

# Learning Temporal Intervals in Neural Dynamics

Boris Duran and Yulia Sandamirskaya

**Abstract**—Storing and reproducing temporal intervals is an important component of perception, action generation, and learning. How temporal intervals can be represented in neuronal networks is thus an important research question both in study of biological organisms and artificial neuromorphic systems. Here, we introduce a neural-dynamic computing architecture for learning temporal durations of actions. The architecture uses a Dynamic Neural Fields (DNFs) representation of the elapsed time and a memory trace dynamics to store the experienced action duration. Interconnected dynamical nodes signal beginning of an action, its successful accomplishment, or failure, and activate formation of the memory trace that corresponds to the action's duration. The accumulated memory trace influences the competition between the dynamical nodes in such a way that the failure node gains a competitive advantage earlier if the stored duration is shorter. The model uses neurally-based DNF dynamics and is a process model of how temporal durations may be stored in neural systems, both biological and artificial ones. The focus of this paper is on the mechanism to store and use duration in artificial neuronal systems. The model is validated in closed-loop experiments with a simulated robot.

**Index Terms**—Neural dynamics, dynamic neural fields, memory for duration, learning timing, neurorobotics, neuromorphic engineering.

## I. INTRODUCTION

Temporal duration of actions and events plays a fundamental role in the organisation of complex behaviours for both biological and artificial agents. The representation of how long each motor action, or a perceptual act, usually takes is fundamentally required to coordinate different sensorimotor loops into a smooth behavioural sequence.

This representation is crucial for understanding how biological neuronal system enable action selection, decision making, and learning – processes in which temporal intervals play an essential role. For instance, in the one of the most elementary learning processes in animals – classical conditioning (i.e., learning an association between an arbitrary stimulus and a rewarding or punishing stimulus), the time interval between the stimuli is a critical part of the learned association [1]. The neuronal system of the animal seems to store the temporal interval during learning and uses this representation during behavior generation to anticipate the upcoming stimulus.

Representing typical durations of actions is also important for building artificial agents (e.g., cognitive robots), controlled by artificial neural networks (ANNs). Cognitive agents are required to perform complex behaviors in changing and unpredictable real-world environments, where temporal structure of events can not be preprogrammed, but has to be perceived

and learned by the agent itself. When artificial neural networks are implemented on conventional computers, the central digital clock of the CPU can be used to measure time. However, such CPU-based conventional computing systems are power-hungry and are not well-suited for real-time, energy efficient implementations of ANNs.

Neuromorphic hardware aims to solve limitations of the conventional von Neumann computing architecture [2] and is becoming widespread [3], [4]. The neuromorphic hardware realises asynchronous neuronal computation and abandons the centralized processor clock [5]. In this hardware, the individual units – artificial neurons – compute by communicating with each other asynchronously using spikes or event-representing signals. Storing the behaviorally relevant time intervals becomes a non-trivial issue in such a decentralised system. Thus, the problem of representing time intervals using the neuronal substrate itself, instead of a digital clock, is becoming more urgent for technical systems.

In spite of the importance of representing temporal intervals for behavior generation and learning, the question of how neuronal networks – both biological and artificial – may represent temporal duration, has not been resolved.

Thus, in the studies of duration representation in *biological neural networks*, a single mechanism for representing temporal intervals could not be postulated. It is still debated whether there is a dedicated time-keeping mechanism (a centralised “clock”) or whether time keeping is an inherent property of neuronal computations, involved in other tasks, such as decision making, memory formation, or movement preparation, and is encoded locally [6]. Both distributed and localised mechanisms were hypothesised to underly temporal processing [7] and include delay lines [8], oscillators [9], the network dynamics [10], and short-term synaptic plasticity [11]. Different mechanisms were suggested for time intervals on different time-scales: from millisecond range to coordinate muscle activity during movement generation [12] to seconds and minutes to succeed in more cognitive tasks, like catching a prey or staying on the task as long as necessary to achieve a goal [13], [14]. Several brain regions have been reported to be involved in temporal processing: the cerebellum for the milliseconds range of temporal intervals, basal ganglia for longer durations, as well as the parietal cortex in more cognitive tasks [15], [16], [17].

An important insight into biological mechanisms that represent temporal intervals has been gained from studies that show that temporal learning is specific to the trained interval: the learned interval can be reproduced with a high precision, but not other intervals [18]. Such specificity suggests a spatial substrate for encoding of timing: different neuronal populations seem to be sensitive to different time intervals and thus the identity of an active neuron, or its position in the

Boris Duran is with the Informatics Research Center, University of Skövde, Skövde, Sweden

Yulia Sandamirskaya is with the Institute of Neuroinformatics, University and ETH Zurich, Zurich, Switzerland. E-mail: yulia.sandamirskaya@ini.uzh.ch

network, determines the encoded time interval. Such duration tuning of neuronal populations has been modelled theoretically [19], established in behavioral experiments [20], and has been supported by neurophysiological observations [21], [22].

