Self-Awareness and Self-Control in NARS

Pei Wang[1], Xiang Li[1], and Patrick Hammer[2]

[1] Department of Computer and Information Sciences, Temple University, USA
`{pei.wang,xiang.li003}@temple.edu`
[2] Institute for Software Technology, Graz University of Technology, Austria
`patrickhammer9@hotmail.com`

Abstract. This paper describes the self-awareness and self-control mechanisms of a general-purpose intelligent system, NARS. The system perceives its internal environment basically in the same way as how it perceives its external environment, though the sensors involved are completely different. NARS uses a “self” concept to organize its relevant beliefs, tasks, and operations. The concept has an innate core, though its content and structure are mostly acquired gradually from the system’s experience. The “self” concept and its ingredients play important roles in the control of the system.

Functions like “self-awareness”, “self-control”, and “self-consciousness” are closely related to advanced forms of intelligence. The difficulty of realizing these functions in a machine is both technical and theoretical, as there is no widely accepted theory about them, and even their definitions are highly controversial. This paper is not an attempt to address all relevant issues. Instead, we will present the relevant aspects of NARS (Non-Axiomatic Reasoning System), a formal model of general intelligence, which has been mostly implemented and is under testing and tuning.

In the following, the conceptual design of NARS is briefly introduced first, then the parts mostly relevant to “self” are described in more detail. Finally, the major design decisions are compared with the related works.

1 NARS Introduction

NARS is designed according to the hypothesis that “intelligence” is *the ability for a system to adapt to its environment and to work with insufficient knowledge and resources*. Under the length restriction, in this paper the system is only introduced very briefly. For details of the system, see the related papers[3] and books [14, 17].

NARS is a reasoning system, with a formal language, Narsese, for knowledge representation, and a formal logic, NAL (Non-Axiomatic Logic), for inference using Narsese sentences as premises and conclusions. NAL belongs to the “term logic” tradition where the smallest component of the language is a *term*, and “subject-copula-predicate” is the simplest format of *statement*. “$S \rightarrow P$” is a basic form of statements, and is called *inheritance statement*, where S is the subject term, P the predicate term, and “$\rightarrow$” the inheritance copula. The intuitive meaning of “$S \rightarrow P$” is “S is a special case of P” and

---

[3] Mostly accessible at `https://cis.temple.edu/~pwang/papers.html`

"P is a general case of S". For example, "$robin \rightarrow bird$" corresponds to "Robin is a type of bird".

In its simplest form, a term is just a string of symbols from an arbitrary alphabet. Starting from these "atomic" terms, *compound terms* can be composed recursively, each with a connector and a list of component terms. Different term connectors represent different relations among the components, as shown by the following examples:

- **Sets:** Term $\{Tom, Jerry\}$ is an *extensional set* specified by enumerating its instances; term $[small, yellow]$ is an *intensional set* specified by enumerating its properties.
- **Intersections and differences:** Term $(bird \cap swimmer)$ represents "birds that can swim"; term $(bird - swimmer)$ represents "birds that cannot swim".
- **Products and images:** The relation "John is the uncle of Zack" is represented as "$(\{John\} \times \{Zack\}) \rightarrow uncle\text{-}of$", "$\{John\} \rightarrow (uncle\text{-}of \,/\, \diamond\ \{Zack\})$", and "$\{Zack\} \rightarrow (uncle\text{-}of \,/\, \{John\} \diamond)$", equivalently.
- **Statement:** "John knows soccer balls are round" can be represented as a *higher-order statement* "$\{John\} \rightarrow (know \,/\, \diamond\ \{soccer\text{-}ball \rightarrow [round]\})$", where the statement "$soccer\text{-}ball \rightarrow [round]$" is used as a term.

