
Organising Sequential Memory in a Neuromorphic Device Using Dynamic Neural Fields.

Raphaela Kreiser^{1*,**}, Dora Aathmani^{2,**}, Ning Qiao¹, Giacomo Indiveri¹,
Yulia Sandamirskaya^{1*},

¹ *Institute of Neuroinformatics, University of Zurich and ETH Zurich, Zurich, Switzerland*

² *The School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-250 USA* ** *These authors have contributed to the manuscript equally (double first authorship).*

Correspondence*:

Yulia Sandamirskaya, Raphaela Kreiser
ysandamirskaya@ini.uzh.ch, rakrei@ini.uzh.ch

2 ABSTRACT

3 Neuromorphic Very Large Scale Integration (VLSI) devices emulate the activation dynamics of
4 biological neuronal networks using either mixed-signal analog/digital or purely digital electronic
5 circuits. Using analog circuits in silicon to physically emulate the functionality of biological neurons
6 and synapses enables faithful modelling of neural and synaptic dynamics at ultra low power
7 consumption in real-time, and thus may serve as computational substrate for a new generation of
8 efficient neural controllers for artificial intelligent systems. Although one of the main advantages of
9 neural networks is their ability to perform on-line learning, only a small number of neuromorphic
10 hardware devices implement this feature on-chip. In this work, we use a reconfigurable on-
11 line learning spiking (ROLLS) neuromorphic processor chip to build a neuronal architecture for
12 sequence learning. The proposed neuronal architecture uses the attractor properties of winner-
13 takes-all (WTA) dynamics to cope with mismatch and noise in the ROLLs analog computing
14 elements, and it uses its on-chip plasticity features to store sequences of states. We demonstrate,
15 with a proof-of-concept feasibility study how this architecture can store, replay, and update
16 sequences of states, induced by external inputs. Controlled by the attractor dynamics and an
17 explicit destabilizing signal, the items in a sequence can last for varying amounts of time and
18 thus reliable sequence learning and replay can be robustly implemented in a real sensorimotor
19 system.

20 **Keywords:** Neuromorphic engineering; on-chip learning, sequence learning; dynamic neural fields

1 INTRODUCTION

21 Mixed-signal analog-digital neuromorphic Very Large Scale Integration (VLSI) systems emulate the
22 biophysics of cortical neurons and synaptic connections between them using the physics of silicon electronic
23 devices (Moradi et al., 2017). Computation and memory are co-localized in these systems. Furthermore,

24 as communication of signals across neurons and modules is asynchronous and data-driven, this leads to
25 ultra low-power consumption and highly efficient real-time processing. While some recent neuromorphic
26 hardware devices aim at speeding up the processing time of computational neuroscience simulations,
27 e.g. SpiNNaker (Furber et al., 2012) or HICANN (Schemmel et al., 2010; Benjamin et al., 2014), other
28 neuromorphic hardware systems have been developed as basic research tools for emulating the properties
29 of real cortical circuits in real-time (Chicca et al., 2014b; Qiao et al., 2015). These latter systems are
30 particularly well suited also as a neural computing substrate for real-time technical applications that can
31 profit from their massively parallel architecture, ultra-low power consumption, and small form-factor, for
32 example, in control of real-time robotic systems with embedded processing (Conradt et al., 2009; Ijspeert,
33 2008; Xi, 2016).

34 Several mixed signal analog/digital neuromorphic devices were previously used for spiking pattern
35 classification (Mitra et al., 2009; Corradi and Indiveri, 2009; Kreiser et al., 2017), motor controllers
36 for robotic devices (Perez-Peña et al., 2013; Serrano-Gotarredona, 2009; Perez-Pena et al., 2014b;
37 Cartiglia et al., 2018; Kreiser et al., 2018; Glatz et al., 2019, submitted), simple stimulus-response
38 based agents (Conradt et al., 2009; Indiveri et al., 2001), and were successful at the level of early sensory
39 processing in vision and audition (Liu and Delbruck, 2010). Typically the neural network connectivity is
40 determined at design-time off-line, to fulfill specific task requirements. However, neuromorphic systems
41 with on-chip learning abilities (Mitra et al., 2009; Qiao et al., 2015; Davies et al., 2018; , ???). In particular,
42 neuromorphic systems that have the ability to modify synaptic weights between neurons with biologically
43 plausible plasticity mechanism allow the construction of low-power adaptive neural processing systems
44 that can be used for building autonomous cognitive agents (Chicca et al., 2014a).

45 In this work, we use a neuromorphic device that is equipped with analog on-chip learning circuits, in
46 order to store a sequence of visual inputs in a robotic sensorimotor loop. Learning such sequence is enabled
47 by a neural architecture that can cope with the challenges brought about by the mixed signal analog/digital
48 neuromorphic hardware.

49 One of the main challenges in applications of mixed signal neuromorphic systems is that of devices
50 mismatch – variability in properties of computing elements due to the fabrication process and the sub-
51 threshold operation. This leads to output noise and variability in properties of neurons and synapses, if
52 realized with analog circuits (Neftci et al., 2011). To make neuromorphic hardware work reliably in face of
53 its variability, the silicon neural networks need to form representations that are stable against mismatch.
54 Biological neural networks face a similar problem of fluctuations of biochemical parameters, nevertheless
55 animals are capable of precise and reproducible behavior, thus biology must have developed efficient
56 solutions to this problem.

57 One of these solutions is population dynamics with soft winner-take-all (WTA) connectivity. The WTA is
58 a computational primitive that leads to continuous-attractor dynamics that were found to be characteristic
59 for many cortical and subcortical neural networks (Wilson and Cowan, 1973; Gerstner and Kistler, 2002).
60 Moreover, in the framework of Dynamic Neural Fields (DNFs), an analogy is drawn between population
61 dynamics with WTA connectivity and behavioral dynamics, observed in experiments studying perceptual,
62 motor, and cognitive behavior (Schöner and Spencer, 2015). DNFs, originally developed to describe the
63 activation dynamics of large neural populations (Amari, 1977; Wilson and Cowan, 1973; Grossberg, 1988),
64 have been used to account for human cognition (Johnson et al., 2008; Schöner and Spencer, 2015) and
65 to develop cognitive architectures for robotics (Sandamirskaya et al., 2013; Richter et al., 2012; Lipinski
66 et al., 2009; Bicho et al., 2012; Gaussier and Zrehen, 1994).

67 Robotic demonstrators are a novel tool entering the area of neuroscience that allows better understanding
68 of brain structure and functionality by offering an “embodiment” for computational models (Pfeifer et al.,
69 2007; Wolfgang and Jean-pierre, 2003; Thelen, 1995). A full spectrum of the currently ongoing projects in
70 the area of neuromorphic robotics was listed and extensively analyzed in (Krichmar and Wagatsuma, 2011).
71 The emerging tendency in brain-inspired robotics is to mimic the organization of nervous infrastructure in
72 order to obtain similar functionality.

73 WTA connectivity is one of the organization schemes that was postulated in the past as an elementary
74 computing unit for both neural processing and neuromorphic architectures (Indiveri et al., 2009; Neftci et al.,
75 2013; Rutishauser and Douglas, 2009). DNF dynamics have been implemented on analog neuromorphic
76 devices (Sandamirskaya, 2013) and in this paper, we build on this work to demonstrate how sequences of
77 states, realised by application of WTA networks, can be learned in plastic synapses on a neuromorphic chip.
78 The neuromorphic chip can be embedded in a robotic agent where states of the sequence are driven by
79 sensory input. In particular, we show how on-chip plastic synapses store a sequence of visual information
80 that is perceived by a neuromorphic camera (Delbruck and Lichtsteiner, 2006), mounted on a robotic agent.

81 Neuronal mechanisms for representing sequences have been studied both in humans (e.g. in behavioral
82 experiments on learning movement sequences (Hikosaka et al., 2002; Deroost et al., 2006) or in serial order
83 errors, e.g. in typing or language production (Henson, 1998), as well as in animals (e.g. in rats performing
84 grooming movements (Aldridge and Berridge, 2003) or during navigation in mazes (Foster and Wilson,
85 2006)). Furthermore, recent recordings from basal ganglia in humans allowed to gain insights into the
86 neuronal mechanisms of learning movement sequences, in particular, in detecting errors in serially ordered
87 sequences of tones (Herrojo Ruiz et al., 2014).

88 From a multitude of neural models for sequence learning (e.g., Rabinovich et al., 2006; Deco and Rolls,
89 2005; Wörgötter and Porr, 2005), we selected one that allows an agent to learn sequences from real
90 sensory data and produce sequences through a physical motor system. It achieves the required stability and
91 robustness of actuators plans and perceptual decisions using the DNF, or likewise WTA attractor dynamics,
92 and its transition ability through an explicit destabilizing signal (Sandamirskaya and Schöner, 2010a). This
93 model is inspired by neuronal findings on serial order encoding in the brain (Aldridge and Berridge, 1998;
94 Carpenter, 1999; Clower and Alexander, 1998; Procyk et al., 2000) and behavioral data on serial order
95 errors (Henson, 1998).

96 In this work, we implement the model on the ROLLS (Reconfigurable On-Line Learning Spiking)
97 neuromorphic processor (Qiao et al., 2015). The model is a neural architecture that creates stable attractor
98 states in a population of “ordinal” neurons that represent the ordinal position in a sequence. An active group
99 of ordinal is associated – through synaptic plasticity – with a neuronal representation of the “content” for
100 each item in a sequence. This content is represented with a population that features WTA network topology
101 and is driven by sensory inputs during sequence learning. During learning, associations between ordinal
102 positions and different regions on the content WTA are established. Learning is supported by co-existence
103 of the attractor states in the two neural populations that are sustained for macroscopic amounts of time.
104 During sequence replay – i.e. acting out of a memorised sequence, – each state in the content WTA is
105 activated by the ordinal nodes for a variable amount of time, that is controlled by an external “condition of
106 satisfaction” signal. To our knowledge, this simple example is the first demonstration of on-chip learning
107 that proceeds autonomously in a mixed signal neuromorphic hardware in a closed sensorimotor loop.

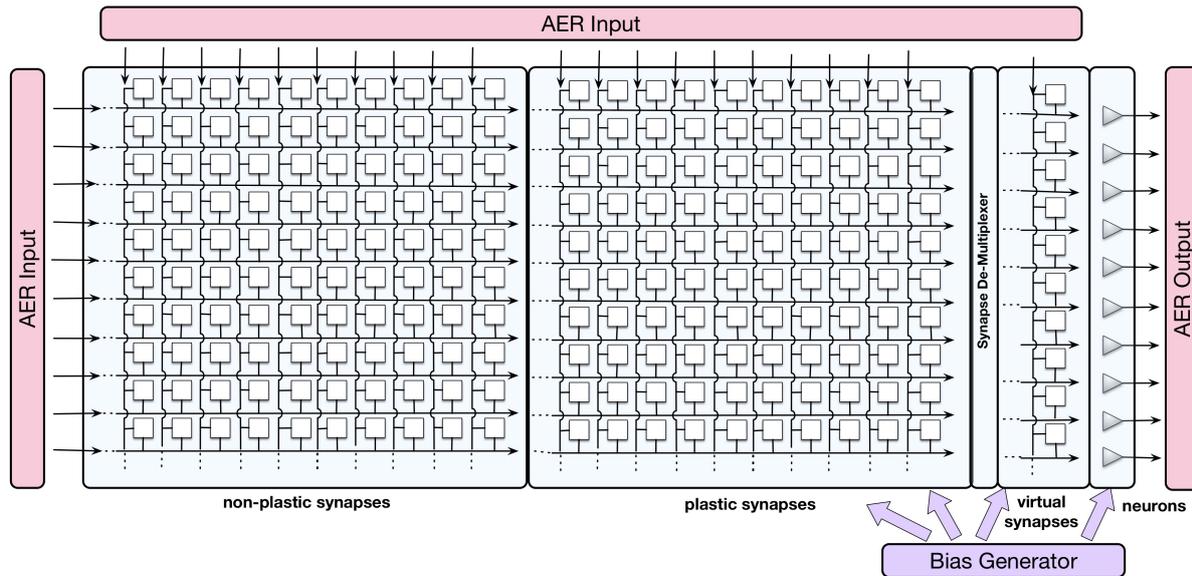


Figure 1. Block diagram of the ROLLS chip architecture. Triangles on the right represent silicon neurons, squares are synapses that are organized in three arrays: non-plastic, plastic, and virtual. The AER blocks manage the input and output traffic of spikes, and the bias generator allows to program different parameter settings of the analog circuits.

2 MATERIAL AND METHODS

108 2.1 Neuromorphic hardware: the ROLLS device

109 The ROLLS neuromorphic device used in this work comprises mixed-signal analog/digital neuron and
 110 synapse circuits that can exhibit a range of biologically realistic dynamics (refractory period, time-course
 111 of integration and leakage, firing rate adaptation, short- and long-term plasticity, etc.). The silicon neuron
 112 circuits implement a model of the adaptive exponential integrate-and-fire (IF) neuron (Brette et al., 2007).
 113 A schematic diagram of the chip architecture is shown in Fig. 1.