In this work, we introduce an architecture that explicitly models such a spatial substrate for encoding elapsed time (duration) using dynamic neural fields. A dynamic neural field is a mathematical formalisation of the dynamics of large homogeneous neuronal populations with strong recurrent interactions [23]. The recurrent, or lateral, connections provide these neural models with capability of working memory, which is essential for storing such transient quantities as the time when some event started or ended, and which is lacking in feedforward neural networks. Indeed, in a feedforward neural network, the computation amounts to relaying the input patterns to output patterns through a learned mapping, where a mere passage of time does not alter the representation that is stored in the network's weights according to the externally paced learning algorithm [24].

The recurrent neuronal networks, to the contrary, have a notion of an internal state which may change over time as activation reverberates in the system. Thus, different reservoir computing networks were used previously to store temporal sequences that include states of different duration [25], [26]. However, the duration of actions, along with other parameters, is implicitly encoded in the recurrent neural network's state. The implicit nature of this representation makes it hard for, e.g., the perceptual system of the agent to access the stored value in order to estimate whether the current duration is longer or shorter than a previously experienced one.

In our model, we go beyond such an implicit timing representation and develop a neural-dynamic architecture, in which the content of the neuronal representation – i.e., the amount of time elapsed between two events – is explicit. Dynamic neural fields offer a computational substrate for such explicit representations of behavioral variables, along with a mechanism to detect, store, and compare them. In particular, DNFs are activation functions defined over behavioral dimensions, which can be perceptual (e.g., color, location, orientation), motor (e.g., velocity, pose), or cognitive (e.g., serial order, label, or, in this work, time). From a biological neuronal perspective, DNFs model the activity of recurrently connected neuronal populations, where each population is responsive to the behavioral parameter of interest. The DNFs architectures have been successfully applied to model cognitive behaviors and their development in humans (e.g., spatial and visual working memory, executive control, habit formation, or word learning) [27] and to control artificial embodied cognitive agents (robots) [28], [29].

In this work, we embed the mechanism for representing temporal intervals in the recently developed DNF architecture for behavioral organisation [30], [31]. In this architecture, neuronal states that correspond to actions are realised by a tuple of DNFs which form an elementary behavior (EB). Elementary behaviors are organised in a sequence using a serial order mechanism, which encodes a sequence of elementary behaviors that corresponds to a particular goal [32], [33].

This system for behavioural organisation offers a compu-

tational neurally inspired substrate that allows to study how timing of actions may be represented in neuronal systems, in particular how durations of actions can be acquired, stored, and reused in subsequent behaviours. Our model utilises a neurally-based learning mechanism of memory trace formation for storing and reusing experienced durations of actions [34]. The presented neuronal architecture allows to implement a neuronal model for representing duration in a closed behavioral loop. In particular, the neural architecture learns durations of actions based on the input, obtained from a (simulated) robotic agent, acting in a sensorimotor task. Our model is consistent with neurobiological findings that postulate a “population clock” as a biological mechanism to store durations of actions and perceptual events [22].

## II. METHODS

### A. The Dynamic Neural Fields (DNFs)

This work uses the mathematical and conceptual framework of Dynamic Neural Fields (DNFs) [27], [28], [35], [36], [37]. A DNF approximates activation of a neuronal population with a continuous activation function,  $u(x, t)$ , defined over a behavioral dimension  $x$  (e.g., space, color, orientation, velocity, or elapsed time). The dynamics equation, Eq. (1), determines how activation  $u(x, t)$  of the DNF evolves over time driven by the external inputs  $S(x, t)$ , a negative resting level  $h$ , and recurrent interactions, shaped by a Mexican hat interaction kernel  $\omega(x - x')$  and a sigmoid non-linearity  $f(u(x, t))$ :

$$\tau \dot{u}(x, t) = -u(x, t) + h + \int f(u(x', t)) \omega(x - x') dx' + S(x, t), \quad (1)$$

$$\omega(x - x') = c_{exc} e^{-\frac{(x-x')^2}{2\sigma_{exc}^2}} - c_{inh} e^{-\frac{(x-x')^2}{2\sigma_{inh}^2}}, \quad (2)$$

$$f(u(x, t)) = \frac{1}{1 + e^{-\beta u(x, t)}}. \quad (3)$$

This dynamics has an attractor solution with a shape of a localised activity peak (bump) over the feature dimension,  $x$ . The location of this peak is determined by the external input to the DNF. The shape of the localised activity region, i.e. the width and height of the peak, is determined by the lateral interaction kernel, Eq. (2). The lateral interaction kernel in a DNF is usually symmetrical and homogeneous (the same for all field locations). A typical kernel is modelled as a sum of Gaussians with a short-range excitation (with width  $\sigma_{exc}$  and strength  $c_{exc}$ ) and a long-range inhibition (width  $\sigma_{inh} > \sigma_{exc}$  and strength  $c_{inh} < c_{exc}$ ). The sigmoidal non-linearity, Eq. (3), shapes the output of the DNF in such a way that only sufficiently activated field locations contribute to the neural interactions in the DNF and between different DNFs;  $\beta$  determines the slope of the sigmoid.

