisons the proposition are divided into two mutually exclusive sets: a driver and one or more recipients or receivers. The receiver network is associated to possibilities P. The swinging between reflexive and reflective passes through the semantics. The LISA semantics elements are associated to the states S in SKUP.

The activation of semantic units is controlled by time. Often the analysts do not have the time to allow runaway activation of semantics since they needs make inferences quickly. Notice that in contrast to reflexive inferences which are fast, the reflective inferences may require more effort. An open problem is to establish, for imposed time frames, the number of swinging from reflexive to reflective and the order in which the swinging should be performed.

The Self takes into account the timescales for transition between levels. This allows the transition from problem source to problem target that is from integration way S→K1→K2→K3 and a differentiation way K3′→K2′→K1′→S′.

Inherently there appear differences between the two ways and this can be the source of creativity. That is because the boundaries where creative research grows require synchronized integration and differentiation tendencies.

Observe that this suppose that problem source and problem target are different.

Fig. 7.8 shows the polytope associated to LISA architecture.

The notations are: S-Semantic units, K1-Localist units, K2-Sub-problems, K3-Problems

Fig. 7.9 suggests a potential application of differential posets as cognitive architecture.

The D operator decomposes the problem while the U operator integrates and builds a problem target.

DORA (Discovery of Relations by Analogy) is a symbolic connectionist network that learns structured representations of relations from unstructured inputs. DORA is an extension of the LISA model of relational reasoning (Doumas et al. 2008).

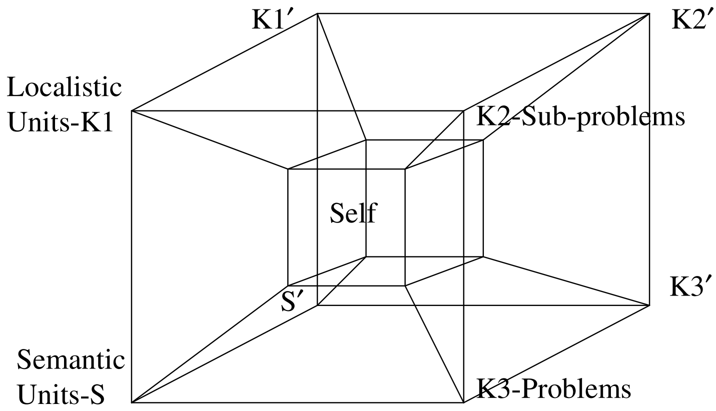

Fig. 7.8 Polytope for LISA framework

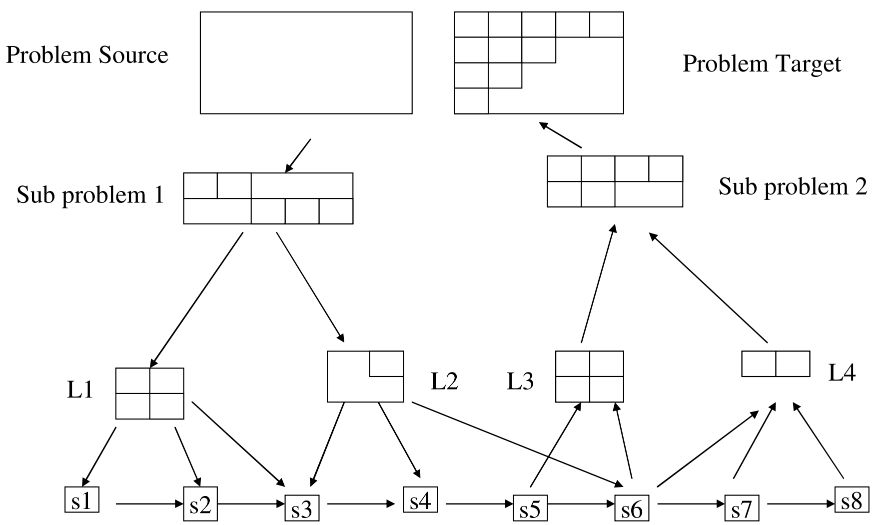

Fig. 7.9 Duality for LISA framework

DORA provides a means by which the representations used by LISA are learned from examples, and, consequently, provides an opportunity to understand the interplay between the dual sources of knowledge accumulation and increasing capacity limits as effectors of the changes in analogy making.