114 The device comprises 256 analog silicon neuron circuits, an array of 256×256 non-plastic programmable
 115 synapses, an array of 256×256 plastic synapses, and a 256×2 linear integrator filters – “virtual synapses”
 116 that can be used to direct external inputs to the neurons. The non-plastic synapses consist of analog circuits
 117 that reproduce short-term adaptation dynamics (Rasche and Hahnloser, 2001; Boegerhausen et al., 2003) and
 118 digital circuits that set the programmable weights. On ROLLS, four different weights are available on chip
 119 and each one of them can be set to a required value regulated by on-chip biases.

The plastic synapses contain analog learning circuits and digital state-holding logic. The learning circuits implement the synaptic plasticity model proposed in (Brader et al., 2007), which is particularly well-suited to VLSI implementation. According to this rule, the synaptic weights are updated based on the timing of the pre-synaptic spike, the state of the post-synaptic neuron’s membrane potential, and an intrinsic calcium variable, which depends on the recent spiking activity. On a long time-scale, the weight values of the plastic synapses drift toward one of two possible states, depending if their value is above or below a weight-threshold parameter. The synapses therefore are bistable and robust to input activity and state-dependent variability. Equations (1) and (2) formalize the plasticity weight update mechanism that operates

on short time-scales:

$$w_i = w_i + \Delta w^+, \text{ if } V_{mem}(t_{pre}) > \theta_{mem} \text{ and } \theta_1 < Ca(t_{pre}) < \theta_{max}; \quad (1)$$

$$w_i = w_i - \Delta w^-, \text{ if } V_{mem}(t_{pre}) < \theta_{mem} \text{ and } \theta_1 < Ca(t_{pre}) < \theta_{max}. \quad (2)$$

120 Here, w_i is the synaptic weight of a plastic synapse; the terms Δw^+ and Δw^- determine the amplitude
 121 of the weight's increase and decrease, respectively. $V_{mem}(t_{pre})$ is the post-synaptic neuron's membrane
 122 potential at the time of the pre-synaptic spike arrival. If V_{mem} is above the threshold θ_{mem} , the post-synaptic
 123 neuron must be about to spike, leading to the temporal sequence of pre- and post-synaptic spikes that
 124 leads to potentiation of the synapse (following to the "classical" spike-timing dependent plasticity rule),
 125 whereas if V_{mem} is below the threshold, the post-synaptic neuron is likely to have just spiked, leading to
 126 temporal sequence of spikes that correspond to depression of the synapse. The Ca variable represents the
 127 neuron's Calcium concentration, which is proportional to the neuron's recent spiking activity. This is the
 128 variable representing the "third factor" in this three-factor learning rule that gates plasticity (Nefci, 2018).
 129 The parameters θ_{min} , θ_2 , and θ_{max} are the thresholds that determine in which conditions the weights are
 130 updated.

131 The long-term drift that determines the synaptic weight bistability properties, and which is superimposed
 132 to this STDP plasticity rule, is governed by the following equations:

$$\frac{d}{dt}w_i = C_{drift}, \text{ if } w_i > \theta_w \text{ and } w_i < w_{max}; \quad (3)$$

$$\frac{d}{dt}w_i = -C_{drift}, \text{ if } w_i < \theta_w \text{ and } w_i > w_{min}, \quad (4)$$

133 where C_{drift} determines the rate of the drift, θ_w is the weight threshold that determines the direction of
 134 the drift, and w_{min} , w_{max} – the value of the high and low weights, respectively. Thus, plastic synapses are
 135 binary in the long term.

136 Fig. 2 shows traces of different components of an active neuron on the ROLLS chip and one of its plastic
 137 synapses, whose weight increases in real time in response to a constant input. A thorough description and
 138 characterization of the circuits can be found in (Qiao et al., 2015).

139 Additional peripheral analog/digital input-output circuits for both receiving and transmitting spikes in
 140 real-time on- and off-chip follow an Address-Event Representation (AER) protocol (Boahen, 1999) and can
 141 be used to stimulate individual synapses on the chip. An on-chip programmable bias generator, optimized
 142 for subthreshold circuits allows the user to create networks with different properties and topologies and to
 143 program the properties of the synapses and neurons (such as time constants, leak currents, etc.).

144 The ROLLS was fabricated using a standard 180 nm CMOS 1P6M process. It occupies an area of
 145 51.4 mm^2 with approximately 12.2 million transistors.

146 2.2 Neuromorphic implementation of a soft-WTA network

147 Implementing a neural architecture, or model, on the ROLLS chip involves two steps. First, the non-
 148 plastic synapses are configured to create the static part of the neural network. For each synapse, one of
 149 four available weights can be selected, alongside with their type (inhibitory or excitatory). In the second
 150 step, the parameters of the neuron and synapse circuits are selected by setting the on-chip biases. These
 151 parameters are globally shared by all neurons and synapses.

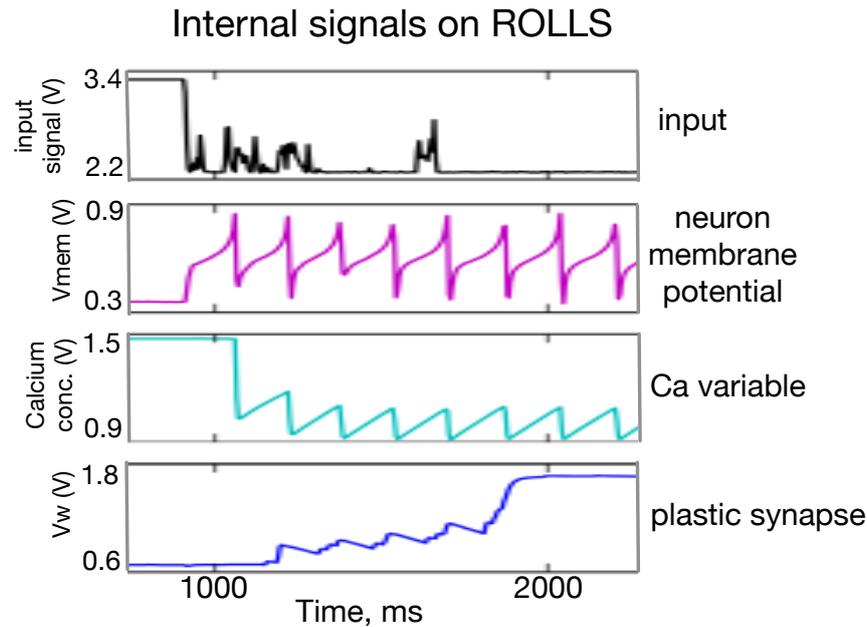


Figure 2. Oscilloscope traces of subthreshold activity of a silicon neuron and a plastic synapse. *From top to bottom:* the input signal to the neuron with a number of (random) spikes, low-pass filtered by the “virtual” synapse; the neuron’s membrane potential; a variable that models the neuron’s calcium concentration; and the synaptic weight of a plastic synapse to which another external input is applied. One can observe potentiation of the plastic synapse after each spike of the monitored neuron and diffusion of the synapse’s weight to a high value after it reaches a certain threshold.

152 In this work, we use a simple 1D Dynamic Neural Field (DNF) (Schöner and Spencer, 2015) to represent
 153 items in a sequence. On the neuromorphic chip, such DNF can be realised using WTA connectivity
 154 pattern (Sandamirskaya, 2013). Such WTAs are considered building blocks for cognitive neuromorphic
 155 architectures (Neftci et al., 2013). In fact, two connectivity patterns can be used to realise a WTA network
 156 on the ROLLS chip, as shown in Fig. 3. The red and blue squares in the figure represent excitatory and
 157 inhibitory synapses that connect neurons (triangles on the right). Tables 1 and 2 present a list of exemplary
 158 biases for neural and synaptic circuits, respectively, that generate winner-take-all behavior in a configuration
 159 presented in the Fig. 3b.

160 In order to realize a soft-WTA behavior (the term *soft* indicates that a group and not only a single neuron
 161 wins the competition) on the ROLLS chip, two types of connectivity networks can be configured.

162 In both settings, local groups of neurons can stabilize their activity by excitatorily projecting to themselves
 163 and to their nearest neighbors. Global inhibition ensures that only one group is active at a time and keeps
 164 the network activity from spreading across the whole population. Stable activity bumps can be achieved
 165 with two connectivity patterns in a population of spiking neurons. In the first pattern (A), we define the
 166 extent of region around each neuron, in which it will be connected to its neighbors with excitatory synapses.
 167 Every connection exceeding this excitatory range will form an inhibitory synapse (see Fig. 3a). In the
 168 second pattern (B), a separate group of inhibitory neurons is introduced, which task is to suppress the
 169 activity of the excitatory population (Fig. 3b). In this case, all excitatory neurons are positively coupled to
 170 the inhibitory group, which inhibits them back.

171 Evidence from neuroscience suggests that inhibition is provided by a separate set of inhibitory
 172 interneurons in many cortical areas (Couey et al., 2013; Buetfering et al., 2014). Thus, a WTA network

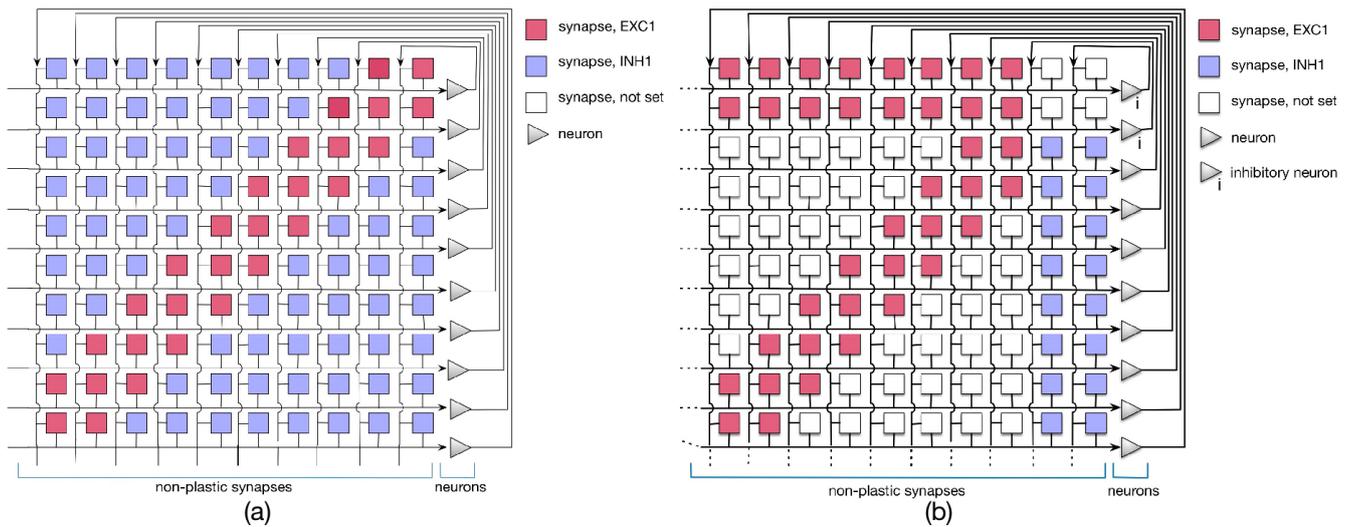


Figure 3. Setting the soft-WTA connectivity on the ROLLS chip. (a) Connectivity using excitatory synapses to the neuron itself and to its nearest neighbors and inhibitory synapses to all other neurons. (b) Connectivity scheme with excitatory projections of neurons to themselves, their nearest neighbors, and a small inhibitory group. The latter inhibits the whole excitatory population.

173 with a separate group presented in Fig. 3b appears to be more biologically plausible. However, from an
 174 engineering point of view, the choice may depend on other considerations, e.g. if the number of available
 175 synapses per neuron is limited, pattern (B) maybe preferable, while if rather the number of neurons is
 176 the bottleneck, the pattern (A) might be more advantageous. One should also consider that the additional
 177 inhibitory group adds a delay to the inhibitory feedback and may render parameter tuning more challenging.

178 **2.3 Neuromorphic architecture for sequence learning**

179 In this work, an architecture for serial order memory proposed in the framework of dynamic neural
 180 fields (Sandamirskaya and Schönner, 2010a; Duran and Sandamirskaya, 2012), was realized in neuromorphic
 181 hardware in the following way.

182 The connectivity pattern, shown in Fig. 4a, includes a population of *ordinal nodes* (yellow region in the
 183 figure). Each ordinal node, if active, represents a position in a sequence. Ordinal nodes activate each other
 184 sequentially. This is achieved through a set of *memory nodes* (orange region in the figure): each ordinal
 185 node activates the respective memory node, which in its turn activates the *next* ordinal node. Memory nodes
 186 feature strong recurrent connections that lead to their self-sustained activity: memory nodes stay active
 187 even when the respective ordinal node is not active any more. Thus, the memory nodes “keep track” of the
 188 position in the sequence during sequential transitions (when ordinal nodes are inhibited, as will be shown
 189 below).