Beside the *inheritance* copula ('$\rightarrow$', "is a type of"), NAL also has three other copulas: *similarity* ('$\leftrightarrow$', "is similar to"), *implication* ('$\Rightarrow$', "if-then"), and *equivalence* ('$\Leftrightarrow$', "if-and-only-if"), and the last two are used between two statements.

A statement is a compound term with a truth-value. It can be formed using two terms and a copula, as well as using statement connectors *negation* ('$\neg$'), *conjunction* ('$\wedge$'), and *disjunction* ('$\vee$'), which are defined similarly (but not utilizing Boolean functions) to those in propositional logic [14]. There are several special types of statements needed for NARS to reason on procedural knowledge as in logic programming [7]:

Event: a statement with a time-dependent truth-value. Two events may happen sequentially or concurrently. Compound events can describe a sequence of events or parts of a complex event. By comparing the occurrence time of an event with the current time, the event gets a tense like "past", "present", or "future".

Operation: an event directly realizable by the system itself via executing the associated code or command. Formally, an operation is an application of an operator on a list of arguments, written as $op(a_1, \ldots, a_n)$. Intuitively, it is a procedure call, where the argument list includes both input and output arguments.

Goal: an event the system wants to realize. It is a statement with an associated "desire-value", indicating the extent to which the system desires a situation where the statement is true.

Since NARS is designed under the Assumption of Insufficient Knowledge and Resources (AIKR for short), the truth-value of a statement measures the extent of evidential support, not the agreement with a corresponding fact. In NAL, a truth-value is a pair of real numbers in [0,1] × (0,1), where the first number, *frequency*, measures the proportion of positive evidence of the statement among all available evidence, while the second number, *confidence*, measures the proportion of currently available evidence among the total amount of available evidence at a moment in the future, after new evidence of a constant amount is collected.

Defined in this way, *truth* in NARS is "experience-grounded". Similarly, the *meaning* of a term is determined by how it is related to other terms in the system's experience. As the experience of a system grows over time, the truth-value of statements and the meaning of terms in the system change accordingly. This *experience-grounded semantics* (EGS) is fundamentally different from the traditional *model-theoretic semantics*, since it defines *truth* and *meaning* according to a (dynamic and system-specific) experience, rather than a (static and system-independent) model. In the simplest implementation of NARS, its *experience* is a stream of Narsese sentences, which will be summarized to become the knowledge of the system.

NAL uses (formal) inference rules to derive new knowledge from existing knowledge. Since every piece of knowledge, also known as *belief*, is true to a degree, each inference rule has a truth-value function that calculates the truth-value of the conclusion according to the evidence provided by the premises.

As a term logic, typical inference rules in NAL are *syllogistic*, and takes two premises (with one common term) to derive a conclusion (between the other two terms). The NAL rules of this type include *deduction*, *induction*, and *abduction*, as specified by Peirce [10], though the truth-value of every statement is extended from $\{0, 1\}$ to $[0,1]\times(0,1)$ [14]. Among the three, *deduction* is a rule that carries out *strong inference*, as its conclusions can approach the maximum confidence 1 for affirmative premises of high confidence, while the other two carry out *weak inference*, where the confidence of the conclusions has a constant upper bound less than 1 for all premises.

Under AIKR, NARS may have inconsistent beliefs, that is, the same statement may obtain different truth-values according to different evidential bases. When the system locates such an inconsistency, it either uses the *revision* rule to produce a more confident conclusion by pooling the evidence (if the evidence bases are disjoint), or use the *choice* rule to pick the belief with higher confidence (if the evidence bases are not disjoint).

NAL also has *compositional rules* that compose or decompose compound terms according to the definition of their connector, so as to summarize the system's experience more efficiently.

The inference rules of NAL can be used in both *forward inference* (from existing beliefs to derived beliefs) and *backward inference* (from existing beliefs and questions/goals to derived questions/goals).