Since we will use a number of DNFs with the same dynamics when describing the duration-learning model, we introduce a notation  $\mathcal{F}(u(x, t))$ , Eq. (4), which stands for the first three terms of Eq. (1): the stabilizing term  $-u(x, t)$ , the resting level  $h$ , and the lateral interactions term (the convolution term).

$$\begin{aligned} \mathcal{F}(u(x, t)) &= -u(x, t) + h + \\ &+ \int f(u(x', t))\omega(x - x')dx'. \end{aligned} \quad (4)$$

These three terms are the same for all DNFs, used in our model (with different parameters marked by sub- and superscripts, e.g.  $h_u$  or  $w^v$ ). In case of zero-dimensional DNFs (dynamic neural “nodes”), the lateral interaction term amounts to self-excitation of the node,  $cf(u(t))$ , with a scaling parameter  $c$ .

The learning mechanism that will be used in this work is the memory trace dynamics, Eq. (5) [34], which has been developed to account for accumulation of motor memory in classical experiments on preservative reaching task (the “A not B” task) [38], [39] and has been used in the DNF framework to represent objects in a neurally inspired object recognition system [40], as well as for modelling category formation [41] and movement preparation [42]. The memory trace forms a low-pass filtered (in time and in space) “copy” of the suprathreshold activity of a DNF. Over several experiences of the active DNF states, these “copies” are accumulated (summed up). The memory trace layer couples back to the DNF as an additive input. In biological neural networks, the memory trace can be interpreted either as a model for strengthening of synaptic connections between a neural population of the DNF and a memory neural population, or as a change of internal neurons’ parameters that increase their excitability.

The memory trace is a linear dynamics without lateral interactions, defined over the same space as the respective DNF and has an attractor at the output of the DNF,  $f(u(x, t))$ :

$$\begin{aligned} \tau^{pr} \dot{P}(x, t) &= \lambda_{build} (-P(x, t) + f(u(x, t))) f(u(x, t)) - \\ &- \lambda_{decay} (1 - f(u(x, t))) P(x, t). \end{aligned} \quad (5)$$

According to Eq. (5), the memory trace,  $P(x, t)$  (also called preshape in the DNF framework), builds-up with a time-constant  $\tau^{pr}/\lambda_{build}$  and decays with a time-constant  $\tau^{pr}/\lambda_{decay}$  on the sites  $x$ , where the DNF  $u(x, t)$  is active ( $f(u(x, t)) > 0$ ) or inactive ( $f(u(x, t)) = 0$ ), respectively.

The memory trace dynamics has some important properties for modelling formation of any kind of long-term memory<sup>1</sup> in DNF architectures. First, because of the spatial low-pass filters (smoothing kernels) in projections between the DNF, the memory trace, and the DNF again, each experienced instance of the behavioral variable is stored as distribution of activation in the memory trace layer. This distribution is centred over the represented value and has a spread, set by the parameters of the neuronal projections. Over multiple experiences of the same value (e.g., multiple trials), the activation distributions are summed-up in the memory trace layer according to Eq. (5). If the individual experiences of the stored value overlap (the variance in the perceived value is low), the distributions (activity bumps) become stronger (larger value over the mean), while keeping the minimal width determined by the neuronal

<sup>1</sup>Long-term meaning here more permanent type of memory than the working memory, but otherwise not specifying the time range of the storage.

interactions. If the variance of the experienced value is large, however, the memory trace becomes wider and less strong. This property plays a crucial role in modelling processes of memory formation and storage with DNFs. Moreover, the spatial precision hypothesis that postulates sharpening of neuronal projections over developmental times [43], [44] can be used to model developmental maturation of the ability to form and use precise long-term memories.

### B. Behavior organisation with DNFs: Elementary Behaviors

A single DNF, as described in the previous section, can represent one value of a behavioral variable as a localised-peak attractor state. In order to represent events extended in time, such as an action that has a beginning and an end, a concept of elementary behaviors (EBs) has been introduced in the DNF framework [31], [30], [33], [34].

The core of an EB are two coupled DNFs: an intention DNF and a condition-of-satisfaction (CoS) DNF (Fig. 1a). The intention DNF holds a representation of the goal’s parameters of the EB and sets attractors for the sensorimotor dynamics that result in a particular overt behavior. Activation of the intention DNF signifies the beginning of an action. The CoS DNF detects when the action is accomplished.

The intention DNF provides a localised input to the CoS DNF, which makes the latter more sensitive to a sensory input that signals a successful accomplishment of the action (this localised input is visible as a subthreshold activity bump in the CoS DNF in Fig. 1a). If the anticipated goal state is achieved, the sensory input to the CoS DNF overlaps with the intention’s input and a suprathreshold localised activity peak emerges in the CoS DNF.

The activated CoS DNF inhibits the respective intention, giving way to the next EB in a behavioral sequence. Sequential switching between the EBs is organised through the CoS and intention “nodes”, or zero-dimensional DNFs (upper part of Fig. 1a). The CoS node integrates the positive activation of the CoS DNF, signalling when an activity peak is formed in this DNF (disregarding the peak’s location). An active intention node, in its turn, sets a localised peak in the intention DNF when an EB is activated, through synaptic connections characteristic for this EB [45].

## III. ARCHITECTURE FOR LEARNING TIMING

### A. Overview of an extended EB

In our previous work, we extended the basic notion of an elementary behavior (EB) with elements that enable learning sequences [46], [30], [33]. These elements include a condition of dissatisfaction (CoD) node, which signals that an EB had to be aborted since its goal could not be reached. In our previous work on reinforcement learning with DNFs, we have used CoD with a fixed timer for every EB [47]. If the execution time of the EB was longer than a fixed maximal time, the CoD was activated and the EB was aborted.