Like LISA, DORA dynamically binds distributed, that is connectionist, representations of relational roles and objects into explicitly relational, that is, symbolic, structures. The resulting representations enjoy the advantages of both connectionist and traditional symbolic approaches to knowledge representation, while suffering the limitations of neither. DORA's basic representational schema is adapted from LISA. In DORA, propositions are encoded by a hierarchy of structure unit.

Predicate and object units locally code for specific roles and fillers. While LISA must use different types of units to code for roles and their fillers, DORA uses the same types of units to code both roles and fillers and differentiates between roles and fillers via its binding mechanism. A comparison between DORA and STAR capabilities is due to Halford (Halford et al. 2010).

7.6 LIDA

LIDA (Learning Intelligent Distribution Agent) is a conceptual and computational framework for intelligent, autonomous, and conscious software agent that implements some ideas of the global workspace, GW, theory (Baars 2002).

LIDA appears as an attempt to adopt strategies observed in nature for creating information processing machinery.

The architecture is built upon the IDA (Intelligent Distribution Agent) framework, which was initially designed to automate the whole set of tasks of a human personnel agent who assigns resources to new tours of duty. LIDA employs a partly symbolic and partly connectionist memory organization, with all symbols being grounded in the physical world (Franklin 2006, Baars and Franklin 2009).

Baars' GW theory has inspired a variety of related consciousness models (Baars 1988). The central idea of GW theory is that conscious cognitive content is globally available for diverse cognitive processes including attention, evaluation, memory, and verbal report. The notion of global availability is suggested to explain the association of consciousness with integrative cognitive processes such as attention, decision making and action selection. Also, because global availability is necessarily limited to a single stream of content, GW theory may naturally account for the serial nature of conscious experience.

GW theory was originally described in terms of a blackboard architecture in which separate, quasi-independent processing modules interface with a centralized, globally available resource. This cognitive level of description is preserved in the computational models of Franklin, who proposed a model consisting of a population of interacting software agents, and Shanahan, whose model incorporates aspects of internal simulation supporting executive control and more recently spiking neurons (Shanahan 2006, 2008).

A central global workspace, GW, constituted by long-range cortico-cortical connections, assimilates other processes according to their salience. Other automatically activated processors do not enter the global workspace.

A neuronal implementation of a global workspace, GW, architecture, the so-called neuronal global workspace was studied (Dehaene et al. 2003).

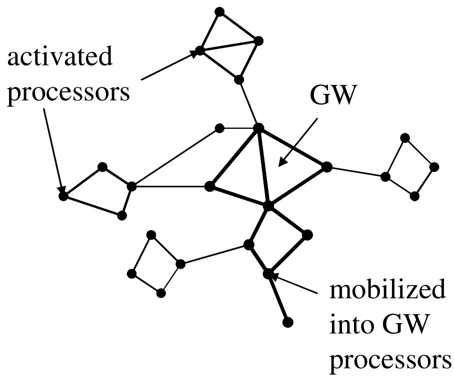

Fig. 7.10 Diagram for neuronal global workspace

Fig. 7.10 contains a schematic of the neuronal global workspace.

In this model, sensory stimuli mobilize excitatory neurons with long-range cortico-cortical axons, leading to the genesis of a global activity pattern among workspace neurons. Any such global pattern can inhibit alternative activity patterns among workspace neurons, thus preventing the conscious processing of alternative stimuli, for example, during the so-called attentional blink. The global neuronal workspace model predicts that conscious presence is a nonlinear function of stimulus salience; that is, a gradual increase in stimulus visibility should be accompanied by a sudden transition of the neuronal workspace into a corresponding activity pattern (Dehaene et al. 2003).

The complementary role of the conscious and unconscious for cognition and self-evolvability was emphasized.

The swinging between conscious and unconscious is an important tool for designing creative systems that can autonomously find solutions to highly complex and ill-defined construction problems.