Equipped with these inference rules, NARS can carry out the following types of inference tasks:

- to absorb new experience into the system's beliefs, as well as to spontaneously derive some of their implications;
- to achieve the input goals (and the derived goals) by selectively executing the available operations according to the system's beliefs;
- to answer the input questions (and the derived questions) according to the system's beliefs.

Under AIKR, new tasks can enter the system at any time, each with its own time requirement, and its content can be any Narsese statement. Working in such a situation, usually NARS cannot perfectly accomplish all tasks in time, but has to allocate its limited time and space resources among them, and has to dynamically adjust the allocation according to the change of context and the feedback to its actions.

In the memory of NARS, beliefs and tasks are organized into *concepts*, according to the terms appearing in them. Therefore, for a term T, concept C_T refers to all beliefs and tasks containing T. For example, the beliefs on "$robin \rightarrow bird$" are referred to within concepts C_{robin} and C_{bird}, as well as other relevant concepts. A "concept" in NARS is a unit of both storage and processing, and models the concepts found in human thinking [14].

To indicate the relative importance of concepts, tasks, and beliefs to the system, priority distributions are maintained among them. The priority of an item (concept, task, or belief) summarizes the attributes to be considered in resource allocation, including its intrinsic quality, usefulness in history, relevance to the current context, etc. Therefore items with higher priority values will get more resources.

NARS runs by repeating an inference cycle consisting of the following major steps:

1. Select a concept within the memory.
2. Select a task referred by the concept.
3. Select a belief referred by the concept.
4. Derive new tasks from the selected task and belief by the applicable inference rules.
5. Adjust the priority of the selected belief, task, and concept according to the context and feedback.
6. Selectively put the new tasks into the corresponding concepts, and report some of them to the user.

All selections in the above steps are probabilistic, biased by priority, that is, the probability for an item to be selected is positively correlated to its priority value. Consequently, the tasks will be processed in a time-sharing manner, with different speeds. For a specific task, its processing does not follow a predetermined algorithm, but is the result of many inference steps, whose combination is formed at runtime, so is neither predictable nor repeatable accurately, because both the external environment and the internal state of the system change in a non-circular manner.

2 "Self" in NARS

In this paper we focus on the aspects of NARS that are directly relevant to self-awareness and self-control. Therefore we will not fully discuss the following topics often involved in the related discussions:

- "Higher-order statement" in NARS covers "statement about statement", "knowledge about operations", etc., which are often taken as functions of "metacognition" [5]. Since such knowledge is typically about individual statements or operations, not about the system as a whole, it is not discussed here. For how this kind of knowledge is processed in NARS, see [14, 17].
- NARS constantly compares the certainty of beliefs, and dynamically allocates its resources among competing tasks. Even though the relevant mechanisms are indeed at the meta-level with respect to beliefs and tasks, they are implicitly embedded in the code, so not generally accessible to the system's deliberation, nor can they be modified by the system itself, therefore they are also not discussed here.

- NARS has mechanisms for feeling and emotion, which are important parts of self-awareness and self-control. However, since they have been discussed in detail in our recent publication [18], they will only be mentioned briefly in this paper.

NARS' beliefs about itself start at its built-in operations. Operation $op(a_1, \ldots, a_n)$ corresponds to a relation the system can establish between itself and the arguments, so it is equivalent to statement "$(\times, \{SELF\}, \{a_1\}, \ldots, \{a_n\}) \to op$" (where the subject term is a *product* term written in the prefix format), since it specifies a relation among the arguments plus the system identified by the special term $SELF$.

Similar to the case of logic programming [7], here the idea is to uniformly represent declarative knowledge and procedural knowledge. So in NARS knowledge about the system itself is unified with knowledge about others. For instance, the operation "open this door" is represented as "$(\times, \{SELF\}, \{door\_1\}) \to open$" [4], while "John opened this door" as "$(\times, \{John\}, \{door\_1\}) \to open$" (tense omitted to simplify the discussion). In this way, imitation can be carried out by analogical inference.

