ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/355999035

Piaget revisited: from action schemas to mental representations

Preprint · November 2021

CITATIONS
0

READS
238

1 author:

Pierre Bonzon
University of Lausanne

42 PUBLICATIONS 89 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Linking neural dynamics to behaviors View project

Project Gestão do Conhecimento View project

All content following this page was uploaded by Pierre Bonzon on 08 November 2021.

The user has requested enhancement of the downloaded file.

Piaget revisited: from action schemas to mental representations

Pierre Bonzon
University of Lausanne, Faculty of Economics, Dept of Information Systems
pierre.bonzon@unil.ch

Abstract

According to Piaget's cognitive development theory, infant behaviors are first driven by action schemas. Cognitive neuroscience, which aims at linking cognition to brain processes, hasn't succeeded yet in grounding such schemas into actual neural circuits. A formalism of symbolic neural dynamics is used here to model them as virtual circuits. Successive circuits linking sensory inputs with perceptual responses implement a differentiation of action schemas through *assimilation* i.e., the insertion of new sensations, followed by *accommodation*, i.e., the incorporation of actions. Simulations relying on a model of visual attention reproduce behaviors up to substage 5 of the sensory-motor stage. The transition to substage 6 i.e., when a child starts mastering invisible object displacements, requires additional functionalities that characterize mental representations as opposed to mere action schemas, and thus represent a shift from perceptual to representational responses. This process eventually achieves *objectification* i.e., the emergence of objects as autonomous and permanent entities.

Keywords: cognitive development, action schema, mental representation, assimilation, accommodation, objectification.

Background

According to Piaget's theory, cognitive development starts with a sensory-motor stage, which itself extends over 6 substages. At the beginning, infant behaviors are driven by action schemas. The transition between substages 5 and 6 stands out as the time a child begins acting on the basis of representations in contrast with perceptual responding. In Piaget's terminology, *assimilation* (i.e., the insertion of new sensations) is followed by *accommodation* (i.e., the incorporation of new actions) through "*differentiation of an existing schema*". This evolutive process achieves *objectification* i.e., the emergence of objects as autonomous and permanent entities. His postulate reads as follows: "*The criterion of this objectification, hence of this rupture in continuity between things perceived and the elementary sensory-motor schemata, is the advent of the behaviors patterns related to absent pictures: search for the vanished object, belief in its permanence, evocation, etc*." (Piaget 1937).

Goals

This raises the issues of how these schemas are acquired, put to work, and eventually give rise to mental representations. Cognitive neuroscience, which aims at linking cognition to brain processes, hasn't succeeded yet in grounding action schemas into actual neural circuits. A new formalism of symbolic neural dynamics is used here to model them as virtual circuits linking sensory inputs with perceptual responses. These circuits are used in turn to simulate successive steps of the sensory-motor stage. As postulated by Piaget, it will be shown how circuit transformations implement a progressive differentiation of action schemas through assimilation followed by accommodation, which at the end achieve *objectification.*

Methodology

It is out the scope of this paper to review the various methodological approaches that have been proposed towards the goal of linking cognition to brain processes (see e.g., Westerman et al 2006; Ashby & Helie 2011, Kriegeskorte & Douglas 2018). Briefly, it is customary to distinguish these approaches according to the formalism they rely e.g.:

- differential equations , which can be used to model dynamical systems either at the level of individual neurons (Hogkin & Huxley 1952) or at a higher level related to behavioral processes (Thelen & Smith 1994; van Gelder 1998)
- artificial neural networks, a example of connectionist models, which are used mainly to model learning processes (Hopfield 1982; Rumelhart & McClelland 1986)
- Bayesian inference rules i.e., an example of statistical models applied to predictive and/or causal processes (Knill & Pouget 2004; Doya 2007).

These various approaches can be further categorized as being either top down or bottom up. Critics have pointed out that unidirectional approaches cannot succeed in projecting cognition into actual brain processes (Carandini 2012; Love 2015; Cooper & Peebles 2015)**.** An intermediate, bidirectional level is needed, they argue, to fill the gap between brain measurements and observed behaviors or, in other words, between brain circuits and cognitive processes. Cognitive processes being executed in connection structures that link sensory circuits with motor responses, this binding requires "*a mechanism that shows how the information (synchrony of activation in this case) can be used by the brain*" (van der Velde et al 2015). This is the approach that will be taken here.

The neural processes that ground cognition being still mostly unknown, this approach cannot rely on, nor provide evidence about the links between actual neural circuits and behaviors. It can be used however to *construct brain structures that might be associated with cognitive processes*. This has been done to simulate possible mechanisms for the emergence of animal awareness (Bonzon 2019) and deterministic behaviors (Bonzon 2020).

Formalism

Towards the end of implementing a ''*middle-out*'' approach identifying plausible structures linking biology and cognition, a new formalism based on symbolic neural dynamics has been proposed (Bonzon 2017). Given under the form of a *virtual machine*, it offers an interface situated at a meso-scale level between behaviors and abstracted micro-circuits implementing synaptic plasticity.

Generally speaking, a virtual machine is a software construction having its own language $\boldsymbol{L}$ that emulates the execution of a program written in another higher level language $\boldsymbol{S}$, thus allowing for interfacing two domains. A classical example is given by the Java machine, where the languages $\boldsymbol{L}$ and $\boldsymbol{S}$ correspond respectively to Java *byte code* obtained from the compilation of Java *source code.* The virtual machine that we shall consider allows for interpreting *logical implications* $\boldsymbol{l} \in \boldsymbol{L}$ representing neural data, which themselves are compiled from symbolic expressions $\boldsymbol{s} \in \boldsymbol{S}$ representing *behavioral data.* On the other way around, symbolic expressions $\boldsymbol{s} \in \boldsymbol{S}$ representing *virtual circuits* for driving behaviors are first compiled into logical implications $\boldsymbol{l} \in \boldsymbol{L}$, which are used in turn to deduce *virtual machine instructions* i.e., the model's grounding.

In a first approximation, this machine does function as a non deterministic learning automaton that repeats a *sense-react* cycle of embodied cognition. In this particular embodiment, brain processes are first abstracted through virtual *microcircuits* implementing synaptic plasticity. Sets of microcircuits can be then assembled into *meso-scale* virtual circuits representing neural assemblies (Huyck & Passmore 2013) linking perceptions and actions. In order to represent *neural assemblies* that participate in overlapping functional entities, sets of neurons are modeled as concurrent communicating *threads* (see the *Supplementary information* section for details).

Tools

This virtual machine has been defined, and thus at the same time implemented, by a logic program of a few hundred lines that can be run on a PC equipped with a Prolog compiler. A *graphical representation* of circuits, which stands in one to one correspondence with their defining symbolic expressions, can be used to define models of virtual cognitive brain structures and processes.

Example: a simple case of learning through operant conditioning

In order to implement a simple form of learning through *operant conditioning*, let us consider

- a **`watch(I)`** thread that drives the learning process, where **`I`** is a sensory input
- a **`spot(I)`** thread discriminating perceptions through an *excite* or an *inhibit* stimulus
- two effector threads **`accept(I)`** and **`reject(I)`** defining output responses.

The following graphical conventions apply to represent virtual circuits:

- each named node e.g., **`watch(I)`**, **`accept(I)`**, etc., stands for a thread
- synaptic connection between neural assemblies are represented by the constructions

```
-*>=>-   and   -*>=>-
/|\              /|\
STP            LTP/LTD
```

where **`STP`** and **`LTD/LTP`** stand respectively for short term *potentiation* and long term *potentiation* or *depression* thread processes activating the *disinhibition* or *inhibition* of neural assemblies; inhibition strength is measured by a negative or null *synaptic weight* between the connected threads (see the *Supplementary information* section for details).

Let us then consider the virtual circuit in Fig. 1.

```
           ---*->=>-accept(I)
          |  /|\
          |  LTP
          |   |
          |   +-------------------------------
          |   |                               |
          |  LTD                              |
          |  \|/             |excite(accept(I))--
watch(I)-+---*->=>-spot(I)|
          |  /|\             |inhibit(accept(I))-
          |  LTD                              |
          |   |                               |
          |  +--------------------------------
          |   |
          |  LTP
          |  \|/
           ---*->=>-reject(I)
```

- at the start, the pathway from **`watch`** to **`spot`** is open, and pathways to **`accept`** and **`reject`** are closed
- thread **`spot`** discriminates inputs through positive and negative stimuli, and thus allows for diverging paths
- **`LTP`** threads open the path to either **`accept`** or **`reject`**, and **`LTD`** threads close the path to **`spot`**

Figure 1 Virtual circuit implementing a simple case of operant conditioning

This circuit matches a fundamental principle in circuit neuroscience according to which, as a result of synaptic plasticity (expressed here through **`LTP/LTD`** threads), *inhibition* in neuronal networks during baseline conditions allows in turn for *disinhibition* and constitutes a key mechanism for learning (Letzkus et al 2015; Zagha et al 2015). As a result, this circuit learns a deterministic behavior driven by two neural populations competing for an output response.

