Sensorimotor in NARS

Pei Wang, Xiang Li, Patrick Hammer
Department of Computer and Information Sciences
Temple University, USA
{pei.wang,xiangliAGI,patrick.hammer}@temple.edu

May 12, 2019

Abstract

This report provides a comprehensive description of the conceptual design of the sensorimotor mechanism in NARS, which is an extension of the previous versions of the system obtained by giving terms procedural interpretations and introducing temporal and spatial compound terms.

1 Overview

Generally speaking, the "sensorimotor interface" of a cognitive or intelligent system is between the system and the part of the environment that is not cognitive or intelligent in any sense. Therefore, it is very different from the communication interface that uses convention-based languages to interact with other systems. Usually the system is considered as equipped with a constant set of sensors and actuators at any given period, and its sensorimotor experience is a stream of their executions. For an adaptive system, the basic function of the sensorimotor mechanism is to support the system's adaptation by providing generalizations of the experience, so as for the system to handle the current and future situations according to what happened to similar situations in the past, so as to achieve its goals as well as possible.

This report summarizes the conceptual design of sensorimotor mechanism in NARS, as an extension of the model described in [Wang, 2013]. The basic idea is to let terms identify sensations, perceptions, operations, and so on, and to carry out the sensorimotor functions as various types of inference, as introduced in the previous publications [Wang, 2006, Wang, 2012, Wang and Hammer, 2015, Hammer et al., 2016, Wang et al., 2018b, Wang et al., 2018a].

This approach is inspired by many previous works in artificial intelligence and cognitive science, especially logic programming [Kowalski, 1979] and the "enactive" view of perception [Kevin O'Regan and Noë, 2001]. It agrees with the neural network models [Hawkins and Blakeslee, 2004, LeCun et al., 2015] in taking perception as multilevel generalization. Compared with the other approaches, the most distinguished features of this approach include

- Do not see the system's knowledge as a model of the world, but as a summary of the system's own experience, biased by the system's motivations;
- Do not treat perception and action as two separate processes handled by different modules, but as an unified sensorimotor mechanism driven by the system and with external and internal consequences;
- Do not treat sensorimotor and high-level cognition as carried out by different mechanisms, but as by a unified reasoning–learning capability;
- Completely acknowledge the Assumption of Insufficient Knowledge and Resources (AIKR).

2 Term and concept

NARS is based on the belief that intelligence and cognition are essentially conceptual-level phenomena, rather than neural-level activities [Wang, 2006]. Therefore, the uniform representation within NARS is centered at *term* and *concept*. Intuitively speaking, in NARS a 'concept' is a recognizable unit in the description of the system's experience, and a 'term' is an identifier, or internal name, of a concept.

Different from the interpretable "symbols" in the "symbolic AI" tradition [Newell and Simon, 1976], in NARS a term (or a concept) does not represent an object or event in the outside world. Instead, as an ingredient or pattern in the experience, its meaning is grounded by its experienced relations with other terms (or concepts) [Wang, 2005]. As the meaning of each concept can be addressed individually using a term, this representation is also different from the fully distributed representation in a connectionist model [Hinton et al., 1986].

The simplest terms in NARS are *atomic*, without internal structure, and their names are arbitrary strings. Beside that, there are various types of *compound* term whose names consist of built-in connectors and component terms, and consequently the meaning of such a term is related to the meaning of its components in a manner specified by the connector. Even so, the experienced relations between such a term (as a whole) and the other terms still contribute to its meaning. Therefore, the meaning of a compound term can be said to be semi-literal (in the structure of its name) and semi-distributed (in its relations with the other terms), and the relative weight of the two aspects change from concept to concept.

Among the terms in NARS, the basic semantic relations are about whether one term can be generalized (or specialized) into another one, which are captured by the Narsese copulas *inheritance* ($\rightarrow$) and *similarity* ($\leftrightarrow$). With these two basic relations, the concepts in NARS form a generalization hierarchy, though concepts in the structure are not strictly organized into levels or layers. Between any terms x and y, "$x \rightarrow y$" and "$x \leftrightarrow y$" could be established, though most of them would have a truth-value with a low frequency or confidence, so will not contribute much in reasoning. Beside this *inheritance/similarity* network, NARS also has a network among the *statements* (which are terms with truth-value), formed by the copula *implication* ($\Rightarrow$) and *equivalence* ($\Leftrightarrow$). These two correspond to the derivation relation among statements, and indicate the correlations among their truth-values. Intuitively speaking, these two concept networks represent the (symmetric or asymmetric) substitutability of terms/concepts in meaning and truth-value, respectively. The topological structures of both networks change constantly as the system runs, and so are the attributes of the nodes and links (priority values, truth values, etc.).