In this work, we introduce a neuro-dynamic mechanism to learn typical durations of actions. The extended EB that enables timed actions is shown in Fig. 1b.

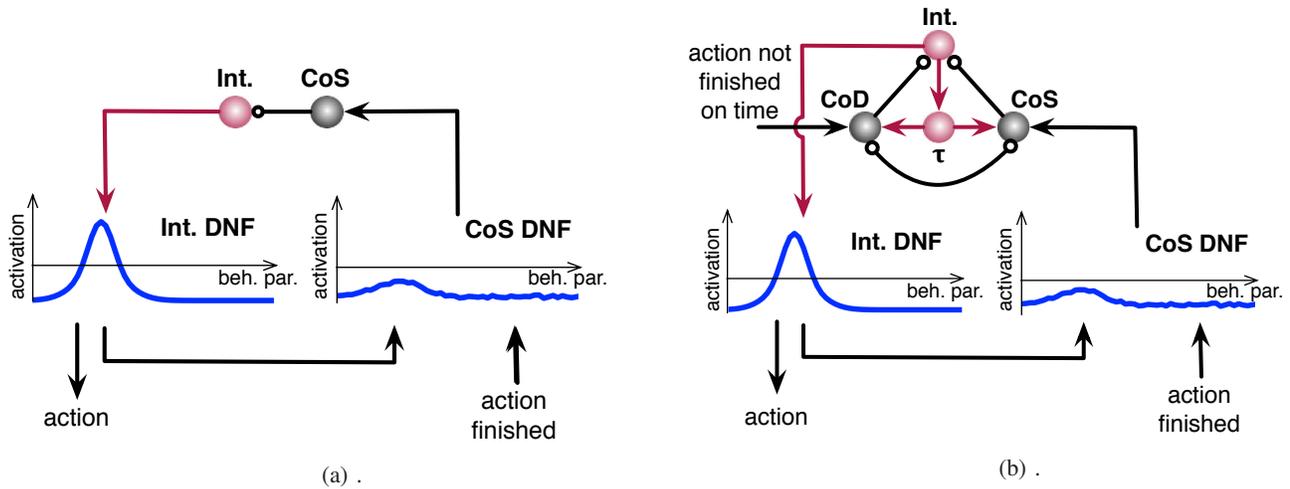


Fig. 1: (a) Elementary behavior (EB) as suggested in [30] consists of intention and condition of satisfaction (CoS) dynamical nodes, connected to respective intention and CoS dynamic neural fields (DNFs). Filled arrows denote excitatory connections, empty circles – inhibitory connections. The CoS DNF is preshaped by input from the intention DNF and can be activated by an external input that matches this preshape (“action finished” input). (b) An extended EB, used in this work: the timing node  $\tau$  is added that has a ramping dynamics, driven by the intention node. The  $\tau$  node drives both the condition of dissatisfaction (CoD) and the CoS nodes, increasingly facilitating activation of the former.

Here, the condition-of-dissatisfaction (CoD) node has the same dynamics as the CoS node and, as the CoS node, inhibits the intention node of the EB when activated (connections ending with a circle in the figure). The CoS and CoD nodes compete with each other through reciprocal inhibition.

Furthermore, a new timing node,  $\tau$ , is introduced. This node has a ramping dynamics, driven by the intention node. The  $\tau$  node impacts the competition between the CoS and CoD node in such a way that over time, the CoD node gains competitive advantage. If the CoS node is not activated by the sensory input, the CoD node wins the competition and signals a failure of the EB to reach the intended outcome in the expected time.

Before learning the action duration, the timing node has a slow dynamics with a large time constant, providing advantage to the CoD node only after a substantial time after activation of the intention node. After learning, the dynamics of this node is adjusted (accelerated) depending on the history of the experienced durations of the same action, acquired and stored in the timing DNF, as described next.

### B. The running peak representation of time

Fig. 2 demonstrates the basic mechanism for storing and using durations of actions in our neural-dynamic architecture. At the core of this architecture, additionally to the EB structure described above, is a timing DNF – a mechanism that we introduced here for the first time to explicitly represent the elapsed time in the DNF framework. This DNF is special in that its interaction kernel is not symmetrical, but is skewed by an ongoing activity of the intention node, shifting the center of mass of the activity peak to the right. This kernel induces a traveling activity peak that moves along the dimension of the Timing DNF with a fixed speed (Fig. 2a). Similar mechanism has been used previously in a robotic controller to generate

smooth movement trajectories with DNFs and to anticipate end-points of movements of an observed human partner [48].

When the action is finished as expected, the CoS DNF and the CoS node are activated by the respective sensory input and inhibit the intention node (Fig. 2b). The interaction kernel of the timing DNF is “unskewed” when the activity of the intention node ceases, becoming symmetrical again. The travelling peak stops at this moment. Now, the activity peak can induce a memory trace growth in the memory trace layer of the timing DNF. A moving peak, to the contrary, cannot induce a large change in the memory trace layer because of its constantly changing location.