Results

The formalism and tools presented above are used to model the acquisition of object permanence. A simple reflex model first prompts the grasping of an object, as typically encountered in sensory-motor substages 1 to 2. Implementing eye saccades allows then for the visual tracking of a moving object (substage 3). Two different inhibitory control modes reproduce in turn the initial *A not B* error and the later correct retrieval of an object that gets hidden in successive locations (substage 4). The incorporation of a learning process relying on a location memory leads to the mastering of visible displacements. Finally, accommodating a perception by enabling its retained image to activate a search achieves the critical transition between substages 5 to 6 when a child faces invisible displacements.

These successive models are defined below by their corresponding circuits, and illustrated through the execution traces of actual simulation runs.

Modeling reflexes (sensory-motor substage 1-2)

The observations related to the first and second sensory-motor substages have been reported in (Piaget 1936). These essentially consist in describing reflex behaviors that are driven by visual attention and culminate in coordinated prehension: the grasping of objects becomes "*systematic when the object and the hand are perceived in the same visual field"*. In other terms, following a reciprocal assimilation, "*all that is to be seen is also to be grasped and all that is to be grasped is also to be seen"*.

The work of Wible et al. (2020) offers a model of visual attention that simulates behavioral and neural correlates as the product of attractor states in a dynamical system. In contrast, the model proposed here offers an account relying on abstractions that fit into the framework of a virtual machine. According to general psychology principles, it distinguishes two intervened steps:

- first *sensation* i.e., the capture of visual data through *sensors*
- then *perception* i.e., the interpretation of these data through virtual circuits linked to *effectors*.

Visual input data include an object *image* and its *position* in space. The capture of an object's image by the human retina results from well defined multilayered neural processes. As it has been demonstrated in rodent animals (O'Keefe & Dostrosky 1971; Moser & Moser 2008), the capture of position data is achieved via multiple receptive fields i.e., *place*, *head direction*, *grid* and *border* cells. The subsequent perception associating these two data results from yet mostly unknown higher level circuits and mechanisms (Lewis et al 2019; Bicanski & Burgess 2019; Anselmi et al 2020). To reduce both steps into tractable abstractions, our models rely on two simplifying hypotheses:

- space will be restricted and defined as a *one dimensional* axis, with visual sensory inputs defined as **`P(X)`**, where **`P`** and **`X`** stand respectively for the stored image of an object and its position on the space axis, which together constitute a *numerical iden*tity i.e., a prerequisite for object permanence (Moore & Meltzoff 2004)
- neural assemblies processing these inputs will be represented by threads activated through short term *potentiation*.

On this basis, a grasping reflex can be driven by the virtual circuit given in Fig. 2. In this circuit, two sensor threads **`sense(view(P(X)))`** and **`sense(hand(X))`** converge to signal that an object **`P`** and a hand are perceived in the same visual field **`X`**. As a result, the effector **`grasp(P(X))`** gets activated through a short term potentiation. In coordination with this visual drive, a grasping reflex involves other multi-modal perceptions e.g., for controlling motor actions (see e.g., Thelen et al 2001; Bonzon 2020). In the developments that follow, it is assumed that the circuit in Fig. 2 will automatically fire after a subject's required motor actions.

```
sense(view(P(X)))-*>=>-grasp(P(X))
                  /|\
                  STP
                   |
sense(hand(X))----
```

- a short term potentiation from **sense(hand(X))** opens the path from **sense(view(P(X)))** to **grasp**

Figure 2 Virtual circuit implementing the grasping of an object

Modeling visual object tracking (sensory-motor substage 3)

Among others explorations, infant early experiences with the world follow from their visual attention being caught by moving objects. As noted by Piaget (1937) in his description of the third sensory-motor substage, the child "*anticipates the perception of successive positions of the moving object* ". The tracking of moving objects results from eye saccades i.e., rapid target-driven eye reflex movements. These reflexes are driven by expected upcoming data anticipating the object's next position, and rely on pattern recognition from preceding inputs to discriminate inputs (Bicanski & Burgess 2019). In the simple case of a single object tracking, this anticipation relies on the *focus* of attention (i.e., the position where the object is expected to next hit the eyes): when the actual sensation does not meet the expectation (i.e., if another object actually hit the eyes), visual attention gets suspended, and a default action is taken. This represents an elementary case of an accommodation, whereby "*a rupture in continuity"* produces diverging paths.

As an example, let us consider a simple simulation scenario that reproduces a characteristic behavior that can be observed in the third sensory-motor substage, defined as follows:

- a toy is seen moving (e.g., carried or rolling) along a one dimensional axis
- if it stands still (e.g., is dropped or stops) in the sight of the observer, he grasps it
- if it disappears behind/under a screen/object, the observer looks at the occluding item.

Two successive eye saccades are sketched in Fig. 3 together with the corresponding sensory input vectors.

```
○⇨        ○
-----------------------------> x
0         1         2
sensor(move(toy(0))),sensor(see(toy(1)))        toy moving from position 0 to position 1


          ○⇨        ■
-----------------------------> x
0         1         2
sensor(move(toy(1))),sensor(see(screen(2)))     toy disappearing behind a screen
```

Figure 3 Successive eye saccades tracking a moving object

As a result of discriminating between inputs to eye saccades, a forward propagation of *excite/ inhibit* stimuli produces divergent circuit paths leading to anticipated or default output responses. This allows for constructing the virtual circuit implementing the visual tracking of a moving object given in Fig. 4.

```
sense(stop(P(X+1)))-----------------------------------------*>=>-grasp(P(X+1))
                                                           /|\
                                                           STP
                                                            |
sense(move(P(X)))-set(focus(P))------------*>=>-track(P(X+1))-
                                          /|\
                                          STP
                                           |
                                  |excite--
sense(see(Q(X+1)))-check(focus(Q))|
                                  |inhibit-
                                           |
                                          STP
                                          \|/
sense(move(P(X)))-set(focus(P))------------*>=>-look(Q(X+1))
```

- two parallel sensor threads first process the input **`move(P(X))`** and set the current focus of attention to **`P`**
- these two threads then wait for a sensor thread to process the input **`see(Q(X+1))`**
- checking the current focus of attention against the previous one produces either an **`excite`** or **`inhibit`** stimulus, leading in turn to apply a short term potentiation to one of two threads **`track(P(X+1))`** and **`look(Q(X+1))`**.
- after the firing of a thread **`sense(stop(P(X+1)))`** signaling that the toy stands still in the observer's visual field, a **`grasp(P(X+1))`** reflex gets activated by a potentiation from **`track(P(X+1))`**

Figure 4 Virtual circuit implementing the visual tracking of a moving object

This virtual circuit implements a first example of *assimilation* (i.e., in this case of the inputs produced by the tracking of a moving object), followed by an *accommodation* discriminating between a continued tracking and a default action (in this case looking at the occluding screen). This is illustrated in the following execution trace from an actual simulation run:

```
○⇨         ○
---------------> x
0         1
```

```
sensor(move(toy(0))),sensor(see(toy(1)))
 track(toy(1))                                                    track toy at 1
```

```
                     ⇧
          ○⇨         ○
-------------------------> x
0         1         2
```

```
sensor(move(toy(1))),sensor(see(toy(2)))
 track(toy(2))                                                    track toy at 2
sensor(stop(toy(2)))
 grasp(toy(2))                                                    grasp toy
```

```
                               ⇧
                    ○⇨         ○
-----------------------------------> x
0         1         2         3
```

```
sensor(move(toy(2))),sensor(see(toy(3)))
 track(toy(3))                                                    track toy at 3
sensor(stop(toy(3)))
 grasp(toy(3))                                                    grasp toy
```

```
                              ○⇨
---------------------------------------------> x
0         1         2         3         4
```

```
sensor(move(toy(3))),sensor(see(screen(4)))
 look(screen(4))                                                  look at screen
```

Modeling visual object tracking and searching (sensory-motor substage 4)

The next substage is marked by a child's ability to search for an object outside of his visual field e.g., behind a screen. At the beginning, the child does not take into account successive object displacements i.e., for him "*the place where the object was found for the first time remains the place where it will be found*", leading to the so-called *A not B error*. Piaget proposed a mix of possible explanations for this phenomenon, including a lack of ability to recall the sequence of displacements, to correctly take into account their order, and to separate objects from their context. This has been summarized as resulting from the persistent association binding an object with the infant's immediate action (Müller et al 2001), or as reflecting the sustained visual attention that accompanies a first reach (Ruffman 2001). In terms of neural processes, this could result from a failure to inhibit a previous response (Diamond 2001) i.e., in other terms to bind successive related sensory inputs and actions. After a while, a correct sequential tracking is steadily observed.