In the previous versions of NARS [Wang, 1995, Wang, 2006, Wang, 2013], the system's experience is a stream of sentence in Narsese, the representation language of NARS. Even in this form, the terms are already "grounded", in the sense that their meaning to the system is not provided by an interpretation that maps them into outsider objects and events, but determined by the roles they play in the system's experience. Of course, in this situation the meaning of terms is much simpler than that in the human mind. Even so, given the nature of term and concept in NARS, it is relatively easy to extend the system' experience to include sensorimotor by letting the system directly interact with the environment, rather than only with other systems via a common language. Sensorimotor experience is different from linguistic experience obtained from the communication with another system, but it is not the only way for the terms to be "grounded" to have meaning [Wang, 2009].

Since a term is just the identifier of a data structure obtained from the system's experience, it can correspond not only to a character string used in communication, but also ingredients and their temporal-spatial patterns in the system's direct (physical, chemical, etc.) interaction with the environment.

To expend the experience of the system from purely linguistic to sensorimotor, the first step is to use a term to name an *operator* that corresponds to an executable routine of the system [Wang, 2012, Wang, 2013]. In NARS, such a term has a special prefix '$\Uparrow$' in name, and its meaning is partly procedural (revealed by the routine) and partly declarative (revealed by its conceptual relations). By applying an operator '$\Uparrow op$' to arguments $a_1, \cdots, a_n$, an operation "$\Uparrow op(a_1, \cdots, a_n)$" is formed, where the order of arguments matters. When treated in reasoning, this operation is semantically equivalent to inheritance statement "$(\times \{SELF\}\ a_1\ \cdots\ a_n) \ \rightarrow \Uparrow op$", that is, the system itself and the arguments form a relation indicated by the operator.

NARS allows the same operator to be applied to argument lists of different lengths, though the related knowledge will be different, too. An argument can be a constant term or a variable term. When an operation is described in its general form with formal arguments, an independent variable indicates an input argument, and a dependent variable indicates an output argument. So operation "$\Uparrow add(\$x, \$y, \#z)$" represents the function $z = add(x, y)$. When an operation is executed, its input arguments must be already instantiated by constant values, such as $\Uparrow add(2, 3, \#z)$, then the input arguments will be passed to the corresponding routine, and after the execution the output arguments are also instantiated, as $\Uparrow add(2, 3, 5)$. In this aspect, the process in NARS is similar to that of logic programming.

An operation can be carried out either by NARS itself or by a connected device outside NARS, and its consequences can be either 'mental' (inside the system, as adding some numbers) or 'physical' (outside the system, as driving a printer). It means NARS can control robots or other devices by issuing commands to them, and their immediate feedback will be taken as the output of the operations. Such a general-purpose sensorimotor interface has been implemented in OpenNARS [Hammer et al., 2016]. When the operation is on the internal environment, its consequences realize self-control, as specified in NAL-9 [Wang, 2013].

The notion of 'operator' in NARS includes both 'sensor' and 'actuator' in the conventional AI systems. Within NARS this distinction can still be made between the operators whose main function is to acquire information from those whose main function is to change the environment, though this distinction is no longer clear-cut. Similarly, we can still consider some terms as abstract 'concepts' that can be named in a communication language, while some others as concrete 'percepts' which can be felt but cannot be explicitly expressed, though this distinction is not fundamental, neither. All these terms will be treated in NARS in very similar manners.

Compared to many cognitive architectures, like Soar [Newell, 1990] and LIDA [Franklin, 2007], NARS is different in that it has a single memory containing concepts of various types, and therefore the operators and percepts are also organized into the *inheritance-similarity* network with the abstract concepts, and the actions and perceptions into the *implication-equivalence* network with the abstract statements. In this way, the "abstract vs. concrete" distinction becomes a matter of degree, indicating the distance of a concept to atomic operations. Similarly, the traditional "conceptual vs. subconceptual" distinction fades away, as the meaning of a concept in NARS may not be fully expressible in Narsese, but is revealed only by operations (including perceptions).