190 The *content dynamic neural field* (content DNF; blue region in the figure) represents the perceptual states
 191 and/or actions that can be associated with a sequential position (an ordinal node) with plastic synaptic
 192 connections (green region). The content DNF is connected to the action system of the agent and sets
 193 attractors to generate behavior. It also receives perceptual input during sequence learning that creates
 194 activity in this field that corresponds to the recognised, e.g. demonstrated, actions.

195 The *condition of satisfaction* (CoS) system (lilac region in the figure) detects when each initiated action
 196 or perceptual state has reached the intended outcome (Richter et al., 2012). To achieve this, the CoS node

197 is driven by a perceptual module that receives input signaling the currently perceived or performed action
198 and input from sensors, configured to activate the CoS node if the end of an action has been detected (e.g.,
199 end of presentation of the object during learning, or end of a goal-directed movement during replay).

200 When the CoS node is activated, it inhibits all ordinal nodes, in particular removing activation of the
201 currently active ordinal node and thus stopping the learning process that was strengthening synaptic
202 connection between this ordinal node and an active region in the content DNF. The activity in the content
203 DNF will also cease in the transition.

204 During learning, this happens because transition means switching to the next item, which happens
205 through sequence of “forgetting” and “detection” instabilities (the old object disappears and the new object
206 appears), which leads to decrease of activation in the DNF. If an item was successfully learned and there’s
207 no perceptual input (Laser off), the ordinal population “recalls” the learned item in the activity in the
208 content DNF. Since the DNF is connected to the CoS, successful learning (and not perceiving visual input)
209 triggers the transition. When the laser is switched on the DVS_{on} population, driven by the visual input,
210 becomes active and inhibits the CoS, next item can be learned.

211 During sequence replay, on the other hand, the activity peak in the content DNF is supported by the active
212 ordinal node. When the CoS becomes active and inhibits the ordinal nodes, the activity in the content DNF
213 also ceases. In both cases, the decrease of activity in the content DNF leads to deactivation of the CoS
214 node, which releases the inhibition on the ordinal nodes. The next ordinal node can become active now,
215 driven by the asymmetric connection from the previous memory node.

216 Fig. 4b shows how this neural dynamic architecture can be realized with populations of spiking neurons – a
217 step required for the implementation in neuromorphic hardware (Sandamirskaya, 2013). Several constraints
218 have to be taken into account: (1) the limited amount of silicon neurons, (2) robustness to mismatch, and
219 (3) shared parameter settings across all neurons that need to exhibit different firing behaviors.

220 To cope with mismatch, we used populations of 10-20 neurons to represent a neuronal node (ordinal,
221 memory, or CoS nodes).

222 **Ordinal groups:** Each ordinal group contains 20 silicon neurons in our experiments, inter-connected via
223 excitatory synapses in an all-to-all fashion. Silicon neurons in different ordinal groups inhibit each other,
224 forming a WTA network. This allows only one ordinal group to be active at a time.

225 **Memory groups:** each ordinal group excites a corresponding memory group that contains 10 neurons.
226 Memory groups remain active for the whole trial due to their high self-excitation. Each memory group has
227 excitatory synapses to the next ordinal group. At the same time, they signal whether their corresponding
228 ordinal node has already been activated, by slightly inhibiting it. This ensures that the ordinal node that has
229 not yet been activated receives the highest excitatory input in the transition phase.

230 **Content group:** every content neuron is connected via plastic synapses with ordinal neurons with the
231 possibility to strengthen synaptic weights towards an active region in the content WTA. Initially, all plastic
232 synapses are depressed (have a low weight) and can only become potentiated once the ordinal and content
233 neurons are co-active.

234 **CoS group** contains 10 neurons that are externally stimulated upon a keystroke in most experiments here
235 in order to trigger a transition.

236 Fig. 5 shows the connectivity matrix of non-plastic synapses, set on the neuromorphic hardware to realize
237 the sequence learning architecture. Fig. 5a shows the implemented architecture for storing a sequence of

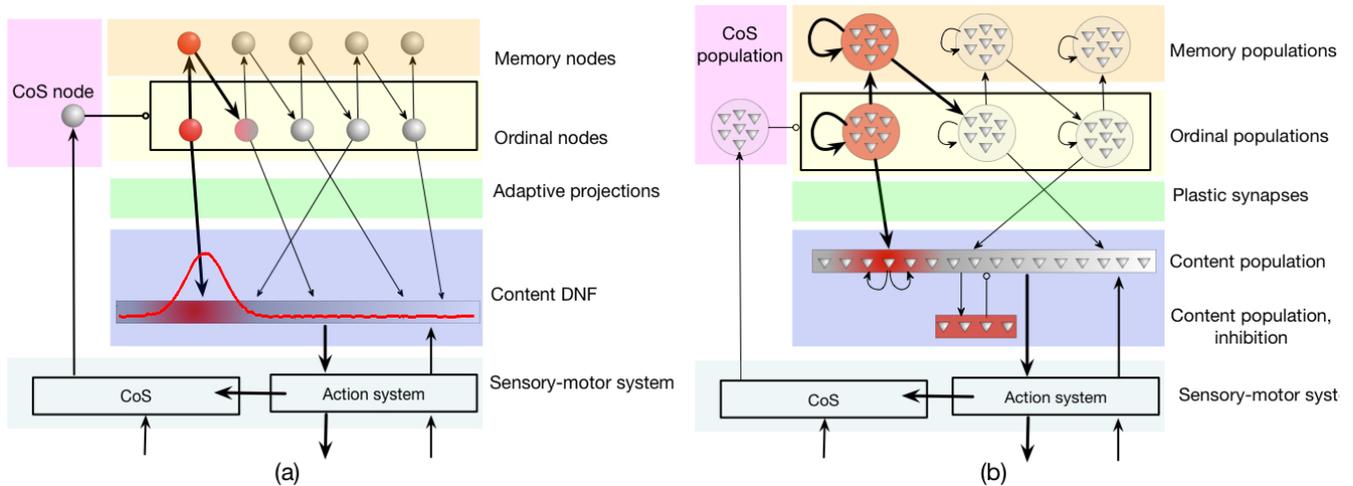


Figure 4. The serial order architecture, introduced by Sandamirskaya and Schöner (2010a). (a) The continuous version of the serial order architecture. A set of discrete neural-dynamic nodes represents ordinal positions within a sequence. The content Dynamic Neural Field (DNF) represents the perceptual or motor features of the stored items. A sequence of items is learned in adaptive connections between the ordinal nodes and the content DNF. (b) The neuromorphic realization of the architecture using populations of neurons. Note that in order to create stabilized peaks of activation that correspond to self-sustained activation of a neural population, neurons within a group need to be recurrently connected. CoS is the condition of satisfaction system that detects sequential transitions both during sequence learning and acting-out.

238 three items and Fig. 5b shows the connectivity matrix for storing a sequence with five items. Note that the
 239 hardware used here is a research prototype with only 256 analog neurons, which, along with mismatch,
 240 limits the number of items that the system can store. For this reason, the size of the content DNF is reduced
 241 for the 5-items architecture.

242 To achieve the desired behavior of the architecture, parameters specifying neuron and synapse dynamics
 243 have to be set in order to meet certain requirements posed on neural populations. Since ROLLS chip
 244 features only 4 different excitatory weights, we additionally used potentiated plastic synapses to strengthen
 245 connections within ordinal and memory groups. Fig. 6 shows the plastic synapse matrix after initializing a
 246 network for storing a sequence of three items. Each red dot shows a potentiated (high) plastic synapse at
 247 initialisation of the architecture (before learning). The diagonal of potentiated synapses for neurons 62 to
 248 92 (orange region) and 93 to 153 (yellow region) in Fig. 6 corresponds to the additional self-excitation
 249 in the memory groups and ordinal groups, respectively, using potentiated plastic synapses. Non-plastic
 250 connectivity in these groups is enhanced by using 30% of randomly potentiated plastic weights. Since state
 251 of the plastic synapses can not be read-out directly, we used a protocol to read them out, in which each
 252 synapse is stimulated and activity of the post-synaptic neuron is read-out, at turned-off plasticity.

3 RESULTS

253 Storing a three-items sequence

254 Fig. 7 shows how a simple sequence of three items can be successfully learned and recalled on
 255 neuromorphic hardware. Here, the architecture is presented with a sequence of items A-B-C and stores
 256 them in plastic synapses that connect ordinal groups (I., II., III.) to the content DNF.

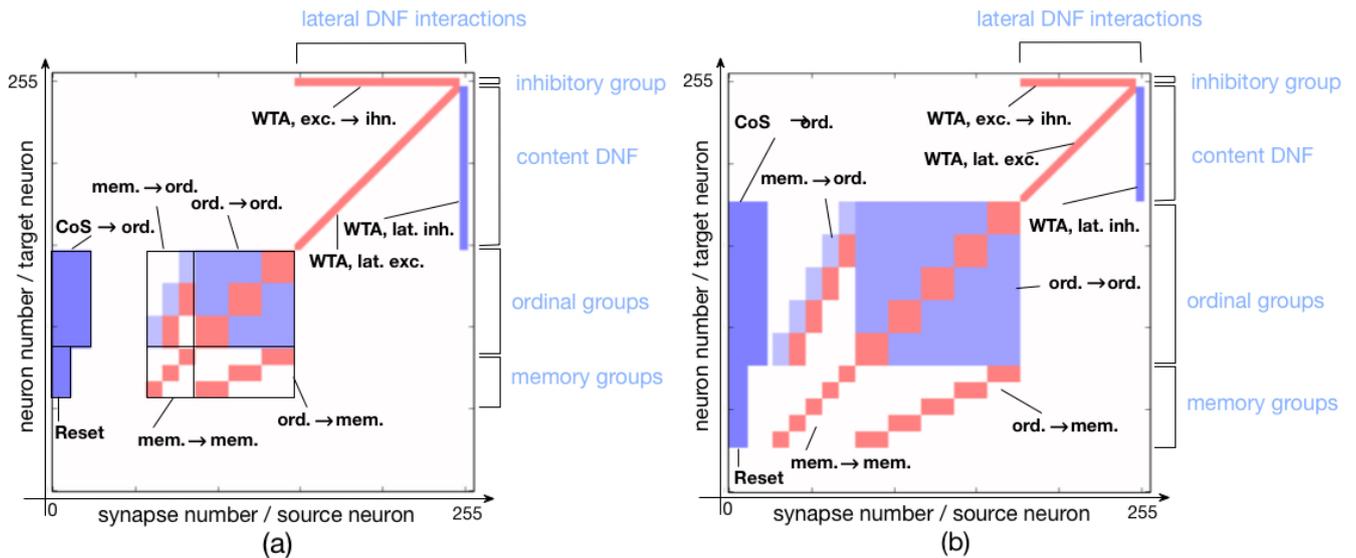


Figure 5. Connectivity matrix of non-plastic synapses, sent to the neuromorphic chip to encode the serial order architecture. Different shades of blue indicate different inhibitory synaptic weights (the darkest being the strongest). Red represents excitatory synapses, which all have the same weight. (a) Connectivity matrix for storing a sequence of three items. (b) Connectivity matrix for storing a sequence of five items.

257 Fig. 7a shows the raster plot of spikes recorded from the neuromorphic chip ROLLS. Each black dot
 258 corresponds to a spike, emitted by one of the 256 neurons on the ROLLS chip. Vertical axis shows neuron
 259 index (0-255), horizontal axis shows time in seconds. Colors mark different populations of neurons on the
 260 chip, according to the serial order architecture (Fig. 4): from bottom to top, lilac is the CoS population,
 261 orange are the memory groups, yellow are the ordinal groups, dark blue is the content WTA / DNF (its
 262 excitatory and inhibitory parts).

263 The neural activity is initially induced by an external input according to the following stimulation protocol.
 264 First, the sequence gets “launched” by the stimulation of the first ordinal group of neurons for 3000 ms
 265 with 200 Hz spiking input applied via virtual synapses (the “go” signal). The transition signal between
 266 sequential items, which suppresses the activity of ordinal neurons, is triggered by stimulating the CoS
 267 group for 500 ms with 800 Hz. An input to the content neurons, which will be replaced with sensory input
 268 in Section 3.1, consists of Poisson spike trains with firing rate distributed along the content population
 269 according to a Gaussian-function, centered at a selected “current item” neuron (marked with A, B, or
 270 C in Fig. 7a), with the maximum of 900 Hz and a standard deviation of 5 neurons. All content neurons
 271 additionally receive random inputs between 0 and 10 Hz to simulate sensory noise. During learning, the
 272 content neurons are stimulated for 6000 ms for each item in the sequence.