A virtual circuit implementing the tracking and searching of a moving object that extends the circuit of Fig. 4 is given in Fig. 5. This circuit illustrates another example of assimilation and accommodation. The sensation produced by a suspended attention, represented by the sensory input from **`sense(halt(Q(X+1)))`**, is followed by a new accommodation i.e., a **`search`** that gets activated by long time potentiation **`LTP`**. Two different **`LTP`** models reflecting a form of *weak* vs. *strong* form of synaptic plasticity (i.e., a brain maturation that allows at the end for binding successive related sensory inputs and actions) are used to reproduce in turn an A not B error and a correct sequential tracking (see the *Supplementary information* section for details). Furthermore, the sensation produced by uncovering an object is assimilated by another sensory input from **`sense(view(P(X+1))`** that produces a grasping reflex.

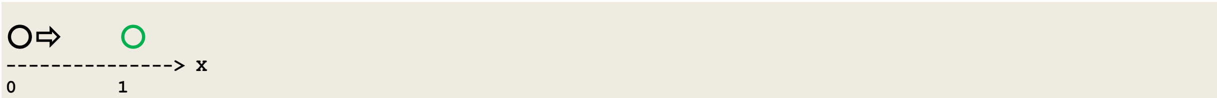
```
sense(stop(P(X+1)))--------------------------------------------*>=>-grasp(P(X+1))
                                                              /|\
                                                              STP
                                                               |
sense(move(P(X)))-set(focus(P))-------------*>=>-track(P(X+1))-
                                           /|\
                                           STP
                                            |
                                  |excite--
sense(see(Q(X+1)))-check(focus(Q))|
                                  |inhibit-
                                            |
                                           STP
                                           \|/
sense(move(P(X)))-set(focus(P))-------------*>=>-look(Q(X+1))--
                                                               |
                                                              LTP
                                                              \|/
sense(halt(Q(X+1)))--------------------------------------------*>=>-search(Q(X+1))-
                                                                                   |
                                                                                  LTP
                                                                                  \|/
sense(view(P(X+1)))----------------------------------------------------------------*>=>-grasp(P(X+1))
```

In addition to the previous circuit,

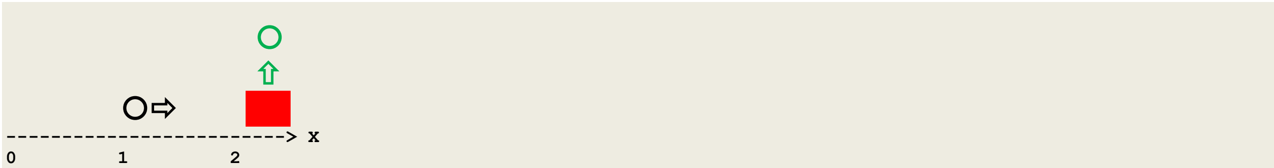
- the thread **`search(Q(X+1))`** gets driven by a **`sense(halt(Q(X+1)))`** thread whenever the toy disappears,
- two different models of long time potentiation **`LTP`** can be used to reproduce in turn an A not B error and a correct sequential tracking
- a thread **`sense(view(P(X+1)))`** signaling that object **`P`** has been uncovered at location **`X+1`** drives a grasping reflex
- this grasping reflex is activated by a potentiation from the **`search`** thread

Figure 5 Virtual circuit implementing the visual tracking and searching of a moving object.

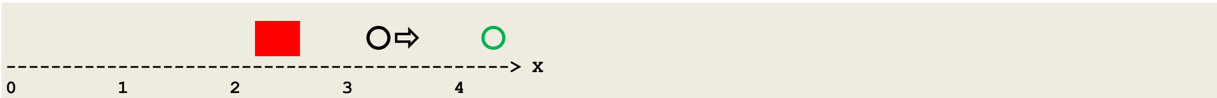
An execution trace of this circuit implemented with a *weak* **LTP** potentiation is given below:



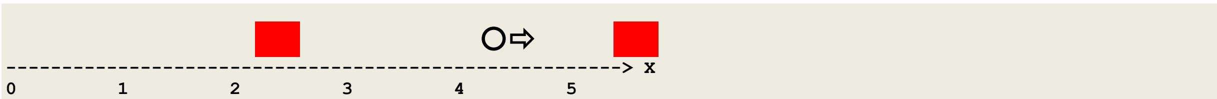
`sensor(move(toy(0))),sensor(see(toy(1)))`	
`track(toy(1))`	*track toy at 1*



`sensor(move(toy(1))),sensor(see(screen(2)))`	
`look(screen(2))`	*look at screen*
`sensor(halt(screen(2)))`	
`search(screen(2))`	*search screen*
`sensor(view(toy(2)))`	
`grasp(toy(2))`	*grasp toy*

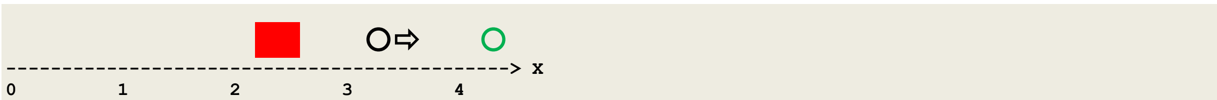


`sensor(move(toy(3))),sensor(see(toy(4))))`	
`track(toy(4))`	*track toy at 4*

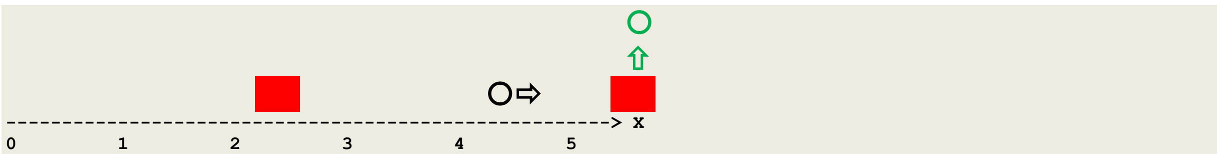


`sensor(move(toy(4))),sensor(see(screen(5)))`	
`look(screen(5))`	*look at 2nd screen*
`sensor(halt(screen(5)))`	
`search(screen(2))`	*search 1st screen*

The unfeasibility of binding a second pair of related inputs at the second screen forced a renewed search at the first screen, thus producing an A not B error. In contrast, the same circuit implemented with a *strong* long time potentiation allows for successive associations of related sensory inputs and actions and produces the following end execution trace, which reflects a correct second search:



`sensor(move(toy(3))),sensor(see(toy(4)))`	
`track(toy(4))`	*track toy at 4*



`sensor(move(toy(4))),sensor(see(screen(5)))`	
`look(screen(5))`	*look at2nd screen*
`sensor(halt(screen(5)))`	
`search(screen(5))`	*search 2nd screen*
`sensor(view(toy(5)))`	
`grasp(toy(5))`	*grasp toy*

Modeling a partially invisible displacement (sensory-motor substage 5)

The first acquisition of the next stage is to account for sequential displacements, i.e., to correct the A not B error. As discussed above, this is achieved in our framework by activating the **`search`** thread through a long time **`LTP`** potentiation implementing a *strong* from of synaptic plasticity. In order to study the dissociation of objects from their context (e.g., when an object's position is not directly perceived because of some invisible part along its way), Piaget (1937) devised a series of experiments: "*hiding an object not directly under a screen, but in box without a lid; box and object are made to disappear under a screen and the box brought out empty".* He then observed what he called an "*empirical or practical apprenticeship*" which, he argued, does not yet involve any image or representation of spatial relations.

Our developments follow closely observation 55 from (Piaget 1937). This observation was divided in three phases:

I. An object is put in a box while the infant watches; the box is then placed under a screen and turned down to leave the object hidden under the screen without the infant noticing it; the box is finally brought out empty. The infant then searches for the object in the box, eventually looks around, but doesn't search for the object under the screen
II. After a few repetition of this technique followed by the same negative result, the box is left under the screen with the object inside; the infant then immediately looks under the screen and grasps the object (NB. in the original description, the infant finds and grasps the box, *opens it*, and take the object out of it; these details will be ignored here for the sake of simplicity, especially since the box was not explicitly said to be closed)
III. Finally, the experiment protocol of phase I is resumed: this time, the infant first looks for the object in the box and not finding it then searches under the screen (NB this positive result is steadily observed only after a few experiments).

The outcome of phase III led Piaget to conclude that mastering partially invisible displacements (NB which are generally but oddly referred to as "visible displacements") could not occur through the awareness of some relation or image, but as a result of a "practical schema" acquired through some kind of learning.