3 Temporal terms

Some terms used in sensation and perception correspond to the arguments of the operations executed by the sensors and actuators. Beside the (partial) procedural interpretation, they differ from the generic (non-sensorimotor) terms in that their meaning also has temporal and/or spatial aspects, and so do the related term connectors.

As described in [Wang, 2013, Wang and Hammer, 2015], temporal information is represented in NARS in several ways, including an internal clock, temporal version of copula and connectors, and the related concepts.

When events a and b are combined into sequence (a, b), their temporal distance is taken into account when the simplicity of (a, b) is evaluated. Consequently, if a and b are not close in timing, their succession can still be perceived as a compound (a, b), though the chance will be relatively lower than the case where they are consequent. The use of (a, b) does not mean that there is no other events in between, but that they are ignored when this compound is used. The same is the case for the other compounds formed from them, such as temporal implication $(a /\!\!\!\Rightarrow b)$.

The temporal compounds of operation are of special importance, as they carries out cognitive functions of "skills" and "programs" [Wang, 2012]. In [Wang, 2013], three operation connectors have been described:

Sequential: If P and Q are operations, so is (P, Q), which means to execute them in that order;

Parallel: If P and Q are operations, so is $(P; Q)$, which means to execute them together;

Conditional: If P is a statement and Q is an operation, $(P \Rightarrow Q)$ is also an operation, which means to execute Q whenever P is true.

These connectors can be used recursively to form more complicated compounds, so as to use Narsese effectively as a programming language with a given set of basic operations.

There are several possible ways to represent an iteration or loop in Narsese:

- By recursion. This approach works well in logic programming, such as in Prolog, so should work in NARS, too. The complexity may come from the concurrent processing of derived tasks in NARS, which conflict with the sequential nature of recursion.
- To treat repeated operation as a special type of conditional operation. In a production (rule-based) system, repeated operations can be achieved by repeatedly using the same rule, like to treat a *while* statement as a repeated *if* statement by using the same operation to gradually achieve a goal—as long as the goal is still there, the same operation can be executed to approach it. This is already supported in OpenNARS.
- To depend on a mental operation *count* to carry out the function of a *for* loop. As this operation needs to be introduced anyway to initiate mathematical ability, it may also be used for this purpose.

We will explore the above possibilities in parallel, as well as in combination. If it turns out to be necessary, new term connectors can be introduced. For example, $(P \infty\!\!\Rightarrow Q)$ can be treated as $(P \Rightarrow Q)$, except it will be executed repeatedly until the condition is no longer true.

Another capability needed for "self-programing" (i.e., compound-operation composition at run time) is the encapsulation of a compound operation. It will turn a compound operation into an atomic operation, so as to reduce its syntactic complexity, as the encapsulation and compilation in programming language. In NARS, this will be implemented as a special application of the *name* mental operator, which will coin a new atomic term for a frequently used compound term and ignore its internal structure. When the compound is an operation, the result will be an equivalence statement between the compound operation (the "body") and an atomic operation (the "interface"). Compared to the general applications of the *name* operator, here the challenge is to identify the input and output arguments to be kept in the interface, though a simple form of this mechanism already exists in the variable-introducing comparison rules that produce similarity statements between an atom and a compound. What is needed is to extend it to meet the need of compound operations in general.

This *encapsulation* operation should be triggered both *automatically* with the composition of qualified compound operation and *deliberately* as the result of a reasoning process.

Temporal inference should also cover the manipulation of "temporary beliefs" that is maintained only in a certain period of time for some special type of term that is neither the constant term nor the variable term in the current NARS:

Constant term: The name of a unique concept in the whole system all the time. Example: *bird*, t_7908, etc. When the same term appear in multiple sentences, they always refer to the same concept. The meaning of a constant term changes with the relevant experience. Here 'constant' does not mean that its meaning cannot change, but indicate the permanent nature of the term–concept mapping.