As mentioned previously, in NARS the meaning of a concept is gradually acquired from the system's experience. However, this "experience-grounded semantics" (EGS) does not exclude the existence of innate concepts, beliefs, and tasks. In the above example, '$SELF$' is such a concept, with built-in operations that can be directly executed from the very beginning. Such operations depend on the hardware/software of the host system, so are not specified as parts of NARS, except that they must obey the format requirements of Narsese. According to EGS, in the initial state of NARS, the meaning of a built-in operation is procedurally expressed in the corresponding routine, while the meaning of '$SELF$' consists of these operations. To the system, "I am whatever I can do" or "I am whatever I can do and feel" are possible ways to express this situation, since in NARS sensation and perception are also operations.

As the system begins to have experience, the meaning of every concept will be more or less adjusted as it is experienced, directly or indirectly. For a built-in operation, the system will gradually learn its preconditions and consequences, so as to associate itself with the goals it can achieve. It is like when we know how to raise our hand first, and then know it as a way to get the teacher's attention. The '$SELF$' concept will be enriched in this way, as well as through its relations with other concepts representing objects and other systems in the outside environment. Therefore, self starts from "what I can do" to include "what I am composed of", "how I look like", "what my position is in the society", etc. "Self" does not have a constant meaning determined by a denotation or definition. Instead, the system gradually learns who it is, and its self-image does not necessarily converge to a "true self".

An operation may be completely executed by the actuator of the host system (e.g., A NARS-based robot raises a hand or moves forward), or partly by another coupled system or device (e.g., A NARS-based robot pushes a button or issues a command to another robot). NARS has an interface for such "external" operations to be registered.

NARS is designed to allow all kinds of operations to be used in a "plug-and-play" manner, i.e., to be connected to the system at run time by a user or the system itself. A learning phase is usually needed for an operation to be used properly and effectively.

---

[4] Here, the inheritance copula encodes that the relation between $\{SELF\}$ and $\{door\_1\}$, is a special case of opening.

In principle, no operation is necessarily demanded in every NARS implementation, except a special type of "mental" operations that work on the system's own "mind". There are several groups of mental operations, including:

Task generation. An inference task in NARS can either be input or derived recursively from an input task. The derivation process does not change the type of the task (new/activated belief, goal, or question). However, in certain situations a task needs to be generated from another one of a different type. For example, a new belief ("It is cold.") may trigger a new goal ("Close the window!"). This relation is represented as an implication statement where the consequent is not a statement, but an operation call, similar as in a production rule.

Evidence disqualification. By default, the amount of supporting evidence for every belief accumulates over time. Therefore, though the frequency value of the belief may either increase or decrease (depending on whether the new evidence is positive or negative), its confidence value increases monotonically. This treatment is supplemented by a mental operation that allows the system to doubt a belief by decreasing its confidence value to a certain extent.

Concept activation. The resource allocation mechanism of NARS already implements a process similar to activation spreading in neural networks. When a new task is added into a concept, the priority of the concept is increased temporarily, and inference in the concept may cause derived tasks to be sent to its neighbors, so their priority (activation) levels will be increased, too. As a supplement, a mental operation allows the system to pay attention to a concept without new tasks added.

Feeling. The system can check the readings of its sensors embedded in its "body" and "mind", so as to "feel" its status, and use the reports to decide its actions. This mechanism has been described in [18]. Beside emotional status, the system can also feel how novel a new input is (so as to give it the attention it deserves) or how busy itself is (so as to decide its resource allocation strategy).

In general, mental operations supplement and influence the automatic control mechanism, and let certain actions be taken as the consequence of inference. Mental operations contribute to the system's self-concept by telling the system what is going on in its mind, and allow the system to control its own thinking process to a certain extent. For instance, the system can explicitly plan its processing of a certain type of task. After the design and implementation phases, the system needs to learn how to properly use its mental operations, just like it needs to learn about the other operations.