These three stages can be implemented through a further differentiation of the previous schema that gives rise to the schema in Fig.6. In this circuit, the practical learning envisioned by Piaget is implemented as a simple case of *operant conditioning*, which involves a **`watch`** and a **`spot`** thread as defined in the virtual circuit of Fig.1. This behavior relies on the remembered position where the object was last seen (or equivalently, did disappear), represented in our formalism by a short term memory **`<look(Q(X+1))>`**. In addition, a **`view`** thread discriminates between desirable and undesirable items.

```
                                                                                   ---*>=>-grasp(I(X+1))
                                                                                  |  /|\
                                                                                  |  LTP
                                                                                  |   |
                                                                                  |   +-----------------------------
                                                                                  |   |                             |
                                                                                  |  LTD                            |
                                                                                  |  \|/                  |excite--
sense(stop(P(X+1)))--------------------------------------------*>=>-watch(P(X+1))-+---*>=>-spot(F(I)(X+1))|
                                                              /|\                 |  /|\                  |inhibit-
                                                              STP                 |  LTD                            |
                                                               |                  |   |                             |
sense(move(P(X)))-set(focus(P))-------------*>=>-track(P(X+1))-                   |   +-----------------------------
                                           /|\                                    |   |
                                           STP                                    |  LTP
                                            |                                     |  \|/
                                  |excite--                                       *---*>=>-search(_(_))-
sense(see(Q(X+1)))|check(focus(Q))|                                               |   |                 |
                                  |inhibit-                                       |   |                 |
                                            |                                     |   |                 |
                                           STP                                    |   |                 |
                                           \|/                                    |   |                 |
sense(move(P(X)))|set(focus(P))-------------*>=>-look(Q(X+1))-+-<look(Q(X+1))-----+   |                 |
                                                              |                   |   |                 |
                                                             LTP                  |   |                 |
                                                             \|/                  |   |                 |
sense(halt(Q(X+1)))-------------------------------------------*>=>----------------*---                  |
                                                                                                       LTP
                                                                                                       \|/
sense(view(I(X+1)))-------------------------------------------------------------------------------------*>=>-view(I(X+1))|excite-grasp(I(X+1))
```

In addition to the previous circuit,

- the **`stop`** thread activates a learning circuit representing a simple case of *operant conditioning* where **`I`** stands for the object contained in box `F`
- this sub-circuit is imbedded in the overall circuit such that the **`search`** is now driven by the **`halt`** thread as a result of learning
- in both case, this search relies on the memorized position where the object did disappear, represented here by **`<look(Q(X+1))>`**
- the object **`I`** found in the box, as captured by the sensory input **`sense(view(I(X+1)))`**, gets discriminated in the **`view`** thread.

Figure 6 Virtual circuit implementing a partially invisible displacement.

Simulation run of a partially invisible displacement

The successive stages of observation 55 are illustrated below through their simulation traces.

I.

```
|O|⇨  |O|

---------------> x
0          1
```

`sensor(move(box(toy)(0))),sensor(see(box(toy)(1)))`

`track(box(toy)(1))` *track box+toy at 1*

```
           |O|⇨
```

```
------------------------> x
0          1         2
```

`sensor(move(box(toy)(1))),sensor(see(screen(2)))`

`look(screen(2))` *look at screen*

```
                                       |_|⇨    |_|
-----------------------------------------------> x
0          1          2          3          4
```

`sensor(move(box([])(3))),sensor(see(box([])(4)))`

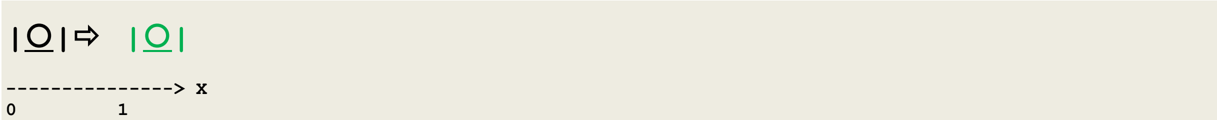
`track(box([])(4))` *track empty box at 4*

`sensor(stop(box([])(4)))`

`watch(box([])(4))` *watch empty box*

`spot(box([])(4))` *spot empty box*

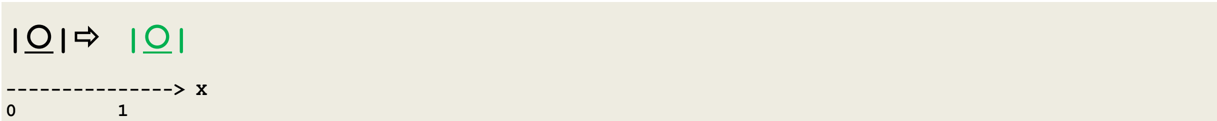
II.



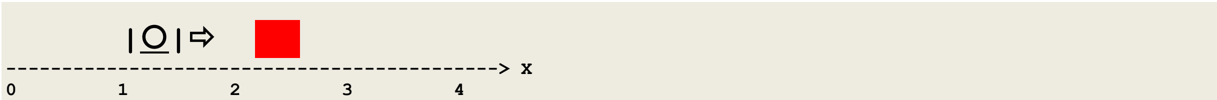

```
sensor(move(box(toy)(0))),sensor(see(box(toy)(1)))
```
`track(box(toy)(1))` — *track box+toy at 1*


```
sensor(move(box(toy)(1))),sensor(see(screen(2)))
```
`look(screen(2))` — *look at screen*
```
sensor(halt(screen(2)))
```
`search(screen(2))` — *search screen*
```
sensor(view(toy(2)))
```
`view(toy(2))` — *view toy*
`grasp(toy(2))` — *grasp toy*

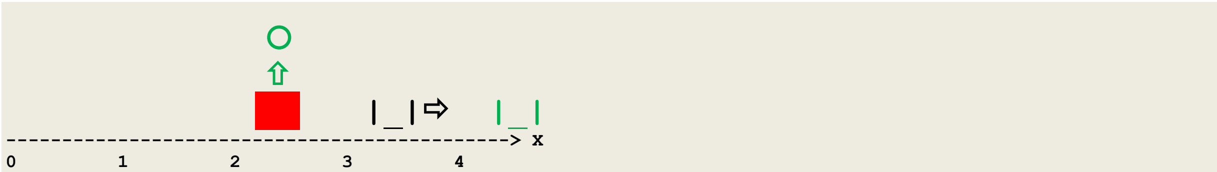
III.




```
sensor(move(box(toy)(0))),sensor(see(box(toy)(1)))
```
`track(box(toy)(1))` — *track box+toy at 1*




```
sensor(move(box(toy)(1))),sensor(see(screen(2)))
```
`look(screen(2))` — *look at screen*




```
sensor(move(box([])(3))),sensor(see(box([])(4)))
```
`track(box([])(4))` — *track empty box at 4*
```
sensor(stop(box([])(4)))
```
`watch(box([])(4))` — *watch empty box*
`spot(box([])(4))` — *spot empty box*
`search(screen(2))` — *search screen*
```
sensor(view(toy(2)))
```
`view(toy(2))` — *view toy*
`grasp(toy(2))` — *grasp toy*

Modeling invisible displacement (sensory-motor substage 6)

The transition between substages 5 and 6 i.e., when a child starts mastering invisible object displacements, demonstrates a shift from perceptual to representational responses i.e. , a capacity that can be invoked in the absence of a perceived reality (McCune 2001). This is summarized as being able to "keep an object in mind" when it is not in sight. This capacity builds up to the sequential tracking of objects that undergo successive invisible displacements. Our developments reproduce here Piaget's observation 64, translated however in a different setting involving a covered box instead of a closed hand. This observation is divided in three phases retaining their original numbering.

Ia. An object is put in a box and the box is covered by a lid while the infant watches; the box is then placed under a screen and emptied to leave the object hidden under the screen without the infant noticing it; the box is finally brought out empty. The infant searches for the object in the empty box, and then goes on searching for it under the screen

Ib. The same experiment is repeated, with the covered box being passed and emptied in a different screen; the infant immediately searches this second screen.

II. The experiment protocol of phase I is resumed, but this time the box passes under two successive screens before stopping; the infant looks for the object under the first screen, and not finding it searches the second screen.

These three phases can be implemented through an ongoing differentiation of the schema in Fig. 6 that ends up with the extended circuit in Fig. 7. At the start, watching the experimenter while he places a toy in a box (or equivalently takes it in his hand palms) and then covers the box (or closes his hands) produces a new sensation involving a *relation* between two objects. This gets assimilated by **`sense(open(F(I)(X)))`** and **`sense(close(F(_)(X)))`** threads, where **`F`** and **`I`** stand respectively for the box and object, and accommodated by the **`image(F(I)(X))`** thread that creates an internal representation **`{image(F(I)(X))}`** implemented via an **`LTS`** long term storage process. In our formalism, the internal representation **`{P}`** of a thread **`P`** extends the mechanism of long term potentiation and allows for an **`LTR(P,Q,R)`** retrieval process to be fired by **`Q`** in order to relate **`Q`** and **`R`**, thus defining the basic mechanisms of an associative memory (see the *Supplementary information* section for details). In the present context, this retrieval process drives the **`watch`** thread (which stands here for **`Q`**) to activate the **`open`** thread (which stands for **`R`**) and thus trigger the opening of the box after it stops.

After opening the box, **`sense(spot(F(I)(X)))`** drives **`spot`** thread to either grasp a desirable item or activate a new search. According to Piaget's observation, infants who at first do open a box and find it empty do not open it again in subsequent trials. This is achieved here through a long term blocking process **`LTB`**. Altogether, this new accommodation enlarges the previous operant conditioning learning process by allowing it to be driven by an image evocation.

