Supplementary material for “A neuronal device for the control of multi-step computations”*

Ariel Zylberberg, Luciano Paz, Pieter R. Roelfsema,
Stanislas Dehaene, Mariano Sigman

Here we develop a neuronal implementation of a production system that solves a restricted Block World (BW) problem. The BW consists of a group of colored blocks arranged on top of a table in columns. The goal is to rearrange them to form a specific configuration. The player can only grab the blocks from the top of any column and place them on top of other columns or on the table. We restrict the general BW problem to a narrow table that can hold up to 3 columns of blocks. With these restrictions, the problem becomes analogous to the Tower of London puzzle [1]. The goal configuration we chose is to arrange all the blocks into a given target column in a specific order.

The neuronal populations are divided into sensory, production, state memory, forcing and variable value neurons (Fig. 2). The complete list of populations is:

- An early sensory layer that senses the game configuration of blocks (position and color).
- An attentional sensory layer that receives a sub-threshold input from the early sensory layer and top-down stimulation from later layers.
- Layers that encode only color and only positions, similar to the proposals in Ref. [2,3].
- Production neurons that integrate input, increasing their firing rate until saturation.
- The phasic firing neurons associated with the execution and inhibition of productions.
- Neurons that represent variables that can bind themselves to a color or position encoding neuron (variable neurons).

*This is part of the Supplementary Material for the paper A Zylberberg, L Paz, P R Roelfsema, S Dehaene, M Sigman, *A neuronal device for the control of multi-step computations*, Pap. Phys. **5**, 050006 (2013); **doi:** http://dx.doi.org/10.4279/PIP.050006; **Received:** 14 June 2013; **Accepted:** 9 July 2013; **Edited by:** G. Mindlin; **Licence:** Creative Commons Attribution 3.0.

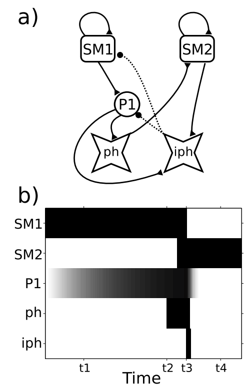

Figure 1: (a) A diagram of the connections involved in the execution of a production **P1**. The state memory **SM1** drives the execution of **P1** and the effect of the production is to inhibit the activity of **SM1** and excite a separate state memory **SM2**. The excitatory connections are drawn as thick lines with an inverted triangle at the receiving end. The inhibitory connections are the dotted lines with a circle at the receiving end. **SM1** is initially firing and reverberates its activity. **P1** integrates the input it receives and excites the phasic firing populations **ph** and **iph**. The former fires after **P1** crosses a threshold of activity and excites **SM2**. The later receives sub-threshold excitation from **P1** and **SM2** (this signals the production has completed its effect) and greatly inhibits both **P1** and **SM1**. (b) Mean firing rates over time for each population. The marked times show different moments of the production execution. Time **t1** shows part of the activity integration of **P1**. Time **t2** corresponds to the time when **ph** starts to fire due to the threshold crossing of **P1**. Time **t3** is when **iph** starts to fire and time **t4** shows the moment when **P1** and **SM1** were completely silenced. The firing rate is normalized between 0 (white) and 1 (black).

- Working memory neurons that encode the state of the computation process (state neurons).
- Neurons that send uniform and equally intense stimuli to other layers and force a winner-take-all (WTA) situation. They are used in variable value assignment or reminder, and we call them forcing neurons or forcers.
- A simplified motor layer that produces changes in the outer world configuration.
- A neuron that represents the position that is being attended (inspired in the processing focus from Ullman [4]).

To simplify the implementation, we allow the sensory layer neurons to configure discrete positions and colors for the blocks. We also propose full motor productions that move the highest block of a given column onto the top of another. These actions require the combination of multiple elementary movements. As we are interested in the problem solving skills that lead to a solution of the BW instance, we simplify the motor aspect of the task.

All the neuron populations (except the production neurons) are modeled as binary valued neurons. They can be “on” or “off” according to the input they

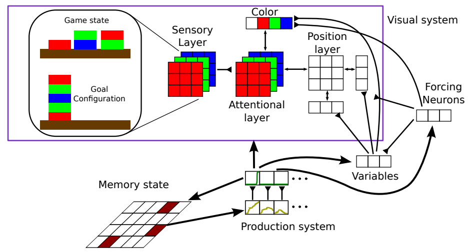

Figure 2: We present a sketch of the network that solves the blocks world problem. The sensory early visual system receives input from the BW configuration and excites the first attentional layer. The later is connected to color and position specific areas. The arrows show there is a connection between layers. The individual connections may be excitatory or inhibitory. The connections with the inverted triangle head indicate only excitatory connections exist.

receive.[1] The production neurons integrate their input and increase their firing rate (**P1** in Fig. 1). When they surpass a certain threshold, a population of phasic firing, binary neurons fires and execute the production they are associated to (**ph** in Fig. 1). Once the task is complete, another group of phasic firing, binary neurons becomes active and inhibits the production population activity (**iph** in Fig. 1). To "catch" the production sout's completion, we propose that the changed state encoding neuron excites the **iph** population in a sub-threshold manner. This, combined with another sub-threshold excitation driven by the production population, activates **iph** only after the whole task was completed. All productions were implemented in this fashion.