Variable term: The name of a unique concept in a sentence. Such a term has a special prefix to indicate its type, and can be substituted by a constant term in reasoning. Example: $\$x$, $\#y\_12$, ?1, etc. Within a sentence, the meaning of a variable term is fully determined by its relations with other terms in that sentence. As $(\$x \rightarrow b) \Rightarrow (\$x \rightarrow c)$ means "A term in the extension of b is also in the extension of c", the $\$x$ can be substituted by f and g when $f \rightarrow b$ and $g \rightarrow b$ are true to a degree, respectively. The term $\$x$ is considered a variable because with different substitutions it can stand for different terms, and refer to different concepts. Here the term–concept relation is temporary, built only during the substitution.

Between these two clearly separated types, there are types of term with temporary term–concept relations with various duration and serve different purposes:

Abstract term: Such a term is similar to a constant term in that it refers to a unique concept in the system. However, the meaning of the term does not directly grounded in the system's experience. Instead, it comes from a definition or convention, and is connected to a constant term through an interpretation. Typical examples are the mathematical notions like "point" and "function".

Symbolic term: Such a term also refers to a unique concept in the system. However, its meaning mainly come from another (constant) term it 'represents' or 'symbolizes'. Typical examples are the linguistic terms in a language used in communication.

Container term: This is the variables in a programming language, which actually correspond to a piece of memory where the content can change from time to time, so the same variable name can have different values in different time.

The above three types of term have the same nature that their relations with concepts may change over time, and to be relevant to the system's behavior, eventually they need to be substituted by a constant term via "interpretation", "grounding", "assignment", "instantiation", or whatever it is called. In NARS, they all can be implemented as similarity statements between such a term and a constant term, created by the mental operator *assume*, with a default truth-value. Depending on the context, there can be additional beliefs about this assumption, though this assumption by itself will allow substitutability between the two terms. Assumptions are different from normal beliefs in that they can be completely overridden by another mental operation. For example, a new assignment to a container term will cost an inconsistency between the old value and the new value, and when the term's "current value" is requested, the choice rule will return the most recent one.

4 Spatial terms

The initial form of spatial information is provided by the relative locations of multiple sensors of the same type that receive signals concurrently. As introduced in [Wang et al., 2018a], a sensational operation may return a multidimensional array as the result. In this case, the location of a sensor in this perceptive field can be indicated by an index or coordinate. A revision of the previous design in [Wang et al., 2018a] is that each sensory term returned by the sensor already has a center of focus, rather than added when they become perceptual terms. In this way, the distinction between sensory terms and perceptual terms is that the former is the raw data returned by the sensors, while the latter is the data that has been processed by the system.

There are two ways to form a perceptual term, using term connectors and mental operators, respectively.

Spatial patterns can be composed as compound terms from the arrays recursively by the Narsese connectors, including *conjunction*, *disjunction*, and *negation*. They are originally defined between individual statements [Wang, 2006], but can be naturally extended to take arrays of statements as arguments. When the arrays have different dimensions, the relative coordinates will aligns the corresponding truth-values properly [Wang et al., 2018a]. The resulting perceptual terms also have their center of focus.

This type of perceptual terms can be formed by inference rules to compose and decompose compound terms. For example, from premises $a \to b$ and $a \to c$ and their truth-values, a conclusion is $a \to (b \cap c)$ with a truth-value associated. The "syllogistic-like" feature of the inference rules in NARS makes the system to only form compound terms with semantically related components, like in the above $(b \cap c)$, b and c are related by a.

The set of mental operators introduced in [Wang et al., 2018a], *focus shifting, in/out zooming, orientation rotating*, can be considered as internalized version of sensor movements. For example, "moving forward" of the body consistently causes "zoom in" of the visual image, while movements in the other dimensions correspond to the "focus shifting". These operators can be used to form complicated perceptions efficiently, as well as to form "sensorimotor contingency" [Kevin O'Regan and Noë, 2001] corresponding to the invariant in sensorimotor coordination.

These mental operations will also provide the basic logical relations among the percepts. For example, zooming out from a given array A may get another array B, which is a "compressed" version of A as being smaller in size, lower in resolution, while still showing the same pattern, so $A \to B$ can be established with a truth-value indicating the quality of this compression.

Unlike the inference rules (which are triggered in a data-driven manner in NARS), the mental operators are usually triggered by goals deliberately, as a result of reasoning.

Using the above connectors and operators, a perceptual term represents a pattern perceived by the system, which is how a "mental image" or an "imagined object or scene" is represented in memory. In general, a mental image is not necessarily represented by a matrix of pixels.