In NARS, "experience" refers to the system's input streams. In the simplest implementation of NARS, the system has only one input channel, where the experience is a stream of Narsese sentences like $S_1, T_1, S_2, T_2, \ldots, S_n, T_n$ from the channel, where each S_i is a Narsese sentence, with T_i to be the time interval between it and the next sentence. A buffer of a constant size n holds the most recent experience.

In more complicated implementations, there are also "sensory" channels each accepting a stream of Narsese terms from a sensory organ. Here a sensor can recognize a certain type of signal, either from the outside of the system (such as visual or audio signals), or from the inside of the system. Within the system, the sensation can come either from the body (somatosensory) or from the mind (mental). Such a channel provides a certain type of "internal experience". Somatosensory input will be especially important for a robotic system, as it needs to be aware of its energy level, network connection status, damages in parts, etc.

A mental sensation may come from the execution of a mental operation, such as the “feeling” operation mentioned above. Also, mental sensations appear as the trace of the system’s inference activity. During each inference cycle, the system “senses” the concept that was selected for processing, and the implication relationship between the premises and the conclusion. Later, this experience can be used to answer questions like “What has been pondered” or “Where does that conclusion come from”, asked either by the system itself or by someone else, as well as used in future inference activities.

On the input buffers the system carries out certain channel-specific preprocessing to form compound terms corresponding to the spatiotemporal patterns of the input. There is also a global buffer that holds a stream of Narsese sentences after preprocessing, where the terms typically combine the data from multiple channels. In this aspect, the external and internal experiences are handled basically in the same way.

A special type of belief formed in this way is the temporal implications between the mental events sensed within the system and the outside events observed by the system. The system will believe that it is some of its ideas that “cause” a certain action to be performed in its environment, and such beliefs will coordinate its “mind” and its “body”.

The internal experience of NARS is the major source of its self-knowledge. Under AIKR, this type of knowledge is also uncertain and incomplete, and is under constant revision. Furthermore, it is subjective and from the first-person perspective. In these aspects, NARS is fundamentally different from the “logical AI” approach [9].

There is no space in this paper to provide working examples, so interested readers should visit the OpenNARS project website.[5]

3 Comparison to Related Work

Restricted by paper length, here we only compare NARS with the related AI works, and not address the huge literature in psychology and philosophy on self, consciousness, and the related topics.

Though many approaches have been proposed for self-awareness and self-control in various forms, most AI systems do not have a “self” concept (no matter under what name) [5]. Such a concept is used in NARS, mainly because *concept* provides a flexible unit for representation and processing, so every identifiable pattern in experience and notion in thought is handled as a concept. Since an intelligent system has the needs to know about itself, it is natural for such a concept to be used to collect all the self-related beliefs and tasks together.

According to the semantics of NARS, the meaning of a concept (or a term naming a concept) is completely determined by its relation with other concepts (or terms). While for most concepts such relations are all acquired from the system’s experience, the system is not born with a blank memory. Each built-in operation contributes meaning to

---

[5] Source code, working examples, and documentations of the current implementation of NARS can be found at `http://opennars.github.io/opennars/`

the concept of $SELF$, by relating the system as a whole to the events it can perceive and/or realize. Starting from these operations, the $SELF$ concept will eventually involve beliefs about

- what the system can sense and do, not only using the built-in operations, but also the compound operations recursively composed from them, as well as the preconditions and consequences of these operations;
- what the system desires and actively pursues, that is, its motivational and emotional structure;
- how the system is related to the objects and events in the environment, in term of their significance and affordance to the system;
- how the system is related to the other systems, that is, the "social roles" played by the system, as well as the conversions in communication and interactions.

All these aspects will make the system's self concept richer and richer, even to the level of complexity that we can meaningfully talk about its "personality", that is, what makes this system different from the others, due to its unique nature and nurture.