Finally, in order to take into account successive invisible displacements, an evocation from the **`{image(F(I)(X))}`** memory drives an **`LTR`** retrieval process that activates a renewed **`search`** via the discriminating **`view`** thread. In order to recall the sequence of displacements and take into account their order, the memory **`<look(Q(X+1))>`** is implemented as a classical *first-in-first-out* (FIFO) data structure.

```
                                                                                                                           |excite-grasp(I(X+1))
sense(spot(F(I)(X+1)))---------------------------------------------------------------------------------*>=>-spot(F(I)(X+1))|
                                                                                                      /|\                  |inhibit-
                                                                                                      LTP                           |
sense(open(F(I)(X)))--*>=>-image(F(I)(X))-+---*-{image(F(I)(X))}-----------------*---                  |                            |
                     /|\                  |  /|\                                 |   |                 |                            |
                     STP                  |  LTS                                 |  LTR                |                            |
                      |                   |   |                                  |  \|/                |                            |
sense(close(F(_)(X)))-                     ---                                   +---*>=>-open(P(X+1))-                             |
                                                                                 |  /|\                                             |
                                                                                 |  LTB                                             |
                                                                                 |   |                                              |
sense(stop(P(X+1)))-------------------------------------------*>=>-watch(P(X+1))-+   +----------------------------------------------
                                                             /|\                 |   |
                                                             STP                 |  STP
                                                              |                  |  \|/
sense(move(P(X)))-set(focus(P))------------*>=>-track(P(X+1))-                   *---*>=>-search(_(_))-
                                          /|\                                    |   |                 |
                                          STP                                    |   |                 |
                                           |                                     |   |                 |
                                  |excite--                                      |   |                 |
sense(see(Q(X+1)))-check(focus(Q))|                                              |   |                 |
                                  |inhibit-                                      |   |                 |
                                           |                                     |   |                 |
                                          STP                                    |   |                 |
                                          \|/                                    |   |                 |
sense(move(P(X)))-set(focus(P))------------*>=>-look(Q(X+1))-+-<look(Q(X+1))>----+   |                 |
                                                             |                   |   |                 |
                                                            LTP                  |   |                 |
                                                            \|/                  |   |                 |
sense(halt(Q(X+1)))------------------------------------------*>=>----------------*---+                 |
                                                                                 |   |                 |
                                                                                 |   |                 |
{image(F(I)(X))}------------------------------*---                               |   |                 |
                                              |   |                              |   |                 |
                                              |  LTR                             |   |                 |
                                              |  \|/                             |   |                 |
                                     |inhibit-+---*------------------------------*---                  |
sense(view(P(X+1)))-*>=>-view(P(X+1))|                                                                 |
                   /|\               |excite-grasp(P(X+1))                                             |
                   LTP                                                                                 |
                    |                                                                                  |
                     ----------------------------------------------------------------------------------
```

In addition to the previous circuit,

- the relation between objects **F** and **I** is assimilated by **sense(open(F(I)(X)))** and **sense(close(F(_)(X)))**
- this sensation gets accommodated by the creation of an internal representation **{image(F(I)(X)))}**
- this image's evocation triggers a retrieval process that leads the **watch** thread to activate an **open** thread driving the opening of the closed box after it stopped
- a **LTB** process blocks subsequent box openings
- after the box is opened, the **sense(spot(F(I)(X)))** thread drives a **spot** thread that leads to either grasp a wanted item or activate a new search
- a retrieval from the **{image(F(I)(X))}** internal representation drives the **view** thread to activate a renewed **search**

Figure 7 Virtual circuit implementing invisible displacements

Simulation run of invisible displacements

The successive stages of observation 64 are illustrated below through their simulation traces.

Ia.

```
|O|
|O|
|O|
---------------> x
0              1
```

sensor(open(box(toy)(0))),sensor(close(box(_)(0)))

{image(box(toy)(0))} *toy in closed box image*

```
|O|⇨   |O|
---------------> x
0              1
```

sensor(move(box(_)(0))),sensor(see(box(_)(1)))

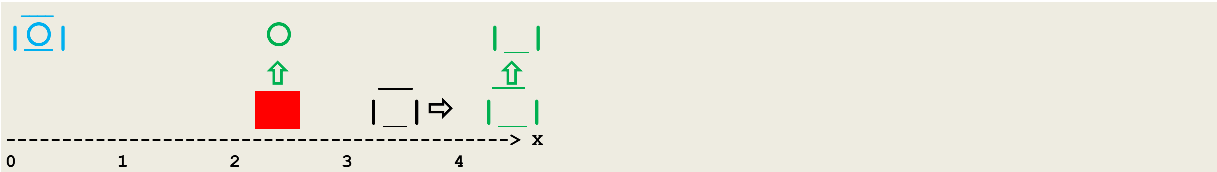
track(box(_)(1)) *track closed box at 1*

```
               |O|⇨
```

```
------------------------> x
0              1            2
```

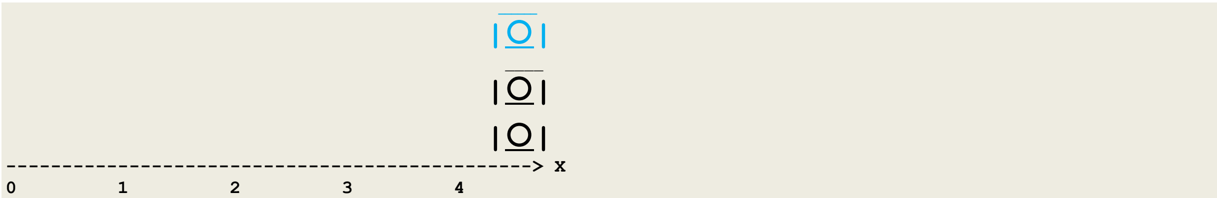
sensor(move(box(_)(1))),sensor(see(screen(2)))

look(screen(2)) *look at screen*

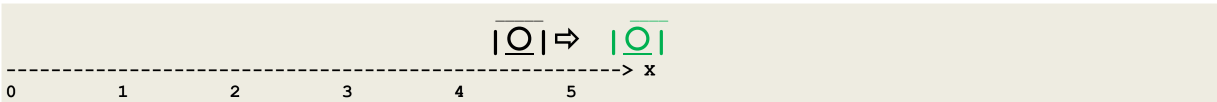


sensor(move(box(_)(3))),sensor(see(box(_)(4)))	
track(box(_)(4))	*track closed box at 4*
sensor(stop(box(_)(4)))	
watch(box(_)(4))	*watch closed box*
{image(box(toy)(0))}	*evoke mental image*
open(box(_)(4))	*open box*
sensor(spot(box([])(4)))	
spot(box([])(4))	*spot empty box*
search(screen(2))	
sensor(view(toy(2)))	*search screen*
view(toy(2))	*view toy*
grasp(toy(2))	*grasp toy*

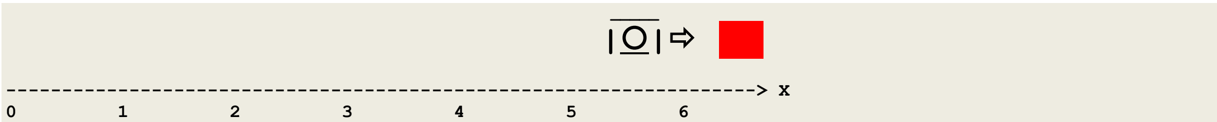
Ib.



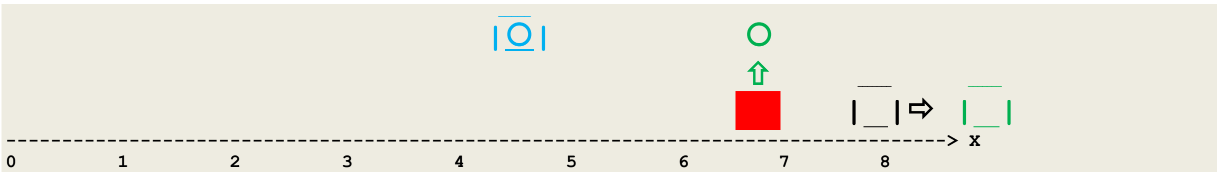
sensor(open(box(toy)(4))),sensor(close(box(_)(4)))	
{image(box(toy)(4))}	*toy in closed box image*



sensor(move(box(_)(4))),sensor(see(box(_)(5)))	
track(box(_)(5))	*track closed box a 5*