273 The WTA connectivity in the content population leads to formation of a localised “activity bump”: noise
 274 is suppressed by global inhibition and activity within the Gaussian is stabilised by the recurrent excitation.
 275 The neural location of the activation peak defines the content (A, B, or C here) and leads to strengthening
 276 of the connections between the active region in the content population and active neurons in one of the
 277 ordinal populations, according to the on-chip learning rule.

278 After learning the full sequence, an external input is sent to the inhibitory *reset group* which suppresses
 279 activity of the memory groups, resetting the ordinal system of the architecture. After the complete

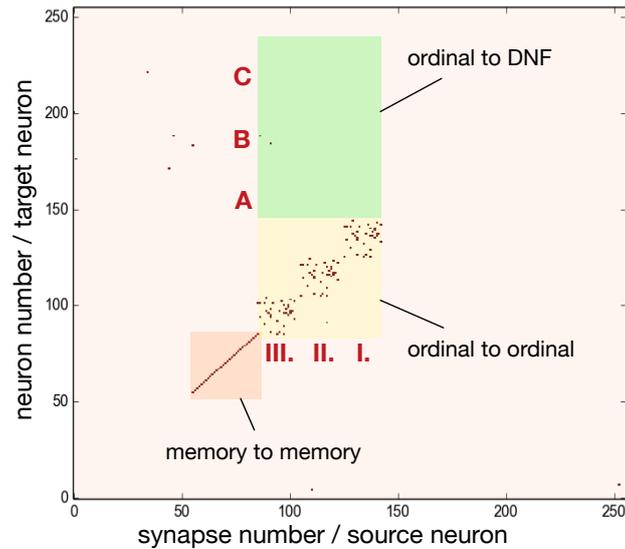


Figure 6. Initial state of the plastic synapses for a sequence of three items before sequence learning, as measured from the ROLLS chip. I., II., III mark pre-synaptic neurons that belong to the ordinal groups; A, B, C mark post-synaptic neurons that belong to different regions of the content DNF. Red dots show potentiated synapses, used for additional strengthening of recurrent connections in the ordinal and memory groups. These synapses don't participate in learning a sequence. Synapses in the green region connect ordinal nodes to content DNF and will represent the sequence after learning. These plastic synapses are depressed (low) at initialisation. The plastic synapses are read-out by activating them sequentially and observing activity of the post-synaptic neuron.

280 suppression of the neural activity in the ordinal system, the recall is triggered by an external stimulation of
 281 the first ordinal group (the “go” signal). Transitions between states are initiated by an external stimulation
 282 of the CoS group and can take place at arbitrary moments in time (as can be seen in the “replay” phase
 283 on the raster plot of Fig. 7a). This transitions can be triggered by sensory input that signals the end of an
 284 action, associated with items in the sequence (where action can also be intrinsic, like an attention shift), or
 285 can be generated internally, e.g. by a neuronal “timer”, as was introduced by Duran and Sandamirskaya
 286 (2017). The serial order architecture leads to a sequential activation of the ordinal groups, which, in their
 287 turn, lead to sequential activation of the stored locations on the content population, through the on-chip
 288 plastic synapses.

289 Note that memory groups keep firing until the end of the teaching or replay period, keeping track of the
 290 unfolding sequence. This activity is achieved by strong recurrent connections in the memory groups and
 291 can be used to monitor sequence learning and replay by a higher-level system in a hierarchical sequence
 292 representation architecture (Duran and Sandamirskaya, 2012).

293 Fig. 7b shows plastic synapses on the ROLLS chip after learning. Here, each red dot corresponds to a
 294 potentiated plastic synapse. As mentioned previously, the portion of plastic synapses within the yellow
 295 and orange colored regions were set to be potentiated in order to increase self-excitation in the ordinal and
 296 memory neural populations and did not participate in learning. Plastic synapses in the green region are
 297 the ones connecting the ordinal groups to the content WTA and these synapses are depressed (set to zero)
 298 initially and are potentiated during sequence learning. Note that there is no direct access to the state of
 299 the plastic synapses on the ROLLS chip, thus to create Fig. 7b, the potentiated synapses were read out by
 300 stimulating each plastic synapse and measuring if the stimulation lead to a postsynaptic spike.

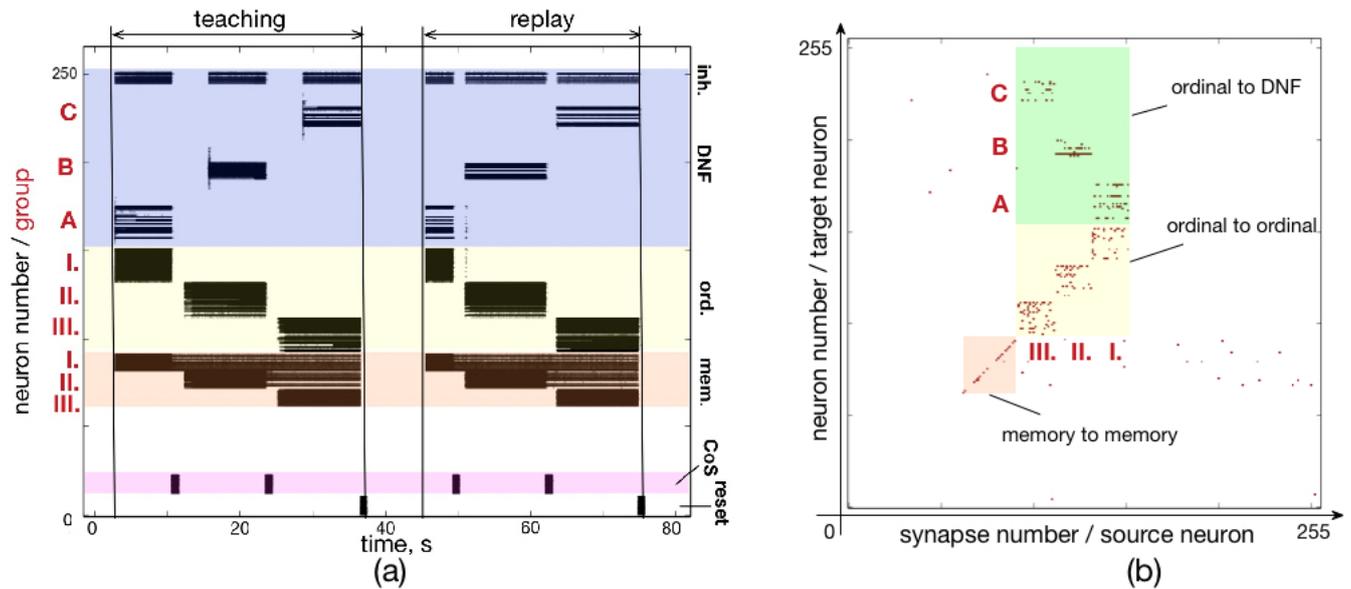


Figure 7. Learning a three-items sequence. (a) Time-course of neuronal spiking activity (raster plot) for the sequence A-B-C. Numbers I., II., III. mark neurons of memory and ordinal populations; letters A, B, C mark three regions on the content DNF; CoS – condition of satisfaction neurons that trigger sequential transitions. In the teaching period, the sequence A, B, C is activated in the content DNF by an external input. In the replay period, this sequence is reproduced by activity flowing from the ordinal nodes to the content DNF through the potentiated plastic synapses. Note that memory nodes stay active until the end of the sequence, when they are inhibited by the Reset population, both during teaching and replay. Active memory nodes represent progress along the sequence. (b) Weights of the plastic synapses on the ROLLS chip after learning the sequence A-B-C. Potentiated synapses in the green region show learned associations from the ordinal groups to the content DNF: I.-A, II.-B, and III.-C.

301 One can observe the potentiated synapses between ordinal group I and item A, ordinal group II and
 302 item B, and ordinal group III and item C. One can also notice a considerable amount of noise in the
 303 plastic synapses (within green region, but also across the chip). Despite this noise, the system is capable to
 304 reproduce the A-B-C sequence, thanks to the WTA dynamics of the content population (seen in the “replay”
 305 part on the raster plot Fig. 7a).

306 Storing a five-items sequence

307 Similar to Fig. 7, Fig. 8 shows how a sequence of five items can be learned and reproduced on the
 308 ROLLS chip. Here, the sequence E-A-B-D-C is stored in the plastic synapses on the ROLLS chip during
 309 the “teaching” period (Fig. 8a) and is reproduced by the chip without external stimulation in the “replay”
 310 period. As for the three-items sequence, the external inputs were used to start both the teaching and the
 311 replay periods and to trigger transitions between sequential elements at arbitrary moments in time (both
 312 during learning and replay). During learning, inputs to the content WTA that correspond to different items
 313 (E-A-B-D-C) were introduced externally (as spike trains with firing rate profile shaped according to a
 314 Gaussian, centered over selected location on the content DNF). During replay, activity in the content DNF
 315 is induced through the potentiated plastic synapses from the ordinal nodes to the content DNF.

316 Fig. 8b shows the plastic weights on the ROLLS chip after learning. Again, potentiated synapses (red
 317 dots) in the yellow and orange regions are just auxiliary weights we used to support recurrent connections
 318 within ordinal and memory groups. Learned synapse are red dots in the green region and correspond to
 319 synapses between ordinal groups (I.-V.) and different regions (A-E) on the content DNF.

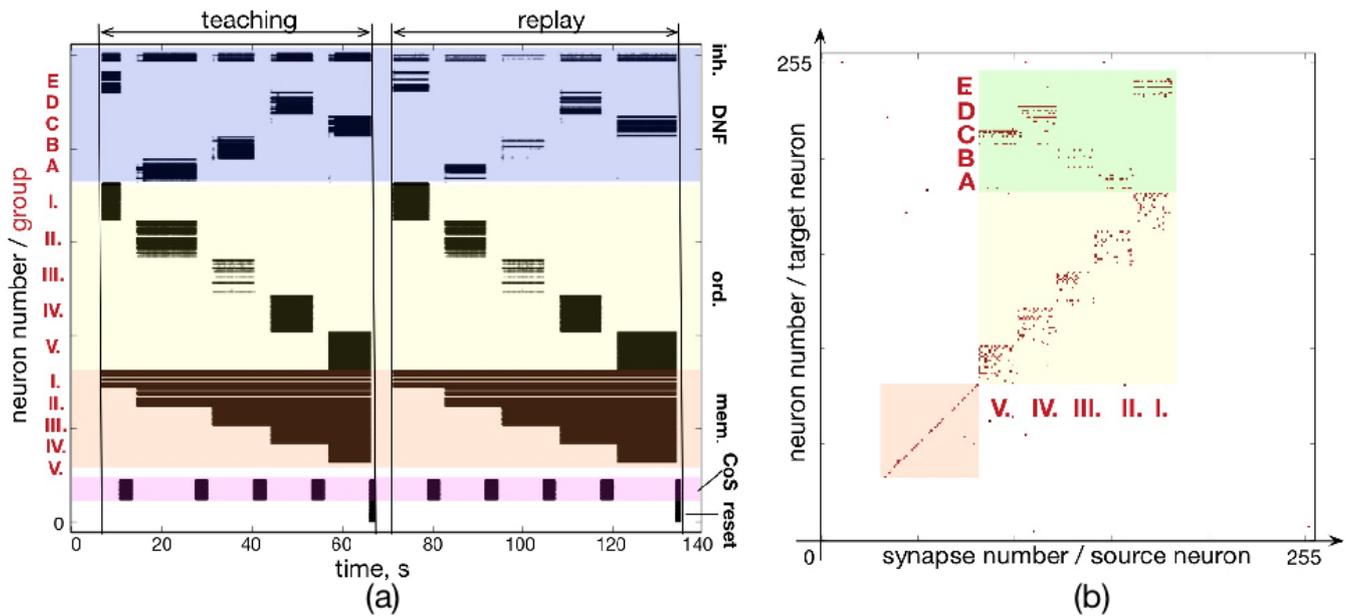


Figure 8. Learning a five-items sequence. (a) Raster plot of the spiking activity on the ROLLS, when the sequence E-A-B-D-C is learned (“teaching”) and reproduced (“replay”). Colors mark different neuronal populations (see also Fig. 7) (b) Weights of plastic synapses after learning the sequence. Plastic synapses in the green region encode the sequence: I.-E; II.-A; III.-B; IV.-D; V.-C

320 Although the learned representation of the sequence is noisy, the system is capable of correctly recalling
 321 the stored sequence. The ordinal nodes activate the locations on the WTA population in the memorized
 322 order, with transitions triggered by the (externally stimulated) CoS population. In the recall session, the
 323 content WTA/DNF does not receive any external stimulation. Activity of neurons in the content DNF is
 324 solely triggered by the formed (learned) associations (potentiated plastic synapses) connecting subsets of
 325 ordinal and content populations.

326 Fig. 9 shows another example of a five-items sequence (A-B-D-E-C) learned on the chip. By using
 327 smaller population sizes, longer sequences can potentially be learned. However, mismatch will become
 328 more noticeable by making the network prone to commit serial order errors.