This treatment is fundamentally different from identifying "self" with a physical body or a constant mechanism within the system. "Self" is not left completely to a mysterious "emergent process", neither. In NARS, the concept of "self" starts with a built-in core, then evolves according to the system's experience. In the process, the self-concept organizes the relevant beliefs and tasks together to facilitate self-awareness and self-control. This is consistent with Piaget's theory that a child learns about self and environment by coordinating sensations (such as vision and hearing) with actions (such as grasping, sucking, and stepping), and gradually progresses from reflexive, instinctual action at birth to symbolic mental operations [11].

A widely agreed conclusion in psychology is that a mental process can be either automatic (implicit, unconscious) or controlled (explicit, conscious), with respect to the system itself. The former includes innate or acquired stimulus-response associations, while the latter includes processes under *cognitive control*, such as "response inhibition, attentional bias, performance monitoring, conflict monitoring, response priming, task setting, task switching, and the setting of subsystem parameters, as well as working memory control functions such as monitoring, maintenance, updating, and gating." [4]

Various "dual-process" models have been proposed in psychology to cover both mechanisms. Such models are also needed in AI, even though the purpose here is not to simulate the human mind in all details, but to benefits from the advantages of both. In general, controlled processes are more flexible and adaptive, while automatic processes are more efficient and reliable. In such a system, there are meta-level processes that regulate object-level processes [5, 8, 13, 12], and such works are also covered in the study of machine consciousness [1, 3]. Even though this "object-level vs. meta-level" distinction exists in many systems, the exact form of the boundary between the two differ greatly, partly because of the architecture of the systems involved. A process should not be considered "meta" merely because it gets information from another process and also influences the latter, since the relation can be symmetric between the two, while normally the object-level processes have no access to the meta-level processes.

As a reasoning system, in NARS "control" means to select the premises and the rule(s) for each inference step, so as to link the individual inference steps into problem-solving processes. The primary control mechanism is coded in a programming language, and is independent of the system's experience. It is automatic and unconscious, in the sense that the system does not "think" about what to do in each step, but is context- and data-driven, while the data involved comes from associations biased by dynamic priority distributions. On top of this, there are mental operations that are expressed in Narsese and invoked by the system's decisions, as a result of "conscious" inference activities. This meta-level deliberative control does not change the underlying automatic routines, but supplement and adjust them.

Deliberative control in NARS is mainly achieved by mental operations, and this treatment is different from the meta-cognition implemented in the other systems [5] in that the operations in NARS are light-weight, rather than decision-making procedures that compare the possible actions in detail with a high computational cost. Also, the preconditions of these operations are largely learned from experience, not predetermined. As these operations can be combined into compounds, the system will gradually learn problem-solving skills, as a form of self-programming [16].

In general, NARS treats its "external experience" and "internal experience" in the same way, and the knowledge about the system itself has the same nature as other knowledge in NARS. Under AIKR, self-knowledge is incomplete, uncertain, and often inconsistent, which is the contrary of what is assumed by the "logical AI" school [9]. The system can only be aware of the knowledge reported by certain mental operations and those in the input buffers, and even this knowledge does not necessarily get enough attention to reveal its implications. The control aspect is the same, that is, the system can only make limited adjustments, so cannot "completely reprogram itself", and nor can it guarantee the absolute correctness of its self-control behaviors.

If self-awareness and self-control are required in an intelligent system, why are such functions absent in most of the AI systems developed so far?

Like many controversies in AI, the different opinions on this matter can be traced back to the different understandings of "AI" [15]. As the mainstream AI aims at the solving of specific problems, the systems are usually equipped with problem-specific algorithms. Even in learning systems that do not demand manually-coded algorithms, they are still approximated by generalizing training data. In general such systems have little need to add itself into the picture, and even meta-cognition can be carried out without an explicit "self" concept involved [5].