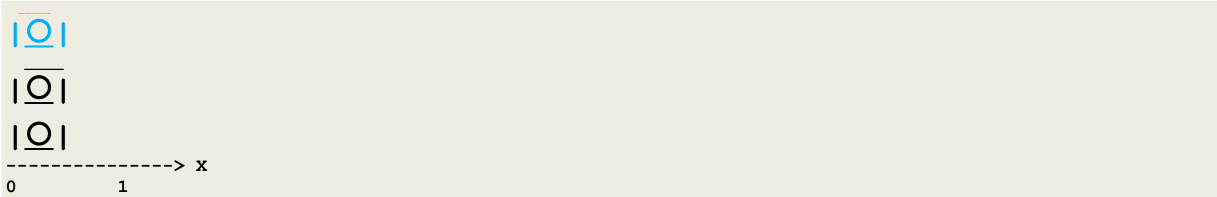


sensor(move(box(_)(5))),sensor(see(screen(6)))	
look(screen(6))	*look at screen*



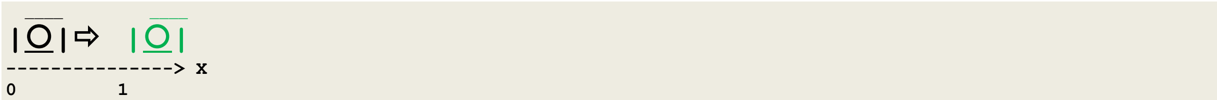
sensor(move(box(_)(7))),sensor(see(box(_)(8)))	
track(box(_)(8))	*track closed box at 8*
sensor(stop(box(_)(8)))	
watch(box(_)(8))	*watch closed box*
{image(box(toy)(4))}	*evoke mental image*
search(screen(6))	*search screen*
sensor(view(toy(6)))	
view(toy(6))	*view toy*
grasp(toy(6))	*grasp toy*

II.



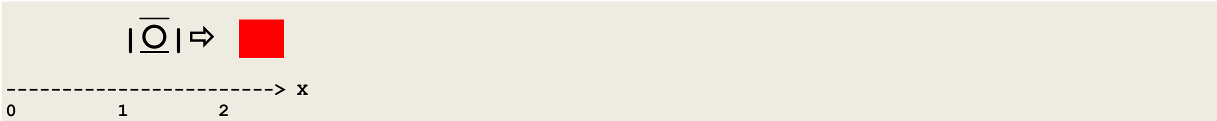
sensor(open(box(toy)(0))),sensor(close(box(_)(0)))

{image(box(toy)(0))} — *toy in closed box image*



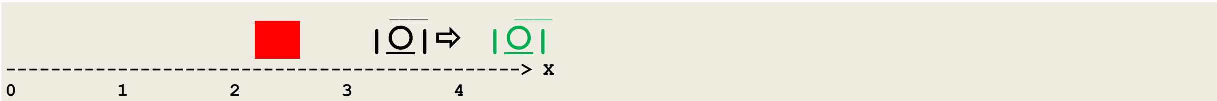
sensor(move(box(_)(0))),sensor(see(box(_)(1)))

track(box(_)(1)) — *track closed box at 1*



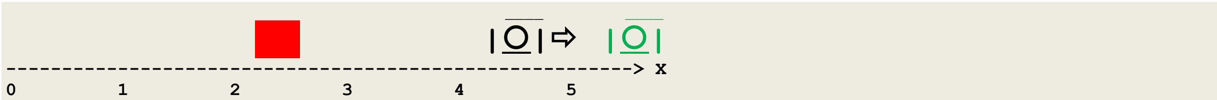
sensor(move(box(_)(1))),sensor(see(screen(2)))

look(screen(2)) — *look at 1st screen*



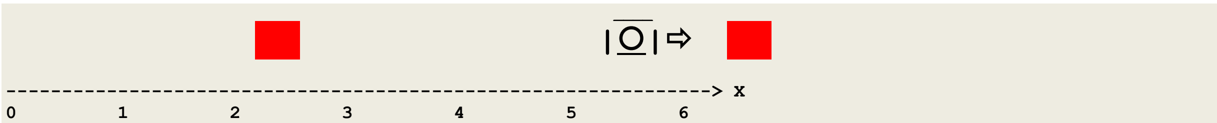
sensor(move(box(_)(3))),sensor(see(box(_)(4)))

track(box(_)(4)) — *track closed box at 4*



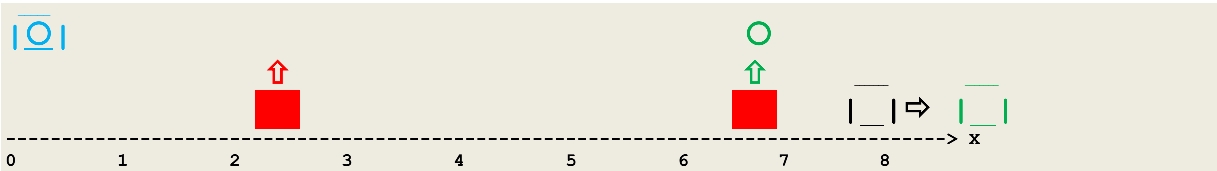
sensor(move(box(_)(4))),sensor(see(box(_)(5)))

track(box(_)(5)) — *track closed box at 5*



sensor(move(box(_)(5))),sensor(see(screen(6)))

look(screen(6)) — *look at 2nd screen*



sensor(move(box(_)(7))),sensor(see(box(_)(8)))

track(box(_)(8)) — *track closed box at 8*

sensor(stop(box(_)(8)))

watch(box(_)(8)) — *watch closed box*

{image(box(toy)(0))} — *evoke mental image*

search(screen(2)) — *search 1st screen*

sensor(view([](2)))

view([](2)) — *view void screen*

{image(box(toy)(0))} — *evoke mental image*

search(screen(6)) — *search 2nd screen*

sensor(view(toy(6)))

view(toy(6)) — *view toy*

grasp(toy(6)) — *grasp toy*

Discussion

This discussion extends in three directions i.e., the relevance of this work to Piaget's cognitive development theory, a comparison with previous work, and perspectives towards grounding cognition into actual neural circuits.

Relevance to Piaget's theory

The models presented in this study can be summarized as follows:

1) modeling object tracking and searching (sensory-motor substage 1 to 4)
Visual attention switches from a tracking thread to a looking thread that triggers a search thread whenever an object disappears.

2) modeling partially invisible displacements (sensory-motor substage 5)
In addition, a memory and learning process activates a search towards the location where the object did disappear (or equivalently was last seen).

3) modeling invisible displacements (sensory-motor substage 6)
In addition, the retained symbolic image of a relation perceived between objects drives a search activated by this image's evocation.

Altogether, these successive circuit transformations, which implement a differentiation of action schemas through assimilation followed by accommodation, eventually achieve *objectification.* This then raises the question about possible links between these models and the psychological processes they aim to represent.

Piaget's early theoretical developments about mental representations can be found in (Piaget 1936, p.242). After discussing an experiment (obs. 129) in which a child looks under a shawl for his shoe in order to strike it, he argues as follows:

"Hence the accommodation of this stage is more refined than that of the schemata hitherto under study, since the mobile schema applies to relations between external things and no longer only to things in their mere connection with the activity itself".

He then goes on asking the question

"Does this accommodation involve representation?

to which he first proposes the following answer:

If one understands representation to mean the capacity to confer upon things a meaning before the action which this meaning permits, it is apparent that representation exists".

A few lines below however, he notes:

"On the other hand, if one understands representation to mean the capacity to evoke by a sign or a symbolic image an absent object or an action not yet carried out, and then nothing yet warrants asserting its existence.

Indeed, as he finally concludes*:*

"In order that he looks for his shoe it is not necessary that the child picture it to himself".

According to this argumentation, a criterion for the existence of a mental representation is the capacity of a symbolic image to evoke an action not yet carried out. This capacity is implemented in our virtual neurological framework through the additional functionalities required for tracking invisible displacements recalled above i.e.:

3a) assimilating a relation between two things through a symbolic image
3b) accommodating this relation by enabling its retained image to drive an action.

It can be concluded that these additional functionalities characterize mental representations as opposed to mere action schemas, and thus implement a shift from perceptual to representational responses.

Previous similar work

Whereas a large body of experimental studies and confrontation of ideas have addressed Piaget's theory, there hasn't been much formal modeling done towards either its validation or falsification. A notable exception addressing the early part of the sensory-motor stage is constituted by the work of Thelen, Schöner and colleagues, who did propose two dynamic field models of *habituation* and *perseveration* (Thelen et al 2001; Schöner & Thelen 2006), which have been recently unified (Aerdker et al 2020). These models implement activation and inhibition of coupled processes that drive "looking and looking away", and thus provide a complementary explanations for the *A not B* error. The connectionist model proposed by Munakata (1998) similarly implements a competition between an *active* and a *latent* memory.