329 Repeating items

330 Fig. 10 shows that sequences with repeating items can be stored on the chip. Here, a sequence A-A-C is
 331 learned and reproduced on the chip. Often, the serial order architectures, in which sequence is represented
 332 by direct connections between items’ representations, have difficulties with sequences with repeated items.
 333 In our simple 3-items example, for instance, item A would have to be connected both to itself and to the
 334 item C in order to represent the A-A-C sequence. Additional mechanism would be needed to distinguish
 335 between the first and the second occurrence of A. Our serial order architecture features a “positional” (i.e.
 336 spatial) representation of serial order (Sandamirskaya and Schöner, 2010a) and does not suffer from this
 337 problem.

338 As can be seen in Fig. 10b, the plastic weights that connect ordinal group I to element A and ordinal
 339 group II to element A are independent of each other and sequences with arbitrary number of repeating
 340 elements in any position in the sequence can be stored in this way.

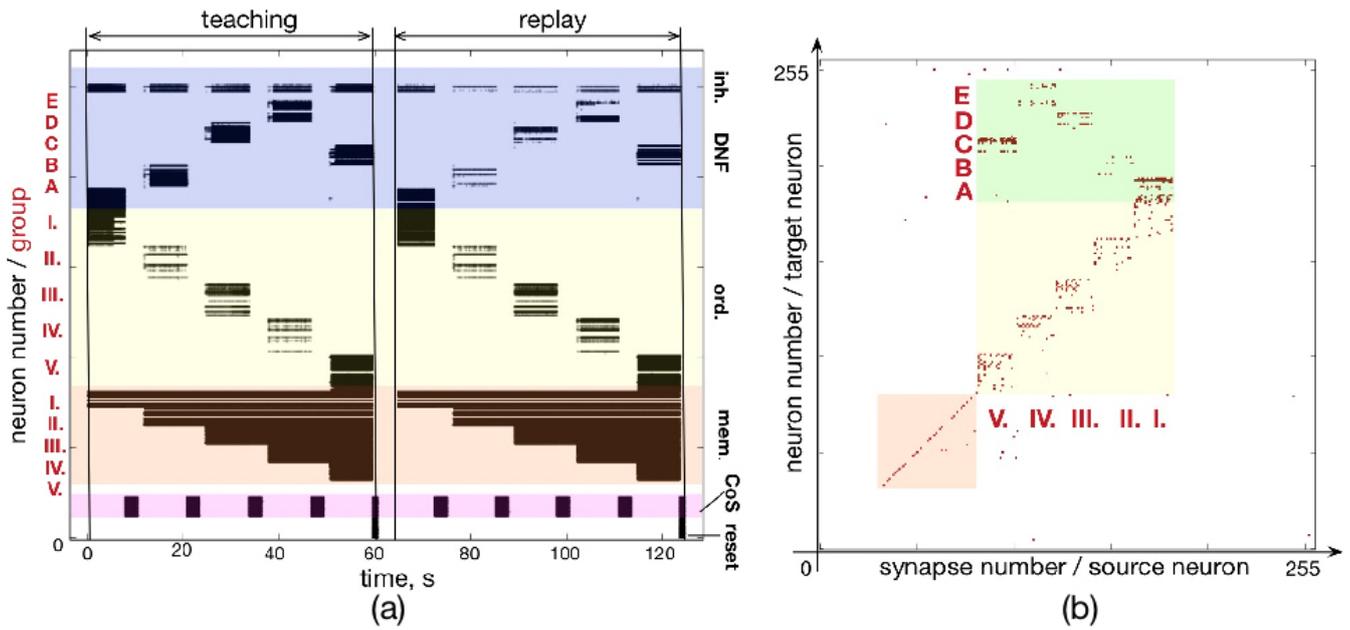


Figure 9. Learning sequence A-B-D-E-C. (a) Time-course of spiking neuronal during learning and recall (similar as Fig. fig:EABDC). (b) Plastic weights after learning the sequence. Weights in the green region encode the sequence by storing the associations between neuronal groups: I.-A; II.-B; III.-D; IV.-E; V.-C.

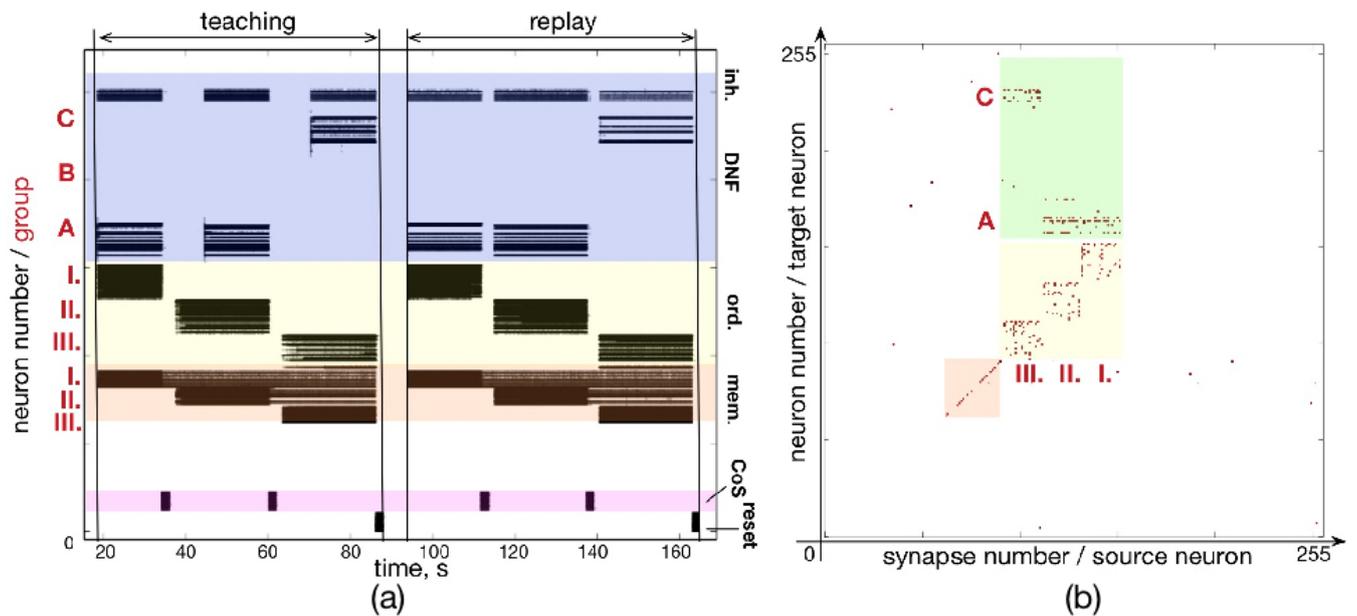


Figure 10. Learning a sequences with repeated items: (a) Time-course of spiking neuronal activity for learning and reproducing the sequence A-A-C. (b) Plastic weights after learning the sequence AAC, storing the sequence: I.-A; II.-B; III.-C.

341 Additional handling for repeated items is often required when sequences are learned in recurrent neural
 342 networks, e.g. (Hochreiter and Schmidhuber, 1997). In our architecture, to the contrary to these networks, all
 343 input representations are stored in the form of independent associations formed between the DNF content
 344 and the ordinal nodes. This feature distinguishes this implementation from other neural architectures
 345 designed to store sequences.

346 **Unlearning a sequence**

347 Another property of the presented serial order architecture is its ability to forget and update previously
348 learned sequences. When a sequence has been learned, activation of an ordinal node induces activation of a
349 certain region on the content DNF, through the plastic synapses. A strong external input to the content DNF,
350 however, can overcome this activation: the winner-take-all connectivity of the content population induces
351 competition between the old item and a new item. If the external input is strong enough to overcome the
352 global inhibition of the content DNF, new activity peak will be created. This new activity peak, in its
353 turn, suppresses the activity induced through the learned plastic synapses. Thus, the synapses that conduct
354 external input to the content DNF need to be stronger than the potentiated plastic synapses for such reset
355 to work (input conducted through plastic synapses can not overcome global inhibition from the strongly
356 supported activity peak, induced by the external input).

357 When a new activity peak is formed in the content DNF, the learning rule leads to potentiation of the
358 plastic synapses leading from the active ordinal node to the newly activated region on the content DNF. At
359 the same time, the synapses to the old item will be slowly “forgotten”, because of the synaptic depression
360 working on synapses with post-synaptic neurons having low (here, zero) firing rate. Thus, the new sequence
361 can be learned, while the old sequence will slowly be forgotten using the synaptic depression of the on-chip
362 learning rule (Brette et al., 2007).

363 Forgetting becomes possible because of the specific learning rule implemented on the ROLLS device:
364 The weight of the plastic synapse is updated upon the arrival of a pre-synaptic spike. Forgetting takes place
365 whenever a pre-synaptic spike is not followed by a post-synaptic spike. In this case, the weight of the
366 plastic synapse gradually decreases and eventually reaches a low (depressed) state.

367 An experiment designed to test the forgetting mechanism is shown in Fig. 11. Fig. 11a shows the firing
368 activity when sequence C - A - B is memorized and recalled. Fig. 11c shows the state of the plastic synapses
369 after learning. After recalling the sequence C - A - B, we stimulated the content DNFs with items in a
370 different order: B - A - C. The firing activity during learning and recalling the new sequence B - A - C is
371 shown in Fig. 11b. The resulting synaptic weight matrix is shown in Fig. 11d.

372 In this figure we can see that the plastic synapses that formed between the first ordinal group and the
373 item C get depressed after stimulating the content neurons with the new sequence B - A - C a single
374 time (compare with Fig. 11c). Instead, the first ordinal group strengthens synapses towards the recently
375 stimulated content neurons B. However, traces of potentiated synapses between content B and ordinal
376 group III remain potentiated after the single trial of learning the new sequence B - A - C. Synapses towards
377 the correct element C in the position III become potentiated gradually.

378 The weight matrices in Fig. 11d show that the previously learned element B remains more consolidated
379 than the newly learned element C. However, Fig. 11b shows that the high activity of the externally
380 stimulated content neurons eventually (after 4 trials) completely suppresses the activity of the previously
381 learned sequence. Hence, during recall the neurons’ firing activity resembles only slightly the old sequence
382 (meant to get forgotten and overwritten) and more strongly represents the new sequence.

383 Fig. 11e shows plastic weights after stimulating the content layer with the items in order B - A - C
384 four times. The new sequence B - A - C is successfully learned and the old sequence C - A - B is almost
385 completely “forgotten”. A small trace of the previous sequence remains, which is suppressed by the WTA
386 dynamics of the content DNF during replay.

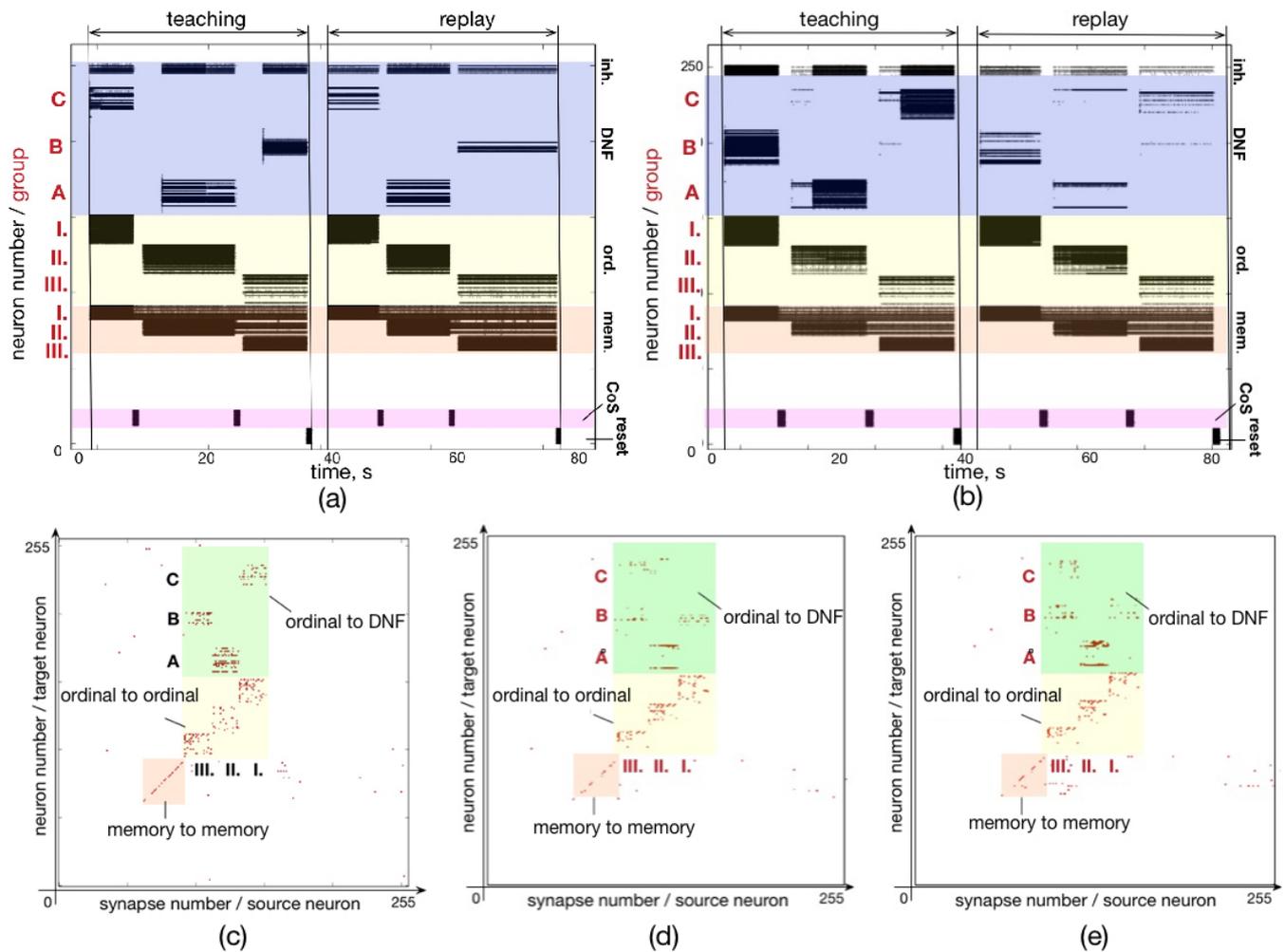


Figure 11. An experiment to demonstrate unlearning: (a) spiking activity on ROLLS during learning and replay of a sequence C-A-B; (b) learning and replay of a new sequence, B-A-C, without resetting the plastic weights; (c) the plastic weights after learning the first sequence (C-A-B); (d) plastic weights after the first trial of learning the second sequence, B-A-C; (e) plastic weights after the fourth trial of learning the second sequence.