When a module p1 invades the workspace, the others, as p2 are blocked at a similar depth.

Fig 7.11 illustrates the global workspace architecture activity.

In GW theory the processes, p1, p2 and so on, said to be unconscious, compete to enter the global workspace GW. This competition is at several levels.

Such processes are often thought of as memory activities, as for instance episodic or working memories.

Suppose that there are two levels of competition indexed by K1 and K2 and the competition is won by one process, for instance p2.

Having entered the GW, the winning process becomes the conscious state of the system. This is continuously broadcast back to the originating processes that change their state according to the conscious state. This results in a new conscious state and so on linking sensory input to memory and conscious states.

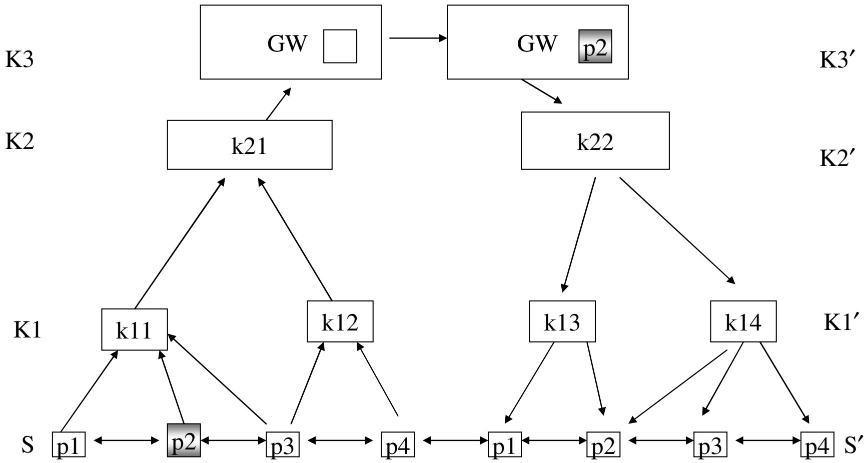

Fig. 7.11 Global workspace architecture

LIDA has distinct modules for perception, working memory, semantic memory, episodic memory, action selection, expectation and automatization (learning procedural tasks from experience), constraint satisfaction, deliberation, negotiation, problem solving, metacognition, and conscious-like behavior. Most operations are done by codelets implementing the unconscious processors, that is, specialized networks of the global workspace theory. A codelet is a small piece of code or program that performs one specialized, simple task. The LIDA framework incorporates three new modes of learning into the older IDA model: perceptual, episodic, and procedural learning, which are all of bottom-up type. Perceptual learning concerns learning of new objects, categories, relations, and so on, and takes two forms: strengthening or weakening of the base-level activation of nodes, as well as creation of new nodes and links in the perceptual memory. Episodic learning, on the other hand, involves learning to memorize specific events that is, the what, where, and when. It results from events taken from the content of *consciousness* being encoded in the transient episodic memory. Finally, procedural learning concerns learning of new actions and action sequences with which to accomplish new tasks. This combines selectionist learning that is, selecting from an obsolete repertoire, and the instructionalist learning, that is, constructing new representations, with functional consciousness providing reinforcements to actions. This architecture may explain many features of mind, however, it remains to be see whether high competence will be achieved in understanding language, vision, and common sense reasoning based on perceptions.

The LIDA model covers a large portion of human-like cognition (Franklin and Patterson 2006). Based primarily on GW theory the model implements a number of psychological and neuropsychological theories.

The LIDA computational architecture is derived from the LIDA cognitive model. The LIDA model and its ensuing architecture are grounded in the LIDA cognitive cycle. Every autonomous agent, human, animal, or artificial, must frequently sample and sense its environment and select an appropriate response, an action.

More sophisticated agents, such as humans, processes make sense of the input from such sampling in order to facilitate their decision making. The agent's life can be viewed as consisting of a continual sequence of these cognitive cycles. Each cycle constitutes a unit of sensing, attending and acting.

A cognitive cycle can be thought of as a moment of cognition, a cognitive moment.