Most sensory and perceptual terms will be eliminated in the resource competition soon, and the retained terms need to have a relatively high budget that is mainly contributed by the following three factors, as the other (non-sensorimotor) concepts:

1. Corresponding to frequent patterns in experience
2. Useful in processing the tasks of the system
3. Relatively simple

The special feature of the sensorimotor terms is that the components in a compound term are not related semantically (via the copulas), but temporally or spatially. As suggested by Gestalt psychology, the temporal/spatial distance among the components play a major role in deciding the relative easiness for a compound term to be formed and kept.

- Some formulas are needed for "spatial projection" in a way similar to the "temporal projection" [Hammer et al., 2016], so that spatially remote components can only form less competent compounds;
- For compound with repeated components, the representation should use iterative structures, as discussed in the context of operation.

Eventually, perceptual terms and operational terms will be combined into procedural knowledge about the invariant conditions and consequences of each operation [Wang, 2012], which will play similar functions as the "schemata" [Piaget, 1963] and "sensorimotor contingency" [Kevin O'Regan and Noë, 2001] in psychology and cognitive science.

In NARS, temporal information and spatial information are handled together in similar ways, but time is not treated as the 4th dimension just as the other spatial dimensions, as time has certain special property that a spatial dimension does not have. For example, "past" and "future" are not fully symmetric as the opposite directions in space.

5 Internal sensorimotor

As an advanced intelligent system, NARS needs to perceive and act on events within its own body and mind [Wang, 2013, Wang et al., 2017]. The "physical" sensorimotor mechanism on its body is quite similar to the external sensorimotor in that it depends on the details of the host hardware, such as the body of a robot, and will be handled via the same sensorimotor interface. On the contrary, the "mental" sensorimotor mechanism is different in that it is mostly independent of the body, and therefore can be considered as part of NARS, and that the sensors and actuators can be directly described at the conceptual level, without the multilevel generalization starting from sensation.

In principle, all internal activities of NARS can be controlled by the system itself. However, under AIKR only some of them will be treated this way, so as to achieve a balance between the stability and efficiency provided by the automatic processes and the adaptivity and flexibility provided by the controlled processes. In general, the former is necessary and primary, while the latter is optional and supplementary. In NARS, the automatic processes are coded into the control mechanisms, while the controlled processes are represented by the mental operations introduced in NAL-9 [Wang, 2013].

The mental operations can be roughly divided into two major groups: those that directly correspond to the automatic activities, and those that evaluate the overall situations.

At the top level, the running of NARS consists of two types of activity: concept firing and task preprocessing. In the future versions, they will be handled as separated processes that can run at different speeds, rather than as two steps of the working cycle as in the current implementation.

Concept firing is the main working cycle of the system, which goes through the following steps

1. Select a concept,
2. Select a related task,
3. Select a related belief,
4. Use the task and belief as premises to derive new tasks.

At the end, the budgets of the involved items (concept, task, belief) are adjusted according the current result.

To keep the continuity of the reasoning process, it may be better to select multiple beliefs for the same task, and to select multiple tasks in the same concept. These adjustments should not change the status of concept as a unit of storage and processing, nor the requirement that each working cycle takes a roughly constant amount of time to finish.

The selections in the above process can be directly carried out as operations, with the corresponding input argument, and the selected item as output, so the system can directly work on a concept or a task in a concept, without waiting for them to pop up as the consequence of attention allocation. On the other hand, the selections made by the automatic processes can be remembered as events in the system's experience, which may lead to their deliberative invoke in the future.

For the reasoning step that from a selected task with content t_1 to a derived task with content t_2, by induction the system can get "$t_1 \Rightarrow t_2$" and "$t_2 \Rightarrow t_1$" for forward and backward inference, respectively. In this way, the belief used is merged into the implicit background. Another implication statement can be generated with the belief in the condition, too, but that result is redundant as that relation is already coded into the inference rule that derives the conclusion. That is, conclusion "$((s \to m) \wedge (m \to p)) \Rightarrow (s \to p)$" is not really useful, but "$(m \to p) \Rightarrow (s \to p)$" is.

Task preprocessing currently selects a new task from the task buffer, then insert it into the related concepts. If the task is a judgment, carry out revision and belief generation; if it is a goal or question, check for an existing solution. After that the task is added into the task bag and wait for its turn to derive new tasks. This process can be invoked by an operation, too.