387 3.1 Robotic implementation: Learning a sequence of visual stimuli

388 In this section, we present an implementation of the serial order architecture on the neuromorphic chip
 389 interfaced to a robotic platform. Here, the ROLLS processor is configured with with a similar connectivity
 390 as for the experiments with artificial external inputs. The only difference is that we found a configuration,
 391 in which the 30% of randomly potentiated plastic weights used in the ordinal and memory populations to
 392 enhance non-plastic self-excitation was not required (bias parameters, used here are listed in Table 2, Exp.2).
 393 The content DNF population here receives input from a neuromorphic camera Dynamic Vision Sensor
 394 (DVS) (Lichtsteiner et al., 2006; Liu and Delbruck, 2010), mounted on top of a robotic vehicle (Conradt
 395 et al., 2009)¹

396 Each pixel of the eDVS is sensitive to the temporal change in luminance and signals events when such
 397 change exceeds a threshold. The change events are communicated off-sensor using the Address Event
 398 Representation (AER) protocol, typically used for spike-based communication.

¹ <https://inilabs.com/products/pushbot>

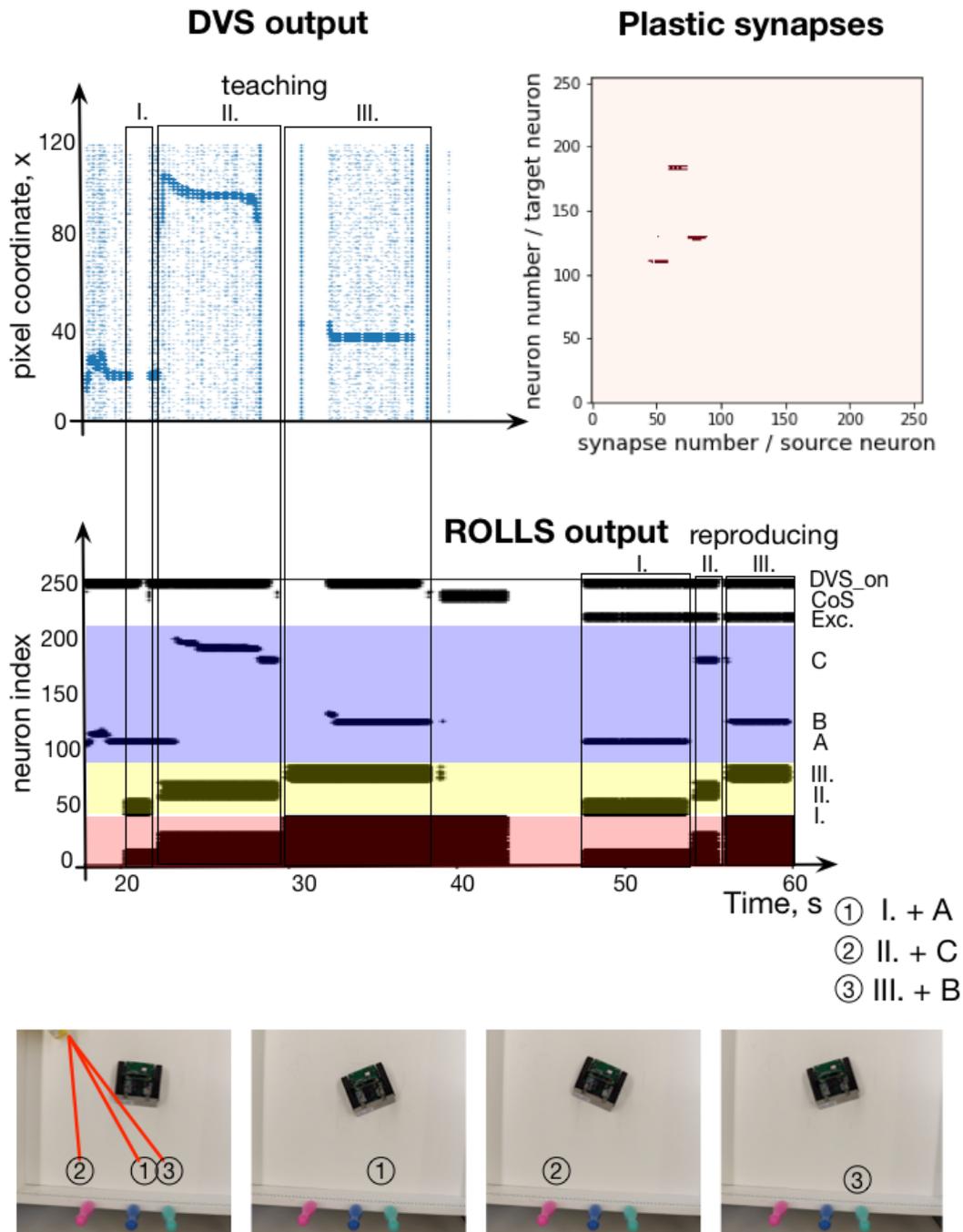


Figure 12. Learning a sequence of cued locations with a robot. **Top, left.** The output of the Dynamic vision Sensor (DVS) camera of the robot: events from rows of the DVS over time. Regions with high activity correspond to horizontal positions of locations, cued with a laser pointer. **Top, right.** Plastic synapses after learning. Dark red dots are synapses with high weights (only synapses from ordinal populations to the content DNF are probed here). **Middle** Spiking activity of neurons on the ROLLS chip during the robotic sequence learning experiment, in which sequence of three locations was learned (A-C-B) and reproduced by turning to center respective location in the field of view of the robot’s DVS (the mapping from position in the camera’s FoV and angle of rotation was hard-coded here for simplicity). **Bottom.** Snapshots of the experiment from an overhead camera. See main text for details.

399 The DVS sends events directly as input spikes to the virtual synapses of the content neurons. Each event
400 is a tuple of pixel coordinates (x,y) and polarity (pol), which signals whether the detected brightness change
401 is positive or negative (we don't distinguish between events with different polarities here). Each DVS event
402 triggers stimulation of a neuron in the content DNF according to the x coordinate of the active pixel². As a
403 sensory stimulus, we used a blinking laser pointer, highlighting different points of the scene, perceived by
404 the robot. The reflection of the laser pointer generates a large number of DVS events that allow the content
405 WTA to easily filter out sensory noise and form activity bump over the highlighted region. This allows us
406 to avoid any sophisticated visual preprocessing.

407 Observing the highlighted region at different positions in the DVS's field of view creates activity at
408 different locations of the content WTA on ROLLS. Simultaneous high activity in the content DNF
409 population and an active ordinal population lead to potentiation of plastic synapses between these
410 populations during sequence learning. During sequence recall, activity of the ordinal populations is
411 transmitted to the content DNF population over the potentiated synapses for each sequential item. The
412 active regions of the content WTA are read-out to set the turning angles for the robot (we hard-coded this
413 mapping here for simplicity, although DNF-based navigation principles could be used to drive robot's
414 movement more directly, as we have demonstrated previously (Milde et al., 2017b; Blum et al., 2017;
415 Milde et al., 2017c)). Active content WTA neurons initiate turning to the stored directions in the order of
416 the learned sequence.

417 The experimental setup is shown in Fig. 12, bottom. The pushbot learns the sequence of angular locations
418 of the highlighted regions. The middle plot on the Fig. 12 shows the resulting spiking activity of this
419 experiment. During learning, the content WTA neurons are activated by DVS events shown in Fig. 12, top,
420 left. This plot shows the DVS events from each column of the sensor over time during the learning phase.
421 The most salient input is amplified (neurons activated by the laser pointer reflection), whereas noise is
422 suppressed due to the global inhibition in the DNF / WTA network.

423 In this experimental example, sequence transition is initiated by activation of the CoS group when the
424 laser pointer is switched off before it is moved to the new position. This creates a clear sequential structure
425 of the task. With a more sophisticated vision processing available, transitions could be triggered by the
426 lack of an overlap of the highlighted region and segmented object when the laser pointer is moved from
427 one object to another one without switching it off.

428 Whenever the laser pointer is switched off, the CoS population is activated. We achieved this by
429 introducing a neural population that is activated by the strong input from the DVS (irrespective of its
430 position), DVS_on population in Fig. 12. This population inhibits the CoS population, which is otherwise
431 activated by the output of the content DNF. When the laser pointer is switched off, the DVS_on population
432 yields its activation and the CoS is activated briefly. The CoS population suppresses all ordinal nodes until
433 the laser is turned on again, activating the DVS_on population. The next ordinal group is then activated,
434 driven by the connectivity of the serial order architecture. The ordinal populations are consequently
435 activated in a sequence and strengthen their synapses to different active content neurons.

436 The learning phase is followed by a global reset that suppresses the activity of the memory nodes, which
437 otherwise keep track of the unfolding sequence. A sequence recall is triggered by a "go" signal – external
438 activation of the first ordinal neural population and a new population, which sustains its activity through
439 the whole experiments and excites the DVS_on population in order to suppress the CoS signal. The target
440 positions, stored in plastic synapses during learning, are reproduced and drive the robot to turn towards

² We drop 80% of events randomly to further reduce events' flow without loss in performance for the simple vision task used here.

441 the orientation that puts the cued positions in the center of field of view of the DVS. Once the movement
442 execution towards the recalled orientation is finished, the DVS_{on} population is inhibited externally leading
443 to activation of the CoS signal and thus a transition in the sequence.

444 Fig. 12, top-right shows the read out of the plastic weights after learning the visual sequence. Here,
445 to visualise the strengths of the plastic connections, the ordinal neurons are stimulated one by one and
446 activity of the whole chip is observed. In particular, we can see that three different positions in the content
447 populations get activated when associated ordinal groups are stimulated. This measurement is based on
448 whether or not a spike of a particular ordinal item triggers the firing of a particular content item, which
449 reveals the state of the connecting plastic synapse. We can see that the sequence of three rotations were
450 successfully stored in the plastic connections. Since we focus on the neuronal architecture for storing
451 sequences here rather than on the aspects of the robotic implementation, we did not quantify precision
452 of the sequence reproduction and only accessed it qualitatively (as can be observed in the accompanying
453 video sequence).

4 DISCUSSION

454 In a proof-of-concept demonstration we have shown how sequences can be stored in a mixed signal
455 analog/digital neuromorphic device with on-chip plasticity (Qiao et al., 2015). We have shown that
456 sequences of different length can be learned and updated and how sequence learning can be driven by
457 sensory input from a silicon retina camera DVS.

458 Computing architectures in neuromorphic hardware can process events efficiently, in an inherently
459 event-based manner, in real-time, and with ultra-low power consumption (e.g., the ROLLS device used
460 here consumes <4mW if all its neurons fire at an average frequency of 30Hz). Of course, on the current
461 prototype stage, neuromorphic hardware requires a regular computer (a credit-card size computer Parallella
462 in our case) to connect the device to the robot and monitor its activity, and an FPGA to be configured
463 efficiently, but in a final embedded application, all computation can happen in the device. The main
464 remaining challenge to achieve this are AER interfaces to different sensors and motors, which are currently
465 being developed (Perez-Pena et al., 2015, 2014a). Power saving is of particular importance in small in agile
466 robots, but even for larger robotic systems, in which at the moment motors dominate the power budget,
467 when (spiking) neural networks are used to process multi-modal sensory information and to control the
468 robot, power consumption of their simulation might become substantial for a real-time application.