During each cognitive cycle the LIDA agent first makes sense of its current situation as best as it can by updating its representation of its current situation, both external and internal. By a competitive process, as specified by GW theory, it then decides what portion of the represented situation is most in need of attention. Broadcasting this portion, the current contents of consciousness enable the agent to choose an appropriate action and execute it, completing the cycle.

Thus, the LIDA cognitive cycle can be subdivided into three phases, the understanding phase, the attention that is, the consciousness phase, and the action selection phase. Fig. 7.12 illustrates some elements of LIDA architecture. It starts in the lower-left corner and develops roughly clockwise (Snaider et al. 2011).

The first module is denoted by S. During the understanding phase, incoming stimuli activate low-level feature detectors in Sensory Memory. The output is sent to Perceptual Associative Memory, where higher-level feature detectors feed in to more abstract entities such as objects, categories, actions, events, and so on. The resulting percept moves to the Workspace, denoted by K1. Here it triggers both Transient Episodic Memory, and Declarative Memory, producing local associations. These local associations are combined with the percept to generate a Current Situational Model, which represents the agent's understanding of what is going on right now.

Attention Codelets, associated here by K2, begins the attention phase by forming coalitions of selected portions of the Current Situational Model and moving them to the GW.

A competition in the GW then selects the most salient, the most relevant, the most important, and the most urgent coalition whose contents become the content of consciousness. These conscious contents are then broadcast globally, initiating the action selection phase, associated here to K3.

The GW space corresponds to the Self. The neuronal global workspace, GW appears in Fig. 7.12 as a working example of the Self.

The action selection phase of LIDA's cognitive cycle is also a learning phase in which several processes operate in parallel.

New entities and associations, and the reinforcement of old ones, occur as the conscious broadcast reaches Perceptual Associative Memory. Events from the conscious broadcast are encoded as new memories in Transient Episodic Memory.

Possible action schemas, together with their contexts and expected results, are learned into Procedural Memory from the conscious broadcast. Older schemas are reinforced.

In parallel with all this learning, and using the conscious contents, possible action schemas are recruited from Procedural Memory. A copy of each such schema is instantiated with its variables bound and sent to Action Selection, where it competes to be the behavior selected for this cognitive cycle. The selected behavior triggers Sensory-Motor Memory to produce a suitable algorithm for the execution of the behavior.

Its execution completes the cognitive cycle.

The Workspace requires further explanation. Its internal structure is composed of various input buffers and three main modules: the Current Situational Model, the Scratchpad and the Conscious Contents Queue. The Current Situational Model is where the structures representing the actual current internal and external events are stored. Structure-building codelets are responsible for the creation of these structures using elements from the various submodules of the Workspace. The Scratchpad is an auxiliary space in the Workspace where structure-building codelets can construct possible structures prior to moving them to the Current Situational Model. The Conscious Contents Queue holds the contents of the last several broadcasts and permits LIDA to understand and manipulate time-related concepts.

The GW mediates between the direct integrative way S→K1→K2→K3 and the reverse differentiation way K3′→K2′→K1′→S′ as shown by Fig. 7.11 and fig. 712.

The reverse epistemology allows making use of the developments of the direct way and will offer is a kind of symmetry-breaking result. The swinging from direct to reverse epistemology is beneficial.