A prospective view

The concept of mental representation still eludes a constructive definition that could be used to conduct simulations linking cognition to actual neural circuits. The failure so far of the Human Brain Project (Markram et al 2015) to connect its simulated neocortical micro-circuitry with a meaningful behavior is highly symptomatic of this impasse. Results in this field typically produce atlases of neuro-images and recordings that allow for multivariate predictive models (see e.g., Martin 2016; Kragel & al 2018). The agenda of cognitive neuroscience science definitively requires a ''*middle-out*'' approach identifying plausible structures linking biology and cognition (Mulder et al. 2014; Frank 2015). Studies pursuing this goal should eventually ground their models of perception into the actual capture of visual data via multiple receptive fields (Lewis et al 2019; Bicanski & Burgess 2019; Anselmi et al 2020). Completing the already available experimental evidence about "an innate preconfigured spatial representation system" (Langston et al 2010 and "the presence of three neuronal representations of space before extensive experience" (Wills et al 2010) should then allow for elucidating the presumptive existence vs requirement of innate vs acquired mental representations relating an object to space.

Conclusion

The present work proposes a new formalism for modeling brain structures that might be associated with evolutive cognitive processes. Implemented in a simulation framework, this formalism was used to demonstrate how action schemas can be enriched with mental representations in order to eventually achieve objectification. Except for the *A not B error*, which in our simulations gets corrected through a brain maturation and thus does not recur, these simulated structures enjoy a functional continuity (in the sense that at any point in their evolution, they still support previous behaviors) as postulated by Piaget's theory (Piaget 1945).

Being limited to a one-dimensional space and implemented as a virtual machine operating at a meso-scale level, these simulations constitute only a first step towards the goal of unveiling detailed causality structures in a real brain. Projecting these abstractions into actual neural processes constitutes the next challenge. As argued by Frank and Badre (2015), attempts to link latent cognitive processes with the neural mechanisms that generate them "have, and will continue to be, instrumental in guiding neuroscientific discoveries".

Supplementary information

This section provides technical information about the formalism defined in (Bonzon 2017). Although its original specifications have been enlarged to accommodate the developments introduced in the present work, its basic principles remain the same.

A new approach to modeling brain functionalities

In this new formalism, brain processes representing synaptic plasticity are abstracted through asynchronous communication protocols and implemented as virtual *microcircuits*. The basic units of these micro-circuits are constituted by *threads*, which correspond either to a single or to a cluster of connected neurons. Contrary to traditional neuron models in which incoming signals are summed in some integrated value, thread inputs can be processed individually, thus allowing for threads to maintain parallel asynchronous communications. Threads can be grouped into disjoint sets, or *fibers* to model neural assemblies, and discrete *weights* (e.g., integer numbers) can be attached to pairs of threads that communicate within the same fiber. A fiber containing at least one active thread constitutes a *stream.* Mesoscale *virtual circuits* linking perceptions and actions are built out of microcircuits. Circuits can be represented either graphically or by sets of *symbolic* expressions. These expressions can be compiled into *virtual code implications* that are used just in time to deduce instructions to be finally interpreted by a *virtual machine* performing contextual deductions.

Basic concepts

To introduce this formalism, let us consider a simple case of synaptic transmission between any two threads **`P`** and **`Q`**. This can be represented by the circuit fragment (or wiring diagram) contained in the simple stream given in Figure 1, where the symbol **`->=>-`** represents a synapse**.**

```
…-P->=>-Q-…
```

Figure 1. Circuit fragment implementing a synaptic transmission

This circuit fragment can be represented by two symbolic expressions involving a pair of **`send/receive`** processes as shown in Fig. 2.

```
thread(P,[…,send(Q)])
thread(Q,[receive(P),…])
```

Figure 2. Thread patterns for a synaptic transmission

In Fig. 2, the thread **`P`** (e.g., a sensor thread **`sense(us)`** with **`us`** representing an external stimulus, as in the example of Fig. 4) will fire in reaction to an external stimulus, with the **`send`** process corresponding to the signal, or spike train, carried by a pre-synaptic neuron's axon. In the thread **`Q`** (e.g., an effector thread **`motor(X)`**, where the variable **`X`** will be instantiated as the result of the stimulus), the **`receive`** process represents the possible reception of this signal by a post-synaptic neuron. The compilation of these expressions will give rise to the execution of virtual code instructions implementing the communication protocol given in Fig. 3.

```
P:        …
          send(Q)         activate Q if Q is not active and post a signal for Q

Q:        receive(P)      wait for a signal from P and proceed if weight(P,Q)>0
          …
```

Figure 3. Communication protocol for an asynchronous communication

This protocol corresponds to an *asynchronous* blocking communication subject to a threshold. It involves a predefined weight between the sender **`P`** and the receiver **`Q`** that can be either incremented or decremented. On one side, thread **`P`** fires thread **`Q`** if necessary and sends it a signal. On the other side, thread **`Q`** waits for the reception of a signal from thread **`P`** and proceeds only if the weight between **`P`** and **`Q`** stands above a given threshold. The overall process amounts to opening a temporary *pathway* between **`P`** and **`Q`** and allows for passing data by instantiating variable parameters (see example below).

Example

As a simple example, let us consider the classical conditioning of *aplysia californica* (Carew et al 1981). In this experiment, a light tactile conditioned stimulus **`cs`** elicits a weak defensive reflex, and a strong noxious unconditioned stimulus **`us`** produces a massive withdrawal reflex. After a few pairings of **`cs`** and **`us`** with **`cs`** slightly preceding **`us`**, **`cs`** alone triggers a significantly enhanced withdrawal reflex. The corresponding circuit is represented in Fig. 4. In this circuit, the symbol **/|** represents the modulation of a synaptic transmission, the sign * used in the upper branch indicates the conjunction of converging signals, and the sign + either the splitting of a diverging signal, as used in the lower branch, or a choice between converging signals, as used in the right branch instantiating the thread **`motor(X)`**, where **`X`** is a variable parameter to be instantiated into either **`cs`** or **`us`**.

```
sense(cs)-*->=>-
         /|\    |
         ltp    +-motor(X)
          |     |
sense(us)-+->=>-
```

Figure 4. A circuit implementing classical conditioning.

In Fig. 4, the thread **`ltp`** (standing for *long term potentiation*) acts as an interneuron reinforcing the pathway between **`sense(cs)`** and **`motor(X)`**. Classical conditioning then follows from the application of hebbian learning i.e., “neurons that fire together wire together”. Though it is admitted today that classical conditioning in *aplysia* is mediated by multiple neuronal mechanisms including a postsynaptic retroaction on a presynaptic site, the important issue is that this activity depends on the temporal pairing of the conditioned and unconditioned stimuli, which leads to implement the thread **`ltp`** as a *detector of coincidence* as done in the protocol given in Fig. 5.

A mechanism for simulating long term potentiation

The generic microcircuit abstracting the mechanism of long term potentiation is reproduced in Fig. 5 with its communication protocol. A long term depression **`ltd`** thread can be similarly implemented by decrementing weights.

As a further theoretical abstraction, this formalism allows to distinguish between a hypothetical *weak* and a *strong* synaptic plasticity reflecting a brain maturation (Bolton et al 2017). Whereas a *strong* synaptic plasticity allows for successive associations of related sensory inputs and actions, a *weak* plasticity allows for only one. This brain maturation In our implementation, this directly follows from the underlying logical programming framework that distinguishes between named variables such as **`X`**, which can create bindings, and anonymous variables denoted by the character “_ “, which cannot.

```
Q---*->=>-R
   /|\
   ltp
    |
P---+...
```

P:	**fire(ltp(Q,R))** …	*fire thread ltp(Q,R)*
ltp(Q,R):	**join(Q)** **increment(weight(Q,R))**	*wait for a signal from Q* *increment weight between Q and R*
Q:	**merge(ltp(Q,R))** **send(R)**	*post a signal for ltp(Q,R)* *fire thread R and post a signal for R*
R:	**receive(Q)**	*wait for a signal from Q and proceed if w eight(Q,R)>0*

In order to detect the coincidence of **P** and **Q**, **P** fires an **ltp** thread that calls on **join** to wait for a signal from **Q**. In parallel, **Q** calls on **merge** to post a signal for **ltp** and then executes a **send(R)** command to establish a link with **R**. After its synchronization with **Q**, **ltp** increments the weight between **Q** and **R**

.

Figure 5. Micro-circuit and communication protocol for *ltp*

An associative memory as the basis of mental representations

Turning now to mental representations, they extend the mechanism of long term potentiation by allowing for two threads **P** and **Q** attached to separate streams (and thus also possibly active at different times) to be associated in order to trigger a recall thread **R.** These two streams are linked together through a double communication protocol applied to a long term memory **ltm(P)** thread, this construct being depicted by the symbol **-{P}-** meaning that **P** is both stored and retrievable through the thread **ltm(P).** This new protocol involves two complementary *long term storage/retrieval* (**lts/ltr**) threads that allow respectively for the building of a storage trace and the later retrieval of a previously active thread. The corresponding microcircuit is given in Figure 6 together with its communication protocol.