When sensorimotor is added into NARS, the observations and actions are also take the form of mental operations, and perception is similar to the above task preprocessing. By observing and summarizing its own actions, the system can gradually learn how to use its operations, similar to the "self-modeling" process [Kwiatkowski and Lipson, 2019].

There are two possible ways to implement the above design:

1. To "operationalize" the automatic working routine by re-factoring the code as a sequence of operations. This is conceptually cleaner, but may be too complicated to realize, as it demands major changes in the code. This solution will be considered when OpenNARS is completely redesigned at a future time.

2. To let the automatic working routine generates events reporting what it is doing, and to code the operations separately so they can do something similar, though not necessarily identical to the working routine. This should be feasible in the current code base, and will be attempted at the current stage.

The "feeling" operations do not directly intervene with the reasoning process, but are responsible for evaluations and appraisals at the whole system level.

At the individual data items, the following attributes can "be felt" by the system:

Priority value of a concept, belief, or task;

Truth value of a statement;

Novelty value of a judgment task, as the difference of its expectation and the previous expectation of the statement in the system;

Satisfaction value of a question task, as the expectation of the best solution found so far; for a goal task, as the complement of the difference between its desire-value and truth-value of the best solution.

At the system level, the major status indicators include:

Satisfaction is a weighted average of the satisfaction values of the recently processed goal and question tasks, biased by the recentness and priority of the task;

Alertness is a weighted average of the novelty values of the recently processed judgment tasks, biased by the recentness and priority of the task;

Busyness is a weighted average of the priority values of the recently processed tasks, biased by the recentness of the task;

Health will be a summary of the body-related sensors indicating whether the corresponding values are in the 'normal' range.

Overall, these four indicators form the meta-goals of the system, where **Satisfaction** and **Health** indicate the mental and physical well-being of the system, respectively, and will drive the system to increase them. On the other hand, **Alertness** and **Busyness** indicate the relative sufficiency of the system's knowledge and resources, respectively, with respect to its current tasks, and will drive the system to keep them in a range in the middle of possible values. They are not "supergoals" as some people assume that will directly guide all behaviors of the system, as most of the tasks and beliefs are not directly associated with them.

The mental operator *feel* will take each of these four as input argument, and report the current value as an event in the system's internal experience. Built-in terms are needed to express the above feelings in Narsese. These operations can either be invoked deliberately, or triggered automatically by an above-threshold value of the indicator. In either case, the system will be driven by them to improve or remedy the situation.

In summary, the mental operations will serve both as the meta-goals and the concrete tasks by overriding the automatic working routine, and realizing self-programming.

6 Sensorimotor inference

In NARS, *sensation* is carried out by the sensors, in which the various signals are converted uniformly into sensory terms. *Perception* corresponds to the construction of perceptual terms with conceptual relations, under the guidance of the system's current motivations, drives, and goals.

When the physical signal comes to the system as a continues flow (as in vision), what the *looking* operations returns are sensory terms acquired according to the system's (conscious or not) decisions on the orientation of the sensors and the sampling rate, and the system's visual experiences is a temporal array of spatial arrays.

For each sensor of type t, the observation operation is invoked to answer the question "$?s \rightarrow [t]$" (intuitively, "What is out there?") or "$\{s\} \rightarrow [t]?$" if there is already an anticipated result. The returned sensory term s will answer the question, and derive a recognition question "$\{s\} \rightarrow ?x$" meaning "What is that s?" which asks for proper descriptions of the sensation.

Different sensorimotor channels may have different "observing frequency", adjustable by the attention allocation mechanism. The same is true for the actuators, as the system needs to "babble" to reveal the conditions and consequences of its operations, especially at the early stage. It probably can be handled in the same way as the concept bag where each concept has an adjustable "firing frequency" determined by its budget.

The same section of sensation flow can be perceived differently, depending on various factors, including the priority of the resulting compound terms. When there are several coherent perceptions corresponding to different levels of generalization and granularity, then they will co-exist with *inheritance* copulas in between. Sometimes incoherent perceptions will compete, and some effort will be paid to disambiguate the situation, mostly using the choice rule.