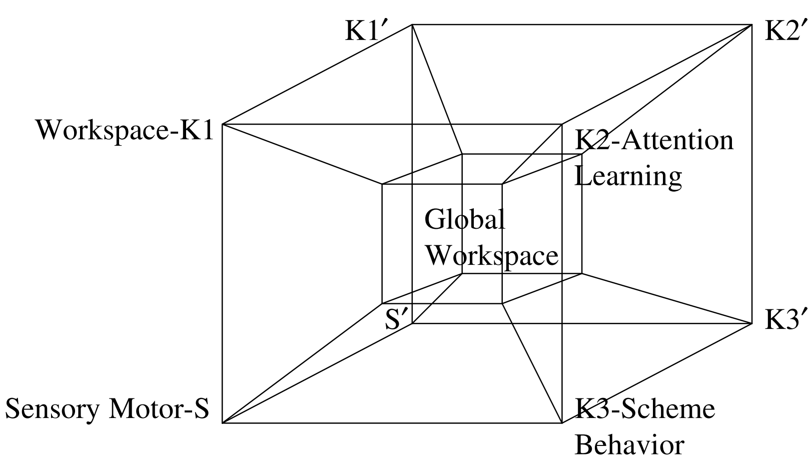

Fig. 7.12 Polytope for LIDA framework

The wave character manifested as swinging behavior is applied for evaluation and creative behavior. The boundaries where creative research grows and new information is created consist of synchronized tendencies. Tendencies to integrate should coexist with tendencies to differentiate and it is the blend of both that counts for self-evolvability.

Fig. 7.13 suggests a potential application of differential posets as cognitive architecture.

The U operator transfer processes as p2 to the GW space while the D operator transfer processes from GW toward field.

One feature of human thought not accounted for by the GW theory is the reflexivity.

This is the capacity for a conscious thought to refer to itself or to other conscious states.

Consider that thought is internally in simulation with the environment. This simulation hypothesis can explain our experience of an inner world.

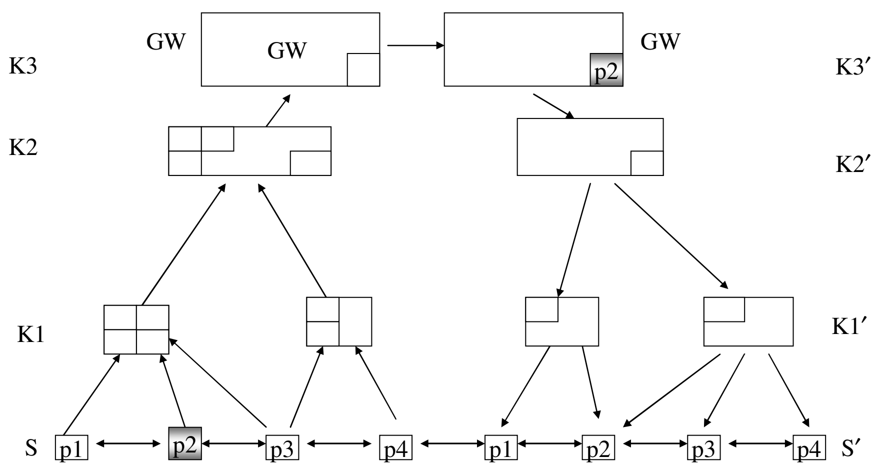

Fig. 7.13 Duality for LIDA framework

The simulation hypothesis is based on the following assertions:

• The brain’s motor centers can be active without producing explicit action

• The brain’s perceptual apparatus can be active without the presence of external stimuli

• Internally generated motor activity can elicit internally generated perceptual activity through associative mechanisms

By augmenting the basic GW workspace architecture with an internally closed loop it is possible to reconcile the GW theory with the so-called simulation hypothesis (Shanahan 2006). The proposal is in support of the hypothesis that organisms whose brains are endowed with such an internal loop are capable of rehearsing the consequences of potential actions prior to actually carrying them out.

Such implementations are useful as a proof-of-concept, despite the present lack of neurological plausibility, both at the level of the neuron model used and in its employment of a single attractor network to model the global workspace.

Finally it should be observed that LISA, LIDA and global workspace GW theory are similar approaches. They mix serial and parallel computations, corresponding to different types of categorical product.

This supports their study by similar polytopic architectures.

7.7 *Physarum* Computing Systems

The slime mold *Physarum polycephalum* is a multinuclear, single-celled organism that has properties making it ideal for the study of resource distribution networks and of cognitive capabilities (Nakagaki et al. 2000, Nakagaki 2001).

The organism is a single cell, but it can grow to tens of centimeters in size so that it can be studied and manipulated with modest laboratory facilities.