The location of the memory trace represents the experienced duration of the action and influences the dynamics of the  $\tau$  node making it to ramp-up faster if the memory trace has stronger activation in its left part (which corresponds to shorter durations). The activity of the  $\tau$  node, in its turn, effects the competitive dynamics between the CoS and CoD nodes (Fig. 2c), giving the CoD a competitive advantage earlier for lower values of durations, represented in the memory trace layer.

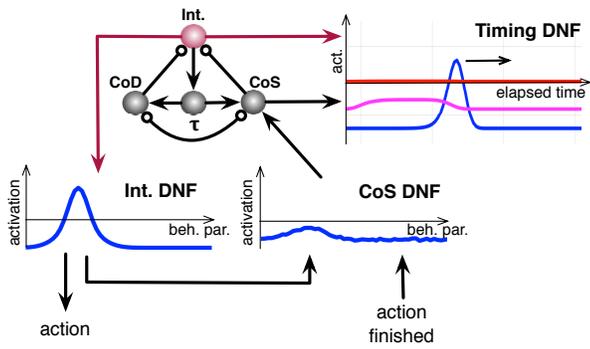
Eqs. (6) show the mathematical expressions for the dynamics of learning the duration:

$$\tau^A \dot{u}^A(x, t) = \mathcal{F}(u^A(x, t)) + I^A(x, t) \quad (6a)$$

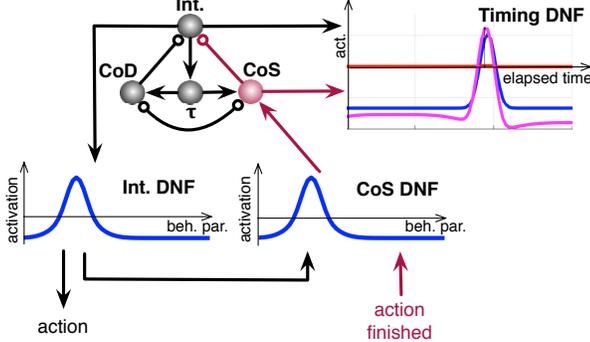
$$\tau^B \dot{u}^B(x, t) = \lambda_{build} (-u^B(x, t) + f(u^A(x, t))) f(u^A(x, t)) - \lambda_{decay} (1 - f(u^A(x, t))) u^B(x, t) \quad (6b)$$

$$\tau^C \dot{u}^C(x, t) = \mathcal{F}(u^C(x, t)) + c^{C,B} \int \omega^{C,B}(x - x') u^B(x', t) dx' \quad (6c)$$

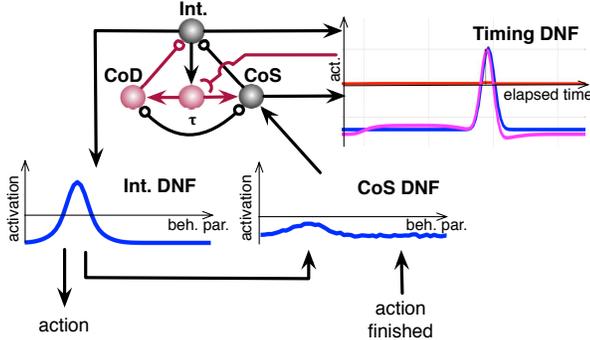
Here,  $u^A$ ,  $u^B$ , and  $u^C$  are the timing DNF (Eq. 6a), its memory trace (Eq. 6b), and the read-out DNF for the memory



(a) An active intention node initiates the action and, at the same time, starts the travelling peak in the Timing DNF.



(b) When the action is accomplished, the condition of satisfaction (CoS) node is activated by the CoS DNF and stops the traveling peak, which now leaves a memory trace at the location, determined by the time interval between activation of the intention and the CoS nodes, thus storing a representation of the action's duration. Location of this memory trace changes the dynamics of the timing node ( $\tau$ ), changing the competitive dynamics between the CoS and CoD nodes on subsequent trials. The active CoS node inhibits the intention node, giving way to the next EB.



(c) If the CoS node is not activated and the travelling peak reaches the location of the previously stored memory trace in the timing DNF, the timing node boosts the CoD node to an extent that it wins the competition with the CoS node and inhibits the intention node to proceed to the next behavior, reporting an error in the current EB.

Fig. 2: The neural-dynamic architecture for storing and using duration of actions.

trace (Eq. 6c), respectively. The  $u^A$  and  $u^C$  follow the generic DNF dynamics, summarised in the term  $\mathcal{F}()$ , as defined by Eq. 4 of Section II-A.  $u^B$  follows the generic memory trace dynamics, Eq. 5, with activity of the timing DNF as input.

The timing DNF receives a localised around zero transient

input  $I^A(x, t)$  from intention node of an active EB, which creates an activity bump in this DNF. The kernel of the timing DNF is an asymmetrical Gaussian kernel (Eq. 7), with the center dynamically shifted by  $\mu_+(t)$  each time the intention node of the EB is activated:

$$w^{A,A}(x, x') = c_+ \exp \left[ -\frac{(x - x' - \mu_+(t))^2}{2\sigma_+^2} \right] \quad (7)$$

$$\mu_+(t) = c_{bias} f(v^i(t))(1 - f(v^s(t))) \quad (8)$$

The kernel's mean,  $\mu_+(t)$ , is dynamically set to zero once the CoS node becomes active, according to Eq. 8, where  $v^i(t)$  and  $v^s(t)$  are activations of the intention and the CoS nodes, respectively. Because of the "skewed" interaction kernel, an activity peak in the timing field, induced by a transient input from the intention DNF, travels with a constant speed along the dimension of the timing DNF. When the CoS node is activated and inhibits the intention node, the kernel is "unskewed", and the peak stops its motion.