As a distinctive difference from an **ltp(Q,R)** thread (which gets fired by **P** and waits for a signal from **Q** in order to relate **Q** and **R**), an **ltr(P,Q,R)** thread is fired by **Q** and waits for a path from **ltm(P)** in order to relate **Q** and **R**, thus defining the basic mechanisms of an associative memory.

```
         Q--+---*->=>-R
            |   /|\
            |   ltr(P,Q,R)
            |    |
P-+---*-{P}-*---
  |  /|\
  |  lts(P)
  |    |
  ---
```

`P:`	`fire(lts(P))`	*fire thread lts(P)*
`lts(P):`	`store(P)` `increment(weight(P,ltm(P)))`	*fire thread ltm(P)* *increment weight between P and ltm(P)*
`ltm(P):`	`feed(_)`	*if weight(P,ltm(P))>0 then open path*
`Q:`	`fire(ltr(P,Q,R))` `send(R)`	*fire thread ltr(P,Q,R)* *fire thread R and post a signal for R*
`ltr(P,Q,R):`	`retrieve(P)` `increment(weight(Q,R))`	*wait for an open path from ltm(P)* *increment weight between Q and R*
`R:`	`receive(Q)`	*wait for a signal from Q and weight(Q,R)>0*

Figure 6 Microcircuit and communication protocol for a long term associative memory.

Acknowledgment

The author acknowledges the comments by an anonymous referee of a previous work, which provided the incentive for the developments reported here.

References

Aerdker, S, Feng, J, and G. Schöner, G (2020) "Motor Habituation: Theory and Experiment," *2020 Joint IEEE 10th International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob)* pp. 1-8. doi: 10.1109/ICDL-EpiRob48136.2020.9278068

Anselmi, F, Murray, M, Franceschiello, B (2020). A computational model for grid maps in neural populations. *J Comput Neurosci* 48**,** 149–159

Ashby F, Helie, S (2011) A tutorial on computational cognitive neuroscience, Modeling the neurodynamics of cognition. *J. Math. Psychol*, 55, 273-289.

Bicanski, A, Burgess, N (2019). A Computational Model of Visual Recognition Memory via Grid Cells. *Current Biology* 29, 979-990

Bolton, S, Hattie, J (2017), Cognitive and Brain Development: Executive Functions, Piaget, and the Prefrontal Cortex. *Arch. Psychol, 1, 3*.

Bonzon, P (2017). Towards neuro-inspired symbolic models of cognition: linking neural dynamics to behaviors through asynchronous communications. *Cogn Neurodyn*,11, 4, 327–353.

Bonzon, P (2019). Symbolic modeling of asychronous neural dynamics reveal potential synchronous roots of awareness. *Front. Comput. Neurosci*. | doi.org/10.3389/fncom.2019.00001.

Bonzon, P (2020). Modeling the synchronization of multimodal perceptions as a basis for the emergence of deterministic behaviors. *Front. Neurorobot.* | doi.org/10.3389/fnbot.2020.570358

Carandini, M. (2012). From circuits to behavior: a bridge too far? *Nature neurosci.*, 15 4, 505-507.

Carew, TJ, Walters, ET, Kandel, ER (1981). Classical conditioning in a simple withdrawal reflex in Aplysia californica. *The Journal of neuroscience*, 1(12), 1426-1437.

Cooper, R, Peebles, D (2015). Beyond Single-Level Accounts: The Role of Cognitive Architectures in Cognitive Scientific Explanation. *Topics in Cogn. Sci. 7* (2) 243–258.

Diamond, A, (2001). Looking closely at infants' performance and experimental procedures in the A-not-B task. *Behav Brain Sci*, 24 (1):39-42.

Doya, K (2007). *Bayesian Brain: Probabilistic Approaches to Neural Coding.* Cambridge, MA: MIT Press.

Frank, MJ (2015). Linking across levels of computation in model-based cognitive neuroscience. In, Forstmann, B, & Frank, M, *Wagenmakers, E-J (Eds), An introduction to model-based cognitive neuroscience, Springer*

Frank, MJ & Badre, D (2015). How cognitive theory guides neuroscience. *Cognition*, 135, 14-20

Frégnac, Y (2017).Big data and the industrialization of neuroscience: A safe roadmap for understanding the brain? *Science*, 358(6362):470-477.

Hodgkin, AL, Huxley, AF (1952). A quantitative description of membrane current & its application to conduction & excitation in nerve. *Journal of Physiology*, 17(4), 500-544.

Hopfield, J (1982) Neural networks and physical systems with emergent collective computational abilities", *Proceedings of the National Academy of Sciences of the USA*, 79, 8, 2554–2558.

Huyck, C, Passmore, P (2013). A review of cell assemblies. *Biol Cybern.* 107(3) ,263–288.

Knill, K, Pouget,A (2004). The Bayesian brain: the role of uncertainty in neural coding and computation. *Trends in Neuroscie.*, 27,12, 712-719.

Kragel, P. et al. (2018). Representation, Pattern information, and Brain Signatures : From Neurons to Neuroimaging. *Neuron*, 99(2), 257-273

Kriegeskorte, N, Douglas, P (2018). Cognitive computational neuroscience. *Nature Neurosci.*, 21,1148–1160.

Langston, R. & al. (2010). Space and direction are already represented in specific neurons when rat pups navigate a location for the first time. *Science*, 328, 1437-1598

Letzkus, J, Wolff, S, Lüthi, A (2015). Disinhibition, a Circuit Mechanism for Associative Learning & Memory. *Neuron*, 88(2), 264-276.

Lewis M, Purdy S, Ahmad S and Hawkins J (2019) Locations in the Neocortex: A Theory of Sensorimotor Object Recognition Using Cortical Grid Cells. *Front. Neural Circuits* 13:22.

Love, B (2015). The algorithmic Level is the Bridge between Computation and Brain. *Topics in Cogn . Scie 7* (2) 230–242.

Martin, A. (2016). GRAPES-Grounding representations in action, perception, and emotion systems: How object properties and categories are represented in the human brain. *Psychon Bull Rev* **23,** 979–990.

McCune, L (2001) Is a field theory of perseverative reaching compatible with a Piagetian view? .*Behav Brain Sci*, 24 (1):53.

Moore, M, Meltzoff, A (2004). Object Permanence After a 24-Hr Delay and Leaving the Locale of Disappearance: The Role of Memory, Space and Identity. *Develop. Psychol.* 40(4):606-20

Moser E. I., Moser M.-B. (2008). A metric for space. *Hippocampus*, 18 1142–1156.

Mulder, MJ, van Maanen, L, Forstmann, BJ (2014). Perceptual decision neurosciences – A model-based review. *Neuroscience*, 277, 872–884

Müller. U, Carpendale, J (2001). Objectivity, intentionality, and levels of explanation. *Behav Brain Sci*, 24 (1):56-57.

Munakata, Y (1998). (1998) Infant perseverative and implications for object permanence theories: A DP model of the A-not-B task. *Develop.Sci* 1:161–84.

O'Keefe, J, Dostrovsky, J (1971). The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34 (1):171-5.

Piaget, J (1936). *La naissance de l'intelligence chez l'enfant*. Delachaux et Niestlé [english translation: Piaget, J (1952). *The origins of intelligence in children,* New York International Universities Press].

Piaget, J (1937). *La construction du réél chez l'enfant*. Delachaux et Niestlé [english translation: Piaget, J (1954). *The construction of reality in the child*, Basic Books].

Piaget, J, (1945). *La formationdu symbole chez l'enfant*. Delachaux et Niestlé [english translation: Piaget, J (1962). *Play, dreams and imitation in childhood.* W.W.Norton & Co].

Ruffman, T (2001). Understanding A-not-B errors as a function of object representation and deficits in attention rather than motor memories. *Behav Brain Sci*, 24 (1):61.

Rumelhart, D, McClelland, J (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Volume 1: Foundations*. Cambridge, MIT Press.

Schöner, E ,Thelen, G. (2006). Using Dynamic Field Theory to Rethink Infant Habituation. *Psych. Review*, 113, 2, 273-299

Thelen, E. & Smith, L. B. (1994) *A dynamics systems approach to the development of perception and action.* MIT Press

Thelen, E, Schöner, G, Scheier, C, Smith LB. (2001). The dynamics of embodiment: a field theory of infant perseverative reaching. *Behav Brain Sci*, 24 (1):1-34.

Van der Velde , F, de Kamps, M (2015). The necessity of connection structures in neural models of variable binding. *Cogn Neurodyn*, 9:359–37.

Van Gelder, T (1998). The dynamical hypothesis in cognitive science. *Behav. Brain. Sci*, 21:1-14

Westermann, G. et al. (2006). Modeling developmental cognitive neuroscience. *Trends in Cog. Sci*, 10,5.

Wills, T et al. (2010). Development of the Hippocampal Cognitive Map in Preweanling Rats. *Science*, 328, 1573-1576.

Wyble, B. at al (2020) Understanding Visual Attention With RAGNAROC: A Reflexive Attention Gradient Through Neural AttRactOr Competition. *Psych. Review*. 126,6: 1163-1198.

Zagha, E, Ge, X, McCormick D (2015). Competing Neural Ensembles in Motor Cortex Gate Goal-Directed Motor Output. *Neuron*, 88(3), 565-577.

View publication stats