The presence of nutrients in the cell body triggers a sequence of chemical reactions leading to oscillations along the cell body. Tubes self-assemble perpendicular to the oscillatory waves to create networks linking nutrient sources throughout the cell body. There are two key mechanisms in the slime mold life cycle that transfer readily to resource distribution network problems. First, during the growth cycle, the slime mold explores its immediate surroundings with pseudopodia via chemotaxis to discover new food sources. The second key mechanism is the temporal evolution of existing routes through nonlinear feedback to efficiently distribute nutrients throughout the organism. In slime mold, it can be shown experimentally that the diameters of tubes carrying large fluxes of nutrients grow to expand their capacity, and tubes that are not used decline and can disappear entirely. Unlike any other circulatory system, networks in slime mold rebuild themselves dynamically to changing environmental conditions.

Nakagaki proposed a simple yet powerful model for tube evolution in *Physarum* to reproduce slime mold maze-solving experiments (Nakagaki 2001). The model captures the evolution tube capacities in an existing network through a coupled system of ordinary differential equations. Flow through the network is driven by a pressure at each node. The diameter of the tubes evolves based on the flux of nutrients through the network.

Nakagaki's group makes considerable claims about robustness and intelligence level in the *Physarum* colonies (Nakagaki et al. 2000, 2001, 2004).

Implementation of a general-purpose computing machine is the most remarkable feature of the plasmodium of *Physarum*.

The cognitive levels of *Physarum* may be compared to these attained by some pointer machines (Ben-Amram 1995, 1998).

Experimentally it was demonstrated that the plasmodium can implement the Kolmogorov–Uspensky Machine (KUM), a mathematical machine in which the storage structure is an irregular graph (Adamatzky 2007). The KUM is a forerunner and direct ancestor of Schoenhage's storage modification machines (Schoenhage 1980).The storage modification machines are basic architectures for random access machines, which represent the basic architecture of modern-day computers. The plasmodium-based implementation of KUM provides a biological prototype of a general-purpose computer.

The key component of the KUM is an active zone, which may be seen as a computational equivalent to the head in a Turing machine. Physical control of the active zone is of utmost importance because it determines functionality of the biological storage modification machine.

Laboratory and computer experiments with *Physarum* show basic operations Add node, Add edge, Remove Edge implemented in the *Physarum* machine. They also provide results on controlling movement of an active zone.

The filaments movements for *Physarum* suggest that their capabilities are at the level of the 1-categories and 2-categories. For 2-categories the pentagon relation is valid.

Physarum is able to disconnect a filament and reconnect it in another position.

This corresponds to 2-categories. At operadic level this corresponds to the associahedron K_4.

For 3-categories the so-called pentagon of pentagons or the associahedron K_5 should be considered. This needs a spatial awareness that allows evaluating the *Physarum* computing capabilities.

Table 7.4 shows the knowledge level associated to different associahedra

It refers to categorification aspects.

Needed are the study of exploratory levels for self-integrative closure and the emergence of the Self corresponding to n≥4.

An interesting test for *Physarum* capabilities would be the evolution in a high-dimensional space with restrictions as shown in Fig. 7.14.

Table 7.4 Categorification for associahedra

Level	K0	K1	K2	K3	Self
	n=0	n=1	n=2	N=3	n≥4
Categories	0-category	1-category	2-category	3-category	4-category
Associahedra	K(2)	K(3)	K(4)	K(5)	K(6)
Geometry	-	Trees	Pentagon	Pentagon of Pentagons	-

Figure 7.14 illustrates the node-disjoint path construction between the source *x* = 0000 and the destination *y* = 1110 in a 4-cube.

The edges on the four node-disjoint paths are labeled with the corresponding dimensions. Since *x* and *y* differ in bits 0, 1, and 2, the four paths correspond to the dimension sequences (0,1,2), (1,2,0), (2,0,1), and (3,0,1,2,3). At least one of these paths is fault-free in the presence of any three or less faulty nodes.

Tsuda described an experimental setup that interfaces an amoeboid plasmodium of *Physarum* with an omni-directional hexapod robot to realize an interaction loop between environment and plasticity in control (Tsuda et al. 2006). Through this bio-electronic hybrid architecture the continuous negotiation process between local intracellular reconfiguration on the micro-physical scale and global behavior of the cell in a macroscale environment can be studied in a device setting.