The memory trace layer,  $u^B(x, t)$  forms a memory trace of activity in the timing field,  $u^A(x, t)$ . During the time when the peak in the timing DNF is travelling, a very low activity in the memory trace field is created. When that peak stops, however (when a CoS node is activated), the activity in memory trace layer is accumulated at the respective location. In this way, the memory trace stores the value of the temporal interval that has elapsed between activation of the intention and the CoS nodes of the EB. Over several trials, the memory trace merges different experienced duration values [34].

The distribution of the accumulated memory traces in the memory trace layer drives the read-out field,  $u^C(x, t)$ . This DNF receives a "one-to-one" input from the memory trace field,  $u^B(x, t)$ , convolved with a Gaussian projection kernel,  $w^{C,B}(x - x')$  and generates a peak at location of the strongest activation, stored in the memory trace field.

The location of this peak changes the timing constant  $\tau^\tau$  of the dynamics of the timing node,  $v^\tau(t)$ , according to Eqs. 9a and 9b (where  $c$  is a constant and  $v^i(t)$  is activity of the intention node, see full model in the Appendix). This location of the peak is extracted using the attractor dynamics of Eq. 9c:

$$\tau^\tau(t) \dot{v}^\tau(t) = f(v^i(t)) \left( -v^\tau(t) + f(v^i(t)) \right), \quad (9a)$$

$$\tau^\tau(t) = c\phi(t), \quad (9b)$$

$$\tau_\phi \dot{\phi}(t) = -\phi(t) \int f(u^C(x, t)) dx + \int x f(u^C(x, t)) dx. \quad (9c)$$

The  $v^\tau$  node develops a ramping (monotonically increasing) activity if the intention node,  $v^i(t)$ , is activated. Activity  $v^\tau$  of the timing node influences dynamics of the CoS and CoD nodes in a symmetrical way, according to Eqs. 10b and 10c in the Appendix. However, the CoS node's resting level is lower and this node requires an external input that signals accomplishment of the action in order to be activated. Thus, effectively, stronger  $v^\tau$  input provides a competitive advantage to the CoD node over the CoS node.

## IV. SIMULATED ROBOTIC EXPERIMENTS

### A. Specific architecture for the robotic scenario.

For the simulated robotic experiments, we created a setup consisting of five target positions on a two-dimensional space and a mobile robot capable to navigate to these targets while avoiding obstacles.

Fig. 3 shows an instantiation of the proposed model for this scenario. Five EBs were created, each containing an intention, CoS, CoD, and timing ( $\tau$ ) nodes and a timing DNF, its memory trace, and read-out DNF (respectively blue, red, and magenta lines on the plots in each EB in Fig. 3). In the figure, the EBs are arranged according to the locations of targets that they represent (see Fig. 4 for the targets' arrangement).

The location of the respective target is encoded in the connections from the intention node of the EB to the shared intention DNF (lower part of Fig. 3). In the experimental setup reported in this article, the dimension of the intention DNF is the heading direction of the robot in a fixed coordinate frame. Location of an activity bump in the intention DNF defines the direction towards the respective target. This is a place-holder that we used in our simulations for the target acquisition dynamics with DNFs, which can be implemented on a real robot, capable to perceive its targets using feature-based representations [49], [50].

The CoS field receives a localised subthreshold input from the intention DNF and is globally boosted when the robot arrives at a target. Again, in this simulation, we didn't simulate the sensory processes that would lead to activation of the CoS field, but such processes were implemented by us previously with DNFs on real robots [50], [30], [51].

During the experiments described next, the activation flow in the architecture typically proceeds as follows:

- At the beginning of a series of learning trials, the memory trace fields of all EBs are initialised to zero and all time constants of the timing nodes are initialised to a large value (500ms).
- In the first trial, the intention node of the first EB is activated. The active intention node activates a localised region in the intention DNF, which initiates action of the robotic agent and at the same time provides a localised subthreshold input to the CoS DNF; a transient input to the timing DNF from the activated intention node triggers a traveling peak.
- When the CoS DNF and, consequently, the CoS node are activated by the input that signals accomplishment of the current action, the intention node is inhibited and the peak in the timing DNF stops; the memory trace starts building-up in the memory trace layer.
- The CoS node is deactivated when support from the inhibited intention node ceases. The process repeats for the next EB, driven by the higher-level sequence generation dynamics [33], not shown here.
- In the beginning of each learning trial, all nodes and fields are initialised to their resting levels with the exception of the memory trace layer and the value of the time constant of the timing node.

- Over several trials, the activation distributions in the memory trace layers of the EBs represent the distribution of durations of the performed actions. The read-out field selects the maximum of this distribution for each EB and adjusts the integration time of the timing node according to the position of this maximum on the dimension of the timing DNF.