The early visual layer only receives input from the outer world and sends a sub-threshold stimulation to the attentional layer. This second group is a copy of the early visual layer that is not directly responsive to the outer world but is connected in a feed-forward and feed-back manner to higher visual layers, to the competition forcers and the production system. These first two tiers have small receptive fields (i.e., discern positions) and are sensitive to color. The higher levels of the visual system that we implement are populations sensible only to color or only to position. The variable neurons can be assigned to these higher level neurons and can reach the attentional layer through top-down stimulation (Fig. 2).

The productions —the set of allowed elemental operations— that we implement are:

1. To assign a value to a variable.
2. To remember the value of a variable.
3. To move the processing focus through the visual field.
4. To get the color being seen by the processing focus.

[1]In reality, the "on" state would be associated with a higher activity rate than the "off" or baseline state. In this work we use the normalized firing rate where 1 corresponds to the maximum firing rate and 0 to the "off" or baseline state.

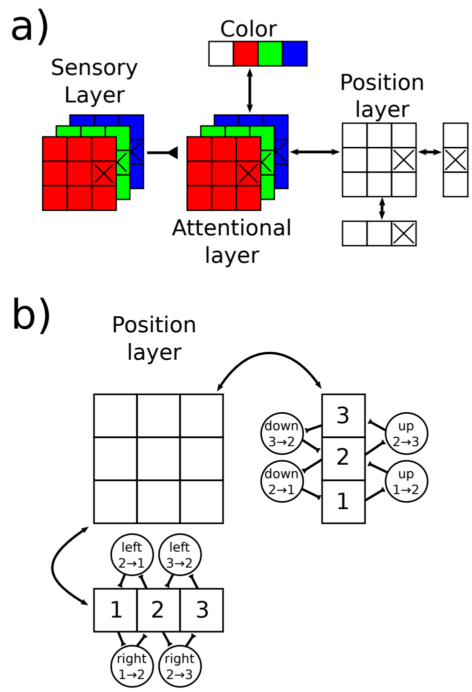

Figure 3: In (a) we show the visual layers and mark with an X the populations that are sensitive to the position $\{2, 3\}$. In (b) we show the asymmetric neurons that serve to move the processing focus one step up, down, left or right.

5. To search for the positions of the blocks of a given color.
6. To move a block from one column to another.

The assignment of values to variables is done through the enhancement of the plastic synapses that link the population pointer of a variable to the value of the encoding population. The *remember* operation forces the value fields to a WTA state and activates the population pointer of the variable. The bound value of a variable receives greater input from the pointer due to the strengthened synapse and wins the competition. The following co-activation further strengthens the binding synapse. The *remember* operation must be executed in order to sustain the value binding through long periods of time, as the synapse strength decays when there is no co-activation.

The movement of the processing focus can be done in two distinct ways. The first way is a straightforward value assignment. The second is to get the location currently bound and move it one step up, down, left or right. To do this, we use the same mechanism as the one used to increase a number by one in the implementation of the counter. First, a given location $\{x, y\}$ in the viewed field is touched by the receptive fields of many neurons. The sensory and attentional

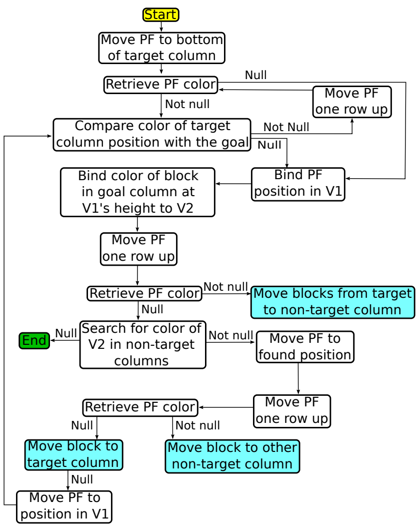

Figure 4: Flow chart of the implemented program that solves the BW instances. Each rectangle is a production that fires. The cyan colored squares represent motor productions. The PF is the processing focus, and V1 and V2 are two variable neuron populations that can bind to different areas.

layers both have receptive fields that distinguish the position of the point. The position layer has two levels, one that encodes the point $\{x, y\}$ and another that encode the x and y coordinates separately. At this level, we add the asymmetric populations that allow activity from a neuron encoding a given x or y to flow to a neighboring position ($x + 1$ or $x - 1$) in an analogous way as the counter increases the *Count* value by one (Fig. 3). The productions *PrepareNext* and *ClearNext* that were used in the counter are reimplemented here but instead of *Next* we used *Up*, *Down*, *Left* and *Right*.