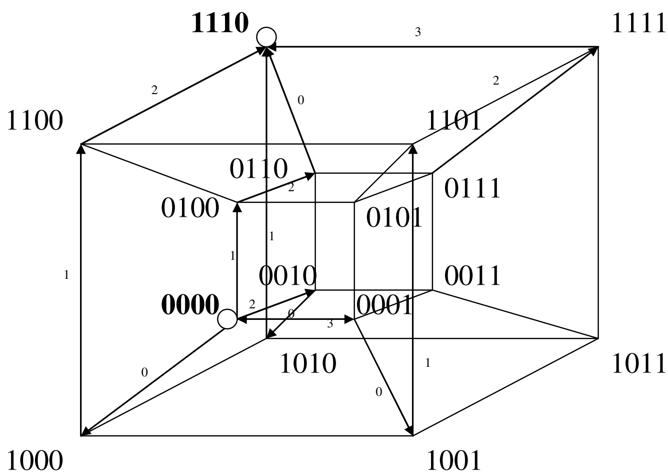

Fig. 7.14 Disjoint paths in 4-cube

The filaments movements for *Physarum* suggest that their capabilities are at the level of 1-categories and 2-categories. For 2-categories the pentagon relation is valid. This means cognitive capabilities

For 3-categories the pentagon of pentagons or a kind of spatial sensitivity should be considered.

References

Adamatzky, A.: Physarum machine: implementation of a Kolmogorov-Uspensky machine on a biological substrate. Parallel Processing Letters 17, 455–467 (2007)

Afshar, M., Dartnell, C., Luzeaux, D., Sallantin, J., Tognetti, Y.: Aristotle's Square Revisited to Frame Discovery Science. Journal of Computers 2(5), 54–66 (2007)

Baars, B.J.: A cognitive theory of consciousness. Cambridge University Press, Cambridge (1988)

Baars, B.J.: The conscious access hypothesis: origins and recent evidence. Trends in Cognitive Science 6, 47–52 (2002)

Baars, B.J., Franklin, S.: Consciousness is computational: The LIDA model of Global Workspace Theory. International Journal of Machine Consciousness 1(1), 23–32 (2009)

Ben-Amram, A.M.: What is a Pointer machine? SIGACT News, ACM Special Interest Group on Automata and Computability Theory 26 (1995)

Ben-Amran, A.M.: Pointer machines and pointer algorithms: an annotated bibliography (1998), http://www2.mta.ac.il/amirben

Birney, D.P., Halford, G.S., Andrews, G.: Measuring the influence of complexity on relational reasoning: The development of the Latin Square Task. Educational and Psychological Measurement 66, 146–171 (2006)

Bringsjord, S., Taylor, J., Wojtowicz, R., Arkoudas, K., van Heuvlen, B.: Piagetian Roboethics via Category Theory: Moving Beyond Mere Formal Operations to Engineer Robots Whose Decisions are Guaranteed to be Ethically Correct. In: Anderson, M., Anderson, S. (eds.) Machine Ethics. Cambridge University Press, Cambridge (2010)

Case, R.: The mind's staircase: Exploring the conceptual underpinnings of children's thought and knowledge. Erlbaum, Hillsdale (1992)

Dehaene, S., Sergent, C., Changeux, J.P.: A neuronal network model linking subjective reports and objective physiological data during conscious perception. Proc. Natl. Acad. Sci. USA 100, 8520–8525 (2003)

Doumas, L.A.A., Hummel, J.E., Sandhofer, C.M.: A theory of the discovery and predication of relational concepts. Psychological Review 115, 1–43 (2008)

Franklin, S.: The LIDA architecture: Adding new modes of learning to an intelligent, autonomous, software agent. In: Proc. of the Int. Conf. on Integrated Design and Process Technology. Society for Design and Process Science, San Diego (2006)

Franklin, S., Patterson, F.G.J.: The LIDA Architecture: Adding New Modes of Learning to an Intelligent, Autonomous, Software Agent. In: IDPT 2006 Proceedings Integrated Design and Process Technology. Society for Design and Process Science (2006)

Halford, G.S.: Children's understanding: The development of mental models. Erlbaum, Hillsdale (1993)