Temporally or spatially distinct patterns will form *implication* or *equivalent* conclusions via induction and comparison. These conclusions will start with low confidence values, but can be strength or weaken by further experience. For example, from premises $a \rightarrow b$ and $a \rightarrow c$ with their truth-values, the induction rule can generate $b \rightarrow c$, $(a \rightarrow b) \Rightarrow (a \rightarrow c)$, and $(\$x \rightarrow b) \Rightarrow (\$x \rightarrow c)$, with the same truth-value. When the two premises are also temporally or spatially related, the conclusions will correspond to different temporal or spatial patterns. Though justified in the same way, these conclusions may end up with very different priority values.

For each type of sensory signal, there is a separate I/O channel as local storage and processing unit. When the observation operation is invoked in the channel, the current perceptual results are entered into the system's experience, as a task in Narsese. Within each I/O channel, some preprocessing happens automatically, which produces temporal–spatial compound terms and statements using the local information within the channel. There is a system-wide experience buffer where tasks from all I/O channels and the inference activities are pooled, where cross-modal perception happens, and the compound terms forms can have components from different sources and of different types. All the buffers in I/O channel only provide short-term memory, while the long-term memory contains concepts with references to tasks and beliefs.

Perception is accomplished by creating inheritance/similarity statements between the current content of the input buffer and existing concepts, though the categorization is not necessary unique and conclusive. Under AIKR, NARS will not general all possible compound terms then select the good ones as the result of perception. Instead, the composition activities start from the operations that drive the sensors. For a sensory term returned from a sensor, the system first try to recognize it as an instance of an existing concept. When no good candidate can be found, the term is processed (such as using zoom-in and focus-shift) until recognizable components are found, which are then combined to form a complete description of the scene. For example, visual patters will not come from randomly or systematically formed matrices using pixels, but as fragments of sensory terms, that is, part of the experience that has actually happened to the system. There will be perceptual terms coming from the system's imagination driven by conceptual description, which may be never experienced by the system, though their components are still derived from the available evidence.

Except the mental operations occurring in the channels, there will be a global (system-wide) operation bag to serialize the executions of the operations. Unless the reasoning activities that can be massively parallel, many operations should be executed sequentially, as they may demand to use system resources exclusively, as well as to produce effects that change the execution context of the following operations.

The overall learning process for sensorimotor skills probably will roughly go through the Piaget's stages [Piaget, 1963]: initially the relatively stable skills are the *sensorimotor* associations where a perception directly trigger an operation, as the more complicated skills are not matured yet. All the inheritance statements formed by sensory operations probably all have the form of $\{s\} \rightarrow [p]$, that is, with the instance-property copula, or, equivalently, with the most specific subject and most general predicate. It is only after generalizations that the terms in the middle are gradually introduced. Gradually, the skills with general terms and variable terms get higher priority, as they cover wider territory than the concrete statements. Finally, the system can use one term as a symbol of another one, so as to carry out abstract and hypothetical inference.

Similarly, the initial implication statements record the temporal order among the observed events, and especially the preconditions and consequences of the operations. It is from these experiences more general beliefs, including those about causal relations, are formed, and gradually grown into "objective knowledge" without mentioning the subjects (self or other agents).

According to this design, new perceptive patterns start from the experienced pattern, and then the system tries to generalize or simplify it in various ways. In the long run the acquired temporal-spatial concepts are those that correspond to the relatively simple, frequent, and useful temporal-spatial patterns in the experience, and their meaning also partially come from their relations with the "abstract" concepts that are not directly associated with sensorimotor.

Overall, inference on procedural knowledge can be seen as a form of simulation, where the system predicts the consequence of its actions before actually takes such actions. Therefore it plays a crucial role in the system's adaptation process.

7 Summary

As other extensions and applications of the NARS model, new functions should be introduced and evaluated in three scopes:

Narsese: First, the language should be properly extended to represent the new compounds. For sensorimotor, it corresponds to the temporal compounds of events, with operations as a special case, and the multidimensional arrays for spatial patterns. Furthermore, a set of mental operators are added.

NAL: In principle, no new inference rule is introduced for sensorimotor, though some existing rules will have additional variants to handle temporal–spatial inference, as those on arrays. The initially triggering condition of the mental operators are also needed.

NARS: The major additions in control are the filtering of temporal–spatial compounds and the operationalization of the working process. After that, the testing and tuning of the whole system will be carried out as usual.