### B. Simulated experiments

The simulated robot was equipped with five range sensors in its frontal part (Fig. 4). The range sensors were used for obstacle avoidance dynamics, providing repelling contributions to the dynamical system controlling the heading direction of the robot if an obstacle was present in the vicinity of the robot (an obstacle was presented on the 4th trial to prolong one of the actions to probe activation of the CoD node).

Target acquisition was realised by setting attractors for the dynamical system controlling the heading direction of the robot (not shown here, but described in detail in [52]) based on the activity peaks in the intention DNF. How this approach to robot navigation is realised is described in work on the attractor dynamics approach to biologically inspired robot navigation [52].

Each of our experiments consisted of five trials, run with the proposed system with a given sequence of targets. Each trial lasted 500 time steps, which was enough for the robot to reach three selected targets. In every trial, the robot was instructed to go first to target #5, then to target #2 and finally to target #4 (Fig. 4a), by setting the connections between the intention nodes of the respective EB and the intention DNF.

In trial 4 (Fig. 4b), an obstacle was added to the scene, which dynamically influenced the trajectory of the robot based on the readings of the range sensors, forcing the robot to perform an obstacle avoidance manoeuvre. This manoeuvre prolonged the first action (going to target #5) on this trial, making this action to take more time than on the other trials.

### C. Results of the simulated experiments

Fig. 5 shows activity of the architecture for EB5 (moving to the fifth target in Fig. 4, which was the first action in the instructed sequence) for two different trials: a "normal" trial 1, when no obstacle was presented on the robot's path (Fig. 4a) and trial 4, when an obstacle was presented (Fig. 4b).

Fig. 5a shows activity in the neuronal architecture on a trial without the obstacle. In the upper plot, activity of the intention, CoS, and CoD nodes is shown. At the point in time, marked with "start of the action", the intention node (blue curve) is activated and its activation is sustained by the self-excitatory connection when the transient task input is removed. The CoS and CoD nodes' activation is ramping slowly driven by the timing node (the green and red curves, respectively), in the subthreshold region. At the moment in time, marked with "end of action", the CoS node is activated by the CoS field, which, in its turn, is driven by the input signalling that the robot has reached the target. This node now inhibits both the intention node and the CoD node; the action is finished.

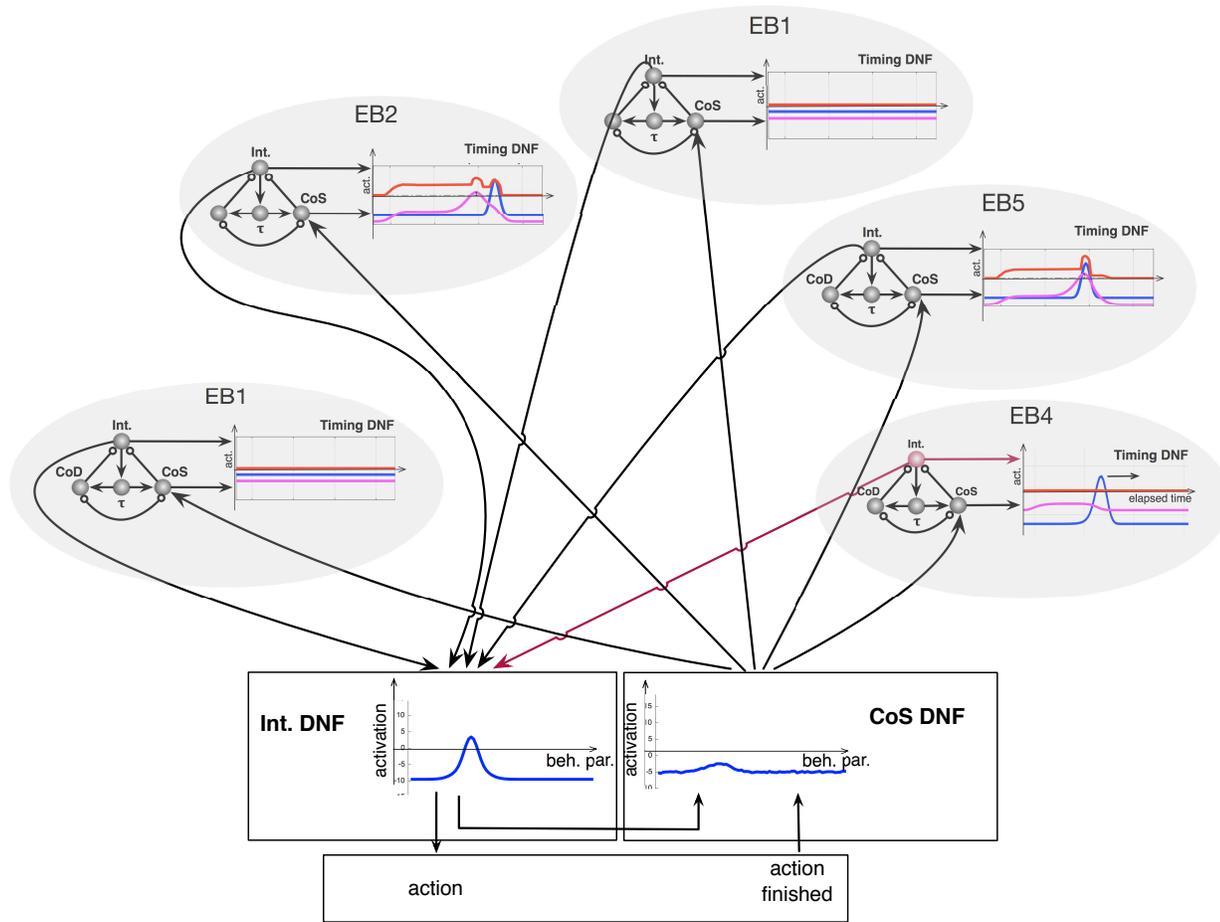


Fig. 3: Overview of the simulated architecture with five EBs.