The motor actions can be very complex. As we are interested in the problem solving capabilities of the network and not on the specific motor production mechanisms, we instantiate actions that move the topmost block of a given column to the top of another. If the later is empty, it places it on top of the table.

Now we detail the inner workings of the "color search" and the "get color" productions. During the first, the processing focus is silenced and a color encoding neuron (the color to be searched) is activated. This is usually done remembering the value assigned to a variable. The color layer and early visual layer excitation combined with a forcing stimulation activate certain populations in the attentional visual layer. These are the ones that simultaneously encode the color being searched and are stimulated by the early visual system (there is a block of that color in that position). After the attentional layer is activated, the higher levels of position encoding are forced in a WTA fashion so that they select one position to be encountered. This production is analogous

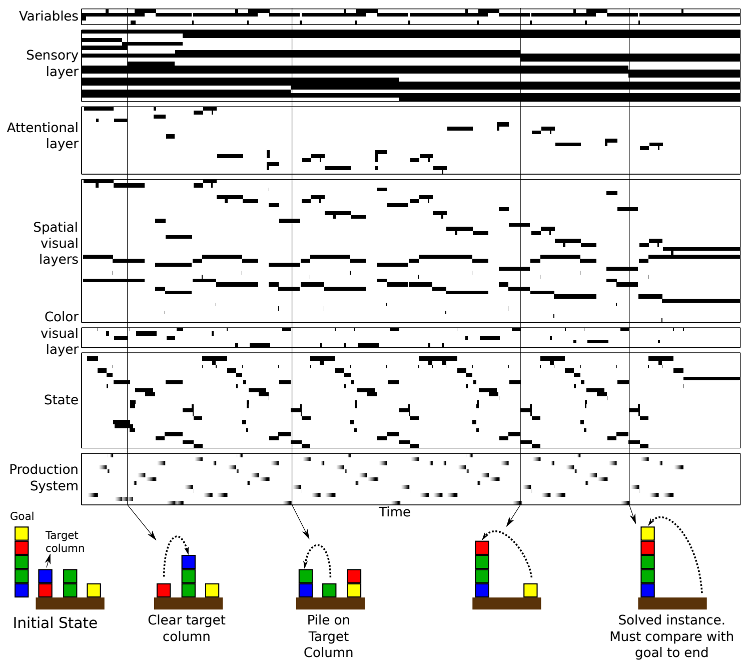

Figure 5: Mean firing rate of the relevant neuronal populations involved during the resolution of an instance of BW. The rate is normalized between 0 (white) and 1 (black). The horizontal axis represents the time in arbitrary units. We also show the BW configuration that is being solved. We mark certain moments when motor actions are executed and show their effect on the blocks configuration.

to the "get color" production. The processing focus is modeled as a variable neuron that is bound to position values. To get its color, the color layer is silenced and the attentional layer population, which encode the focus position and receives input from the early visual system, is activated. This then forces a WTA competition amongst the color encoding neurons. This mechanism is also proposed in Ref. [2,3].

The BW instances we wish to solve have all the blocks placed in one target column. Our implementation of the network is intended to solve these not necessarily in the minimum amount of moves. We wrote a finite state controller (Fig. 4) that acts according to the state encoded by the state neurons. Initially, the processing focus is moved to the target column and this then activates the state neuron that targets the compare production between the target and goal columns blocks. The rest of the program unfolds and elicits a complicated network activity (Fig. 5). The state neurons force an accumulation of a production and, when it reaches a threshold, triggers a phasic burst which changes the state and/or the activity in the visual system.

The productions are chained in a specific order that ultimately solve every instance of the BW game (Fig. 4). In our implementation, the state memory neurons indicate the block (blocks of Fig. 4) that is being executed and determine which production wins the WTA competition.

References

[1] T Shallice, *Specific impairments of planning*, Philos. T. R. Soc. Lond. **298**, 199 (1982).

[2] A Zylberberg, S Dehaene, P R Roelfsema, M Sigman, *The human Turing machine: A neural framework for mental programs*, Trends Cogn. Sci. **15**, 293 (2011).

[3] P R Roelfsema, *Elemental operations in vision*, Trends Cogn. Sci. **9**, 226 (2005).

[4] S Ullman, *Visual Routines*, Technical report, Massachusetts Institute of Technology: Artificial Intelligence Laboratory (1983).