In AGI systems, the situation is different. Here we have projects aimed at simulating the human brain according to psychological theories [6, 2], which surely needs to simulate the self-related cognitive functions. Even in the function-oriented projects, self-awareness and self-control are desired to meet the requirements for the system to work in various situations [13, 12].

For NARS, the need for self-awareness and self-control follows from its working definition of intelligence, that is, adaptation under AIKR [15]. To adapt to the environment and to carry out its tasks, the system needs to know what it can do and how it is related to the objects and other systems in the environment, and an explicitly expressed "self" will organize all the related knowledge together, so as to facilitate reasoning and decision making.

NARS treats $SELF$ like other concepts in the system, except that it is a "reserved word" which has innate associations with the built-in operations, including the mental operations. NARS also treats internal and external experience uniformly, so self-awareness and self-control have nothing magical or mysterious, but are similar to how the system perceives and acts upon the external environment.

Though the study of self-awareness and self-control in NARS is still at an early stage, the conceptual design described above has been implemented, and is under testing and tuning. There are many details to be refined, however we believe the overall design is in agreement with the scientific knowledge on these processes in the human mind, and also meets the needs and restrictions in AGI systems.

References

1. Baars, B.J., Franklin, S.: Consciousness is computational: The LIDA model of global workspace theory. International Journal of Machine Consciousness 1, 23–32 (2009)
2. Bach, J.: Principles of Synthetic Intelligence PSI: An Architecture of Motivated Cognition. Oxford University Press, Oxford (2009)
3. Chella, A., Frixione, M., Gaglio, S.: A cognitive architecture for robot self-consciousness. Artificial Intelligence in Medicine 44, 147–154 (2008)
4. Cooper, R.P.: Cognitive control: Componential or emergent? Topics in Cognitive Science 2, 598–613 (2010)
5. Cox, M.T.: Metacognition in computation: A selected research review. Artificial Intelligence 169, 104–141 (2005)
6. Franklin, S.: A foundational architecture for artificial general intelligence. In: Goertzel, B., Wang, P. (eds.) Advance of Artificial General Intelligence, pp. 36–54. IOS Press, Amsterdam (2007)
7. Kowalski, R.: Logic for Problem Solving. North Holland, New York (1979)
8. Marshall, J.B.: A self-watching model of analogy-making and perception. Journal of Experimental & Theoretical Artificial Intelligence 18(3), 267–307 (2006)
9. McCarthy, J.: Making robots conscious of their mental states. In: Machine Intelligence 15, Intelligent Agents [St. Catherine's College, Oxford, July 1995]. pp. 3–17 (1995)
10. Peirce, C.S.: Collected Papers of Charles Sanders Peirce, vol. 2. Harvard University Press, Cambridge, Massachusetts (1931)
11. Piaget, J.: The Origins of Intelligence in Children. W.W. Norton & Company, Inc., New York (1963), translated by M. Cook
12. Rosenbloom, P.S., Demski, A., Ustun, V.: The Sigma cognitive architecture and system: Towards functionally elegant grand unification. Journal of Artificial General Intelligence 7, 1–103 (2016)
13. Shapiro, S.C., Bona, J.P.: The GLAIR cognitive architecture. International Journal of Machine Consciousness 2, 307–332 (2010)
14. Wang, P.: Rigid Flexibility: The Logic of Intelligence. Springer, Dordrecht (2006)
15. Wang, P.: What do you mean by 'AI'. In: Proceedings of the First Conference on Artificial General Intelligence. pp. 362–373 (2008)
16. Wang, P.: Solving a problem with or without a program. Journal of Artificial General Intelligence 3(3), 43–73 (2012)
17. Wang, P.: Non-Axiomatic Logic: A Model of Intelligent Reasoning. World Scientific, Singapore (2013)
18. Wang, P., Talanov, M., Hammer, P.: The emotional mechanisms in NARS. In: Proceedings of the Ninth Conference on Artificial General Intelligence. pp. 150–159 (2016)