469 First robotic architectures that deploy neuromorphic controllers have been introduced over the last years
470 using analog (Milde et al., 2017a) and digital (Conradt et al., 2015) neuromorphic devices. Compared to
471 the mixed-signal neuromorphic hardware used here, digital realisations of neuromorphic computing offer
472 more flexibility in terms of neuronal and synaptic models at the cost of increased power consumption and
473 device size. Choosing a well-suited neuromorphic system is highly task-dependent and it has been shown
474 that large digital neuromorphic devices (e.g., the SpiNNaker platform (Furber et al., 2012)) can also be
475 used to control autonomous mobile robots (Conradt et al., 2015). The presented on-chip sequence learning
476 neuromorphic architecture can be realised both in analog and digital neuromorphic devices.

477 The main limitation when using analog hardware is the mismatch in device parameters that leads to noisy
478 and unreliable computing elements – neurons and synapses. We showed how the use of population-based
479 representations (“place-code”) and attractor dynamics of neural fields allow to nevertheless produce reliable
480 behavior with these noisy elements, thus unleashing this ultra-low power of analog neuromorphic hardware
481 for practical use.

Table 1. Biases of the silicon neuron

Parameter	Description	Value
IF_RST	The reset threshold current	1pA
IF_DC	A constant current injected to all neurons	1pA
IF_BUF	Buffer for oscilloscope	10.9nA
IF_ATHR	Neuron's adaptive threshold current	1pA
IF_RFR1	The duration of the neuron's refractory period	1.5nA
IF_RFR2	The duration of a specified neuron's refractory period	1.5nA
IF_AHW	The adaptation time constant	1pA
IIF_AHTAU	Neuron's adaptive Tau	7.4nA
IIF_TAU2	Time constant for specified neurons	22.6pA
IF_TAU1	Time constant for all other neurons	24uA
IF_NMDA	The sensitivity of the neurons	1pA
IF_CASC	The cascade current	1pA
IF_THR	The neuron's firing threshold	280nA

482 The proposed neuromorphic architecture presents a crucial building block for complex “cognitive”
 483 neuromorphic robotic systems, since sequence learning and sequence generation are key to the most basic
 484 robotic tasks, such as map formation (in a simultaneous localisation and mapping task) or production of
 485 motor sequences. Showing that sequences can be learned, updated, and replayed with flexible timing on a
 486 neuromorphic device is thus a crucial stepping stone for neuromorphic cognitive robots and for autonomous
 487 learning using plastic on-board synapses that realise a local learning rule.

488 The particular model for sequence representation, used here (Sandamirskaya and Schöner, 2010a), has
 489 several properties, advantageous for a neuromorphic implementation: First, it allows for flexible timing of
 490 sequential elements during sequence learning and replay, allowing the sensory input signalling completion
 491 of actions to drive sequential transitions. Second, it circumvents problems of some other serial order models
 492 that have to do with ordinal vs. chaining representation of serial order (Henson, 1998). Thus, repeated
 493 items – both on adjacent or distant positions in a sequence – are not a problem for the model. The length
 494 of the sequence can be arbitrary and is limited by the required number of neurons, which grows linearly
 495 with the sequence length. A model can be easily extended to represent hierarchical sequences (Duran and
 496 Sandamirskaya, 2012), sequences of state coming from different modalities (Sandamirskaya and Schöner,
 497 2010b), or sequences with intrinsic timing of transitions (Duran and Sandamirskaya, 2017). Finally, and
 498 most importantly, a sequence here can be learned with a very simple Hebbian learning rule in a fast – one
 499 shot – learning process. The sequence can be refined and corrected by further repetitions, but it can be
 500 replayed already after a single presentation. Thus, we find this model a promising building block for a wide
 501 range of future neuromorphic architectures that require storing sequences of states, e.g., in reinforcement
 502 learning, map formation, imitation learning, or human-robot interaction.

5 APPENDIX

503 5.1 On-chip biases to realise a WTA on the ROLLS chip

CONFLICT OF INTEREST STATEMENT

504 The authors declare that the research was conducted in the absence of any commercial or financial
 505 relationships that could be construed as a potential conflict of interest.

Table 2. Biases for the recurrent synaptic connections

Parameter	Description	Value: Exp.1	Value: Exp.2
WHT_STD	Controls the magnitude of short term depression	1pA	1pA
PWLK	Controls the pulse width of the synaptic current	448.4pA	69.7pA
WHT_INH	Controls the magnitude of the excitatory weight independent when set to value 0	3.8nA	5.5nA
WHT_INH0	Controls the magnitude of the inhibitory current component which is injected into the excitatory DPI when set to value 1	8.7nA	198.4nA
WHT_INH1	Controls the magnitude of the excitatory current component which is injected into the excitatory DPI when set to value 2	199.4nA	336.7nA
WHT_EXC	Controls the magnitude of the excitatory weight independent when set to value 0	110.9nA	689.7pA
WHT_EXC0	Controls the magnitude of the excitatory current component which is injected into the excitatory DPI when set to value 1	291.2nA	3.3uA
WHT_EXC1	Controls the magnitude of the excitatory current component which is injected into the excitatory DPI when set to value 2	1.4nA	1.4nA
DPIE_THR	Controls the threshold of excitatory synapses	33.1nA	5.38nA
DPIE_TAU	Controls the time constant of excitatory synapses	402.2pA	853.7pA
DPII_TAU	Controls the time constant of inhibitory synapses	1pA	20.5pA
DPII_THR	Controls the threshold of inhibitory synapses	262.6nA	33.0nA

AUTHOR CONTRIBUTIONS

506 RK and DA were the main driving force in neuromorphic realisation and conducting experiments with the
 507 robots, NQ and GI provided support with neuromorphic hardware and bias tuning. YS has initiated the
 508 project, provided the main idea, and guided its implementation in neuromorphic hardware. All contributed
 509 to writing.

FUNDING

510 This project was funded by the UZH grant FK-16-106, ZNZ fellowship, SNSF Ambizione grant
 511 (PZOO2.168183), ERC grant “NeuroAgents” (724295), and the ICT grant “NeuRAM3” (687299).

ACKNOWLEDGMENTS

512 We would like to thank Prof. Dr. Jörg Conradt (TU Munich and KTH) for lending us the pushbots and
 513 Julien Martel (INI, ETHZ) for sharing the eDVS drivers.

REFERENCES

- 514 (????). MaRiSc90a.pdf
- 515 Aldridge, J. W. and Berridge, K. C. (1998). Coding of serial order by neostriatal neurons: a “natural action”
 516 approach to movement sequence. *The Journal of neuroscience : the official journal of the Society for*
 517 *Neuroscience* 18, 2777–87
- 518 Aldridge, J. W. and Berridge, K. C. (2003). Basal Ganglia Neural Coding of Natural Action Sequences.
 519 *The Basal Ganglia VI* 54, 65–73. doi:10.1007/978-1-4615-0179-4
- 520 Amari, S. (1977). Dynamics of pattern formation in lateral-inhibition type neural fields. *Biological*
 521 *Cybernetics* 27, 77–87

- 522 Benjamin, B. V., Gao, P., McQuinn, E., Choudhary, S., Chandrasekaran, A. R., Bussat, J.-M., et al. (2014).
523 Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations. *Proceedings*
524 *of the IEEE* 102, 699–716. doi:10.1109/JPROC.2014.2313565
- 525 Bicho, E., Erlhagen, W., Sousa, E., Louro, L., Hipolito, N., Silva, E. C., et al. (2012). The power of
526 prediction: Robots that read intentions. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ*
527 *International Conference on (IEEE)*, 5458–5459
- 528 Blum, H., Dietmüller, A., Milde, M., Conradt, J., Indiveri, G., and Sandamirskaya, Y. (2017). A
529 neuromorphic controller for a robotic vehicle equipped with a dynamic vision sensor. In *Robotics:*
530 *Science and Systems (RSS)*
- 531 Boahen, K. A. (1999). Point-to-Point Connectivity Between Neuromorphic Chips using Address-Events.
532 *Ieee Transactions on Circuits & Systems* 47, 416–434
- 533 Boegerhausen, M., Suter, P., and Liu, S.-C. (2003). Modeling short-term synaptic depression in silicon.
534 *Neural Computation* 15, 331–348
- 535 Brader, J. M., Senn, W., and Fusi, S. (2007). Learning Real-World Stimuli in a Neural Network with
536 Spike-Driven Synaptic Dynamics. *Neural Computation Massachusetts Institute of Technology* 19,
537 2881–2912. doi:10.1162/neco.2007.19.11.2881
- 538 Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., et al. (2007). Simulation of
539 networks of spiking neurons: a review of tools and strategies. *Journal of computational neuroscience* 23,
540 349–398
- 541 Buetfering, C., Allen, K., and Monyer, H. (2014). Parvalbumin interneurons provide grid cell-driven
542 recurrent inhibition in the medial entorhinal cortex. *Nature Neuroscience* 17, 710–718. doi:10.1038/nn.
543 3696
- 544 Carpenter, a. F. (1999). Motor Cortical Encoding of Serial Order in a Context-Recall Task. *Science* 283,
545 1752–1757. doi:10.1126/science.283.5408.1752
- 546 Cartiglia, M., Kreiser, R., and Sandamirskaya, Y. (2018). A neuromorphic approach to path integration: a
547 head direction spiking neural network with visually-driven reset. In *IEEE Symposium for Circuits and*
548 *Systems, ISCAS*
- 549 Chicca, E., Stefanini, F., Bartolozzi, C., and Indivei, G. (2014a). Neuromorphic Electronic Circuits for
550 Building Autonomous Cognitive Systems. *Proceedings of the IEEE* 102, 1367–1388
- 551 Chicca, E., Stefanini, F., Bartolozzi, C., and Indiveri, G. (2014b). Neuromorphic electronic circuits for
552 building autonomous cognitive systems. *Proceedings of the IEEE* 102, 1367–1388. doi:10.1109/JPROC.
553 2014.2313954
- 554 Clower, W. T. and Alexander, G. E. (1998). Movement sequence-related activity reflecting numerical order
555 of components in supplementary and presupplementary motor areas. *Journal of neurophysiology* 80,
556 1562–6
- 557 Conradt, J., Berner, R., Cook, M., and Delbruck, T. (2009). An embedded AER dynamic vision sensor for
558 low-latency pole balancing. *2009 IEEE 12th International Conference on Computer Vision Workshops,*
559 *ICCV Workshops* , 780–785doi:10.1109/ICCVW.2009.5457625
- 560 Conradt, J., Galluppi, F., and Stewart, T. C. (2015). Trainable sensorimotor mapping in a neuromorphic
561 robot. *Robotics and Autonomous Systems* 71, 60–68. doi:10.1016/j.robot.2014.11.004
- 562 Corradi, F. and Indiveri, G. (2009). Real-time classification of complex patterns using spike-based
563 learning in neuromorphic VLSI. *IEEE Transactions on Biomedical Circuits and Systems* 3, 32–42.
564 doi:10.1109/TBCAS.2015.2479256
- 565 Couey, J. J., Witoelar, A., Zhang, S. J., Zheng, K., Ye, J., Dunn, B., et al. (2013). Recurrent inhibitory
566 circuitry as a mechanism for grid formation. *Nature Neuroscience* 16, 318–324. doi:10.1038/nn.3310