This design naturally allows sensation and perception to be treated as interwoven with operations, as argued by the enactivists in cognitive science [Kevin O'Regan and Noë, 2001]. This approach is fundamentally different from the traditional "perceive-decide-act" cycle accepted in most AI systems. Instead, the system repeatedly executes various operations, though the major function for some of them is to report the situations of the relevant aspects of the current environment, both inside and outside.

References

[Franklin, 2007] Franklin, S. (2007). A foundational architecture for artificial general intelligence. In Goertzel, B. and Wang, P., editors, *Advance of Artificial General Intelligence*, pages 36–54. IOS Press, Amsterdam.

[Hammer et al., 2016] Hammer, P., Lofthouse, T., and Wang, P. (2016). The OpenNARS implementation of the Non-Axiomatic Reasoning System. In *Proceedings of the Ninth Conference on Artificial General Intelligence*, pages 160–170.

[Hawkins and Blakeslee, 2004] Hawkins, J. and Blakeslee, S. (2004). *On Intelligence*. Times Books, New York.

[Hinton et al., 1986] Hinton, G. E., McClelland, J. L., and Rumelhart, D. E. (1986). Distributed representation. In Rumelhart, D. E. and McClelland, J. L., editors, *Parallel Distributed Processing: Exploration in the Microstructure of cognition, Vol. 1, Foundations*, pages 77–109. MIT Press, Cambridge, Massachusetts.

[Kevin O'Regan and Noë, 2001] Kevin O'Regan, J. and Noë, A. (2001). A sensorimotor account of vision and visual consciousness. *The Behavioral and brain sciences*, 24:939–73; discussion 973.

[Kowalski, 1979] Kowalski, R. (1979). *Logic for Problem Solving*. North Holland, New York.

[Kwiatkowski and Lipson, 2019] Kwiatkowski, R. and Lipson, H. (2019). Task-agnostic self-modeling machines. *Science Robotics*, 4(26):eaau9354.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep Learning. *Nature*, 521:436–444.

[Newell, 1990] Newell, A. (1990). *Unified Theories of Cognition*. Harvard University Press, Cambridge, Massachusetts.

[Newell and Simon, 1976] Newell, A. and Simon, H. A. (1976). Computer science as empirical inquiry: symbols and search. *Communications of the ACM*, 19(3):113–126.

[Piaget, 1963] Piaget, J. (1963). *The Origins of Intelligence in Children*. W.W. Norton & Company, Inc., New York. Translated by M. Cook.

[Wang et al., 2018a] Wang, P., , and Hammer, P. (2018a). Perception from an AGI perspective. In *Proceedings of the Eleventh Conference on Artificial General Intelligence*, pages 259–269.

[Wang, 1995] Wang, P. (1995). *Non-Axiomatic Reasoning System: Exploring the Essence of Intelligence*. PhD thesis, Indiana University.

[Wang, 2005] Wang, P. (2005). Experience-grounded semantics: a theory for intelligent systems. *Cognitive Systems Research*, 6(4):282–302.

[Wang, 2006] Wang, P. (2006). *Rigid Flexibility: The Logic of Intelligence.* Springer, Dordrecht.

[Wang, 2009] Wang, P. (2009). Embodiment: Does a laptop have a body? In *Proceedings of the Second Conference on Artificial General Intelligence*, pages 174–179.

[Wang, 2012] Wang, P. (2012). Solving a problem with or without a program. *Journal of Artificial General Intelligence*, 3(3):43–73.

[Wang, 2013] Wang, P. (2013). *Non-Axiomatic Logic: A Model of Intelligent Reasoning.* World Scientific, Singapore.

[Wang and Hammer, 2015] Wang, P. and Hammer, P. (2015). Issues in temporal and causal inference. In *Proceedings of the Eighth Conference on Artificial General Intelligence*, pages 208–217.

[Wang et al., 2017] Wang, P., Li, X., and Hammer, P. (2017). Self-awareness and self-control in NARS. In *Proceedings of the Tenth Conference on Artificial General Intelligence*, pages 33–43.

[Wang et al., 2018b] Wang, P., Li, X., and Hammer, P. (2018b). Self in NARS, an AGI system. *Frontiers in Robotics and AI*, 5:20.