Halford, G.S., Wilson, W.H., Phillips, S.: Processing capacity defined by relational complexity. Implications for comparative, developmental and cognitive psychology. Behavioural and Brain Sciences 21(6), 803–831 (1998a)

Halford, G.S., Wilson, W.H., Phillips, W.: Relational complexity metric is effective when assessments are based on actual cognitive processes. Behavioral Brain Sciences 21(6), 803–864 (1998b)

Halford, G.S., Wilson, W.H., Phillips, S.: Relational knowledge: the foundation of higher cognition. Trends in Cognitive Science 14(11), 497–505 (2010)

Hummel, J.E., Choplin, J.M.: Toward an integrated account of reflexive and reflective reasoning. In: Gleitman, L.R., Joshi, A.K. (eds.) Proceedings of the Twenty Second Annual Conference of the Cognitive Science Society, pp. 232–237. LEA, Mahwah (2000)

Hummel, J.E., Holyoak, K.J.: Distributed representation of structure. A theory of analogical access and mapping. Psychological Review 104, 427–466 (1997)

Inhelder, B., Piaget, J.: The Growth of Logical Thinking from Childhood to Adolescence. Basic Books, New York (1958)

Luzeaux, D., Sallantin, J., Dartnell, C.: Logical extensions of Aristotle's square. Logica Universalis 2, 167–187 (2008)

Moretti, A.: The Geometry of Logical Opposition. PhD. Thesis. University of Neuchâtel, Switzerland (2009)

Nakagaki, T.: Smart behavior of true slime mold in a labyrinth. Research in Microbiology 152(9), 767–770 (2001)

Nakagaki, T., Yamada, H., Hara, M.: Smart network solutions in an amoeboid organism. Biophysical Chemistry 107(1), 1–5 (2004)
Nakagaki, T., Yamada, H., Toth, A.: Maze-solving by an amoeboid organizm. Nature 407(6803), 470–470 (2000)
Nakagaki, T., Yamada, H., Toth, A.: Path finding by tube morphogenesis in an amoeboid organism. Biophysical Chemistry 92(1-2), 47–52 (2001)
Philips, S., Wilson, W.H., Hadford, G.: What do transitive inference and class inclusion have in common? categorical (co)products and cognitive development. PLoS Comput. Biol. 5(12), e1000599 (2009)
Philips, S., Wilson, W.H.: Categorial compositionality: A category theory explanation for the systematicity of human cognition. PLoS Comput. Biol. 6(7), 1–14 (2010)
Piaget, J.: Genetic Epistemology. Columbia University Press, New York (1970)
Piaget, J.: The construction of Reality in the Child. Ballantine Books, New York (1971)
Piaget, J.: Introduction a l'epistemologie genetique. La pensee Mathematique. Presses Universitaires de France, Paris (1973)
Schoenhage, A.: Storage modification machines. SIAM J. Comp. 9, 490–508 (1980)
Shanahan, M.: A cognitive architecture that combines internal simulation with a global workspace. Consciousness and Cognition 15, 433–449 (2006)
Shanahan, M.: A spiking neuron model of cortical broadcast and competition. Conscious Cogn. 17(1), 288–303 (2008)
Snaider, J., McCall, R., Franklin, S.: The LIDA Framework as a General Tool for AGI. In: The Fourth Conference on Artificial General Intelligence, Mountain View, CA (2011)
Tsuda, S., Zauner, K.-P., Gunji, Y.-P.: Robot Control: From Silicon Circuitry to Cells. In: Ijspeert, A.J., Masuzawa, T., Kusumoto, S. (eds.) BioADIT 2006. LNCS, vol. 3853, pp. 20–32. Springer, Heidelberg (2006)
Wilson, H.W., Halford, G.S.: Robustness of Tensor Product networks using distributed representations. In: Proceedings of the Fifth Australian Conference on Neural Networks (ACNN 1994), pp. 258–261 (1994)
Yang, Y., Bringsjord, S.: Mental Metalogic: A New, Unifying Theory of Human and Machine Reasoning. Erlbaum, Mahwah (2005)