- 567 Davies, M., Srinivasa, N., Lin, T.-h., China, G., Cao, Y., Choday, S. H., et al. (2018). Loihi: a
568 Neuromorphic Manycore Processor with On-Chip Learning. *IEEE Micro* 38, 82–99
- 569 Deco, G. and Rolls, E. T. (2005). Sequential memory: a putative neural and synaptic dynamical mechanism.
570 *Journal of cognitive neuroscience* 17, 1–41. doi:10.1162/0898929053124875
- 571 Delbruck, T. and Lichtsteiner, P. (2006). Freeing vision from frames. *The Neuromorphic Engineer* 3, 3–4
- 572 Deroost, N., Kerckhofs, E., Coene, M., Wijnants, G., and Soetens, E. (2006). Learning sequence
573 movements in a homogenous sample of patients with Parkinson’s disease. *Neuropsychologia* 44,
574 1653–62. doi:10.1016/j.neuropsychologia.2006.03.021
- 575 Duran, B. and Sandamirskaya, Y. (2012). Neural Dynamics of Hierarchically Organized Sequences: a
576 Robotic Implementation. In *Proceedings of 2012 IEEE-RAS International Conference on Humanoid
577 Robots (Humanoids)*
- 578 Duran, B. and Sandamirskaya, Y. (2017). Learning temporal intervals in neural dynamics. *IEEE
579 Transactions on Cognitive and Developmental Systems* , 1–14
- 580 Foster, D. J. and Wilson, M. a. (2006). Reverse replay of behavioural sequences in hippocampal place cells
581 during the awake state. *Nature* 440, 680–683. doi:10.1038/nature04587
- 582 Furber, S. B., Lester, D. R., Plana, L. A., Garside, J. D., Painkras, E., Temple, S., et al. (2012). Overview
583 of the SpiNNaker System Architecture. *IEEE Transactions on Computers* 62, 2454–2467
- 584 Gaussier, P. and Zrehen, S. (1994). Navigating with an animal brain: a neural network for landmark
585 identification and navigation. In *Intelligent Vehicles’ 94 Symposium, Proceedings of the (IEEE)*, 399–404
- 586 Gerstner, W. and Kistler, W. M. (2002). *Spiking neuron models: Single neurons, populations, plasticity*
587 (Cambridge university press)
- 588 Glatz, S., Kreiser, R., Martel, J. N. P., Qiao, N., and Sandamirskaya, Y. (2019, submitted). Adaptive
589 motor control and learning in a spiking neural network, fully realised on a mixed-signal analog/digital
590 neuromorphic processor. In *ICRA*
- 591 Grossberg, S. (1988). Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural
592 Networks* 1, 17–61
- 593 Henson, R. N. A. (1998). Short-Term Memory for Serial Order: The Start-End Model. *Cognitive
594 Psychology* , 73–137
- 595 Herrojo Ruiz, M., Brücke, C., Nikulin, V. V., Schneider, G.-H., and Kühn, A. a. (2014). Beta-band
596 amplitude oscillations in the human internal globus pallidus support the encoding of sequence boundaries
597 during initial sensorimotor sequence learning. *NeuroImage* 85, 779–793. doi:10.1016/j.neuroimage.
598 2013.05.085
- 599 Hikosaka, O., Nakamura, K., Sakai, K., and Nakahara, H. (2002). Central mechanisms of motor skill
600 learning. *Current opinion in neurobiology* 12, 217–22
- 601 Hochreiter, S. and Schmidhuber, J. J. (1997). Long short-term memory. *Neural Computation* 9, 1–32.
602 doi:10.1162/neco.1997.9.8.1735
- 603 Ijspeert, A. J. (2008). Central pattern generators for locomotion control in animals and robots: A review.
604 *Neural Networks* 21, 642–653. doi:10.1016/j.neunet.2008.03.014
- 605 Indiveri, G., Chicca, E., and Douglas, R. J. (2009). Artificial Cognitive Systems: From VLSI Networks
606 of Spiking Neurons to Neuromorphic Cognition. *Cognitive Computation* 1, 119–127. doi:10.1007/
607 s12559-008-9003-6
- 608 Indiveri, G., Murer, R., and Kramer, J. (2001). Active vision using an analog VLSI model of selective
609 attention. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on* 48,
610 492–500

- 611 Johnson, J. S., Spencer, J. P., and Schöner, G. (2008). Moving to higher ground: The dynamic field theory
612 and the dynamics of visual cognition. *New Ideas in Psychology* 26, 227–251
- 613 Kreiser, R., Moraitis, T., Sandamirskaya, Y., and Indiveri, G. (2017). On-chip unsupervised learning in
614 winner-take-all networks of spiking neurons. In *Biological Circuits and Systems (BioCAS)*
- 615 Kreiser, R., Pienroj, P., Renner, A., and Sandamirskaya, Y. (2018). Pose estimation and map formation
616 with spiking neural networks: towards neuromorphic slam. In *2018 IEEE/RSJ International Conference
617 on Intelligent Robots and Systems, IROS*
- 618 Krichmar, J. L. and Wagatsuma, H. (2011). *Neuromorphic and brain-based robots*, vol. 233. doi:10.3233/
619 978-1-60750-959-2-209
- 620 Lichtsteiner, P., Posch, C., and Delbruck, T. (2006). A 128 X 128 120db 30mw asynchronous vision sensor
621 that responds to relative intensity change. *2006 IEEE International Solid State Circuits Conference -
622 Digest of Technical Papers*, 2004–2006doi:10.1109/ISSCC.2006.1696265
- 623 Lipinski, J., Sandamirskaya, Y., and Schöner, G. (2009). Behaviorally Flexible Spatial Communication:
624 Robotic Demonstrations of a Neurodynamic Framework. In *KI 2009, Lecture Notes in Artificial
625 Intelligence*, eds. B. Mertsching, M. Hund, and A. Z. (Berlin: Springer-Verlag), vol. 5803, 257–264
- 626 Liu, S.-C. and Delbruck, T. (2010). Neuromorphic sensory systems. *Current opinion in neurobiology* 20,
627 288–95. doi:10.1016/j.conb.2010.03.007
- 628 Milde, M., Blum, H., Dietmüller, A., Sumislawska, D., Conradt, J., Indiveri, G., et al. (2017a). Obstacle
629 avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic
630 processing system. *Frontiers in Neurorobotics* 11. doi:10.3389/fnbot.2017.00028
- 631 Milde, M., Dietmüller, A., Blum, H., Indiveri, G., and Sandamirskaya, Y. (2017b). Obstacle avoidance and
632 target acquisition in mobile robots equipped with neuromorphic sensory-processing systems. In *IEEE
633 International Symposium on Circuits and Systems (ISCAS)*. 1–4
- 634 Milde, M. B., Blum, H., Dietmüller, A., Sumislawska, D., Conradt, J., Indiveri, G., et al. (2017c). Obstacle
635 avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic
636 processing system. *Frontiers in Neurorobotics* 28, 11
- 637 Mitra, S., Fusi, S., and Indiveri, G. (2009). Real-time classification of complex patterns using spike-based
638 learning in neuromorphic VLSI. *IEEE Transactions on Biomedical Circuits and Systems* 3, 32–42.
639 doi:10.1109/TBCAS.2008.2005781
- 640 Moradi, S., Qiao, N., Stefanini, F., and Indiveri, G. (2017). A scalable multi-core architecture with
641 heterogeneous memory structures for Dynamic Neuromorphic Asynchronous Processors (DYNAPs)
- 642 Neftci, E. (2018). Data and Power Efficient Intelligence with Neuromorphic Learning Machines Embedded
643 Learning to Solve the Challenges of Programming Neuromorphic Hardware. *ISCIENCE* 5, 1–15.
644 doi:10.1016/j.isci
- 645 Neftci, E., Binas, J., Rutishauser, U., Chicca, E., Indiveri, G., and Douglas, R. J. (2013). Synthesizing
646 cognition in neuromorphic electronic systems. *Proc Natl Acad Sci U S A* 110, E3468–76. doi:10.1073/
647 pnas.1212083110
- 648 Neftci, E., Chicca, E., Indiveri, G., and Douglas, R. (2011). A Systematic Method for Configuring VLSI
649 Networks of Spiking Neurons. *Neural computation* 23, 2457–2497. doi:10.1162/NECO.a_00182
- 650 Perez-Pena, F., Delbruck, T., Liu, S. C., Linares-Barranco, A., and Jimenez-Moreno, G. (2015). Event-
651 based control system on FPGA applied to the pencil balancer robotic platform. In *Proceedings of 1st
652 International Conference on Event-Based Control, Communication and Signal Processing, EBCCSP
653 2015*. 2, 1–5. doi:10.1109/EBCCSP.2015.7300699

- 654 Perez-Pena, F., Linares-Barranco, A., and Chicca, E. (2014a). An approach to motor control for spike-
655 based neuromorphic robotics. *IEEE 2014 Biomedical Circuits and Systems Conference, BioCAS 2014 -*
656 *Proceedings* , 528–531doi:10.1109/BioCAS.2014.6981779
- 657 Perez-Peña, F., Morgado-Estevez, A., Linares-Barranco, A., Jimenez-Fernandez, A., Gomez-Rodriguez, F.,
658 Jimenez-Moreno, G., et al. (2013). Neuro-inspired spike-based motion: from dynamic vision sensor
659 to robot motor open-loop control through spike-VITE. *Sensors (Basel, Switzerland)* 13, 15805–15832.
660 doi:10.3390/s131115805
- 661 Perez-Pena, F., Morgado-Estevez, A., Serrano-Gotarredona, T., Gomez-Rodriguez, F., Ferrer-Garcia,
662 V., Jimenez-Fernandez, A., et al. (2014b). Spike-based VITE control with dynamic vision sensor
663 applied to an arm robot. *Proceedings - IEEE International Symposium on Circuits and Systems* ,
664 463–466doi:10.1109/ISCAS.2014.6865171
- 665 Pfeifer, R., Lungarella, M., and Iida, F. (2007). Self-organization, embodiment, and biologically inspired
666 robotics. *Science (New York, N.Y.)* 318, 1088–93. doi:10.1126/science.1145803
- 667 Procyk, E., Tanaka, Y. L., and Joseph, J. P. (2000). Anterior cingulate activity during routine and
668 non-routine sequential behaviors in macaques. *Nature Neuroscience* 3, 502–508
- 669 Qiao, N., Mostafa, H., Corradi, F., Osswald, M., Sumislawska, D., Indiveri, G., et al. (2015). A Re-
670 configurable On-line Learning Spiking Neuromorphic Processor comprising 256 neurons and 128K
671 synapses. *Frontiers in neuroscience* 9, 141
- 672 Rabinovich, M., Huerta, R., and Afraimovich, V. (2006). Dynamics of Sequential Decision Making.
673 *Physical Review Letters* 97, 8–11. doi:10.1103/PhysRevLett.97.188103
- 674 Rasche, C. and Hahnloser, R. H. (2001). Silicon synaptic depression. *Biological Cybernetics* 84, 57–62
- 675 Richter, M., Sandamirskaya, Y., and Schöner, G. (2012). A robotic architecture for action selection
676 and behavioral organization inspired by human cognition. In *IEEE/RSJ International Conference on*
677 *Intelligent Robots and Systems, IROS*
- 678 Rutishauser, U. and Douglas, R. J. (2009). State-dependent computation using coupled recurrent networks.
679 *Neural Computation* 21, 478–509
- 680 Sandamirskaya, Y. (2013). Dynamic Neural Fields as a Step Towards Cognitive Neuromorphic
681 Architectures. *Frontiers in Neuroscience* 7, 276
- 682 Sandamirskaya, Y. and Schöner, G. (2010a). An Embodied Account of Serial Order: How Instabilities
683 Drive Sequence Generation. *Neural Netw.* 23, 1164–1179. doi:10.1016/j.neunet.2010.07.012
- 684 Sandamirskaya, Y. and Schöner, G. (2010b). Serial order in an acting system: a multidimensional dynamic
685 neural fields implementation. In *Development and Learning, 2010. ICDL 2010. 9th IEEE International*
686 *Conference on*
- 687 Sandamirskaya, Y., Zibner, S. K. U., Schneegans, S., and Schöner, G. (2013). Using Dynamic Field
688 Theory to extend the embodiment stance toward higher cognition. *New Ideas in Psychology* 31, 322–339.
689 doi:http://dx.doi.org/10.1016/j.newideapsych.2013.01.002
- 690 Schemmel, J., Brüderle, D., Grübl, A., Hock, M., Meier, K., and Millner, S. (2010). A wafer-
691 scale neuromorphic hardware system for large-scale neural modeling. In *ISCAS 2010 - 2010 IEEE*
692 *International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems.* 1947–1950.
693 doi:10.1109/ISCAS.2010.5536970
- 694 Schöner, G. and Spencer, J. P. (eds.) (2015). *Dynamic Thinking: A Primer on Dynamic Field Theory*
695 (Oxford University Press)
- 696 Serrano-Gotarredona, e. a. (2009). CAVIAR: A 45k neuron, 5M synapse, 12G connects/s AER hardware
697 sensory-processing-learning-actuating system for high-speed visual object recognition and tracking.
698 *IEEE Transactions on Neural Networks* 20, 1417–1438. doi:10.1109/TNN.2009.2023653

- 699 Thelen, E. (1995). Time-scale dynamics and the development of an embodied cognition. In *Mind as*
700 *motion: Explorations in the dynamics of cognition*. 69–100
- 701 Wilson, H. R. and Cowan, J. D. (1973). A mathematical theory of the functional dynamics of cortical and
702 thalamic nervous tissue. *Kybernetik* 13, 55–80
- 703 Wolfgang, T. and Jean-pierre, D. (2003). *Dynamical Systems Approach To Cognition, The: Concepts And*
704 *Empirical Paradigms Based On Self-organization, Embodiment, And Coordination Dynamics*, vol. 10
705 (World Scientific)
- 706 Wörgötter, F. and Porr, B. (2005). Temporal sequence learning, prediction, and control: a review of
707 different models and their relation to biological mechanisms. *Neural computation* 17, 245–319. doi:10.
708 1162/0899766053011555
- 709 Xi, A. (2016). Control of the Compass Gait Biped Robot. In *International Conference on Artificial*
710 *Intelligence and Soft Computing* (Springer), 758–769