On the middle plot of Fig. 5a, activation of the Timing DNF is depicted. The vertical axis on this plot shows the behavioral dimension of this DNF, which is, effectively, the temporal duration (the scale is arbitrary since the duration is stored and read out by the same dynamics). The horizontal axis is simulation time (in time steps). The yellow region depicts the localised activity peak that moves with a constant speed along the dimension of duration. This movement stops when the CoS node is activated (“end of action” mark on the plot). In this moment, the memory trace activation starts to build-up (lower plot of Fig. 5a, yellow region). Only weak memory trace was left up to this point by the moving peak.

Fig. 5b shows the same dynamical nodes and fields of the EB5 during trial 4, in which an obstacle is present in the environment. The robot performs an obstacle avoidance maneuver, which makes the EB5 to take longer than on the three previous trials. On the upper plot, the ramping subthreshold activity of the CoS and CoD nodes is faster now, after the robot has learned the usual duration of the EB5. The CoD node thus reaches the activation threshold shortly after the usual action time elapses. This node inhibits the intention and the CoS nodes and aborts the action of the EB5. The timing DNF shows that the moving peak now continues its motion until the CoD node is activated (middle plot). The memory trace is weakened at the location of the previously

learned duration, a weak trace is stored for the longer durations (bottom plot).

Fig. 6 shows the same dynamics as Fig. 5 for EB2 (move to the target 2 in Fig. 4, which is the second action in the instructed sequence) on the trial 1 (Fig. 6a) and trial 4 (Fig. 6b). This action is shorter on trial 4 than on the three previous trials.

Similar as in Fig. 5a, the CoS node is activated in Fig. 6a and the memory trace of activity of the timing DNF is accumulated at location, corresponding to the experienced duration of the EB2’s action. On trial 4, the first action (EB5) was aborted without reaching the goal, consequently, the path to the target #2 is shorter and the EB2 takes less time than on the three previous trials. A new memory trace is formed in the memory trace field (Fig. 6b, bottom) of EB2 that represents an instance of a new, shorter experienced duration. This action is completed on time and the CoS node is activated by the sensory input (top plot).

Fig. 7 shows the snapshots of the memory trace layer,  $u^B(x, t)$ , for three actions of the instructed sequence (EB5, EB2, and EB4) over all five trials (from top to bottom). Note, how for the EB5 (left plot), on the fourth trial (fourth line from the top), the strength of the memory trace is reduced and the center of the memory trace bump is shifted to the right (longer durations). On the other hand, for EB2 (middle

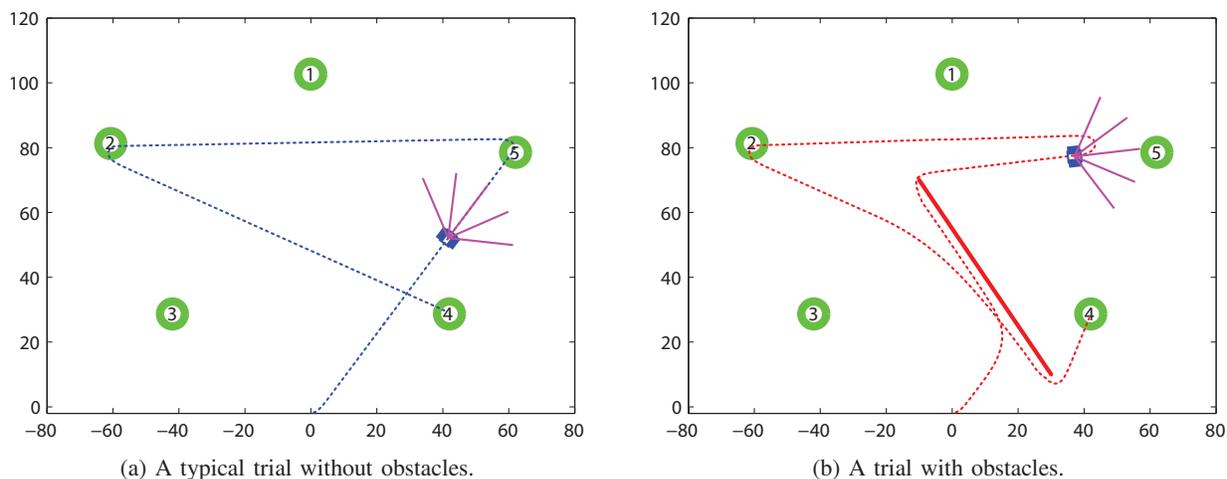


Fig. 4: Snapshot of the path followed by a simulated mobile robot performing a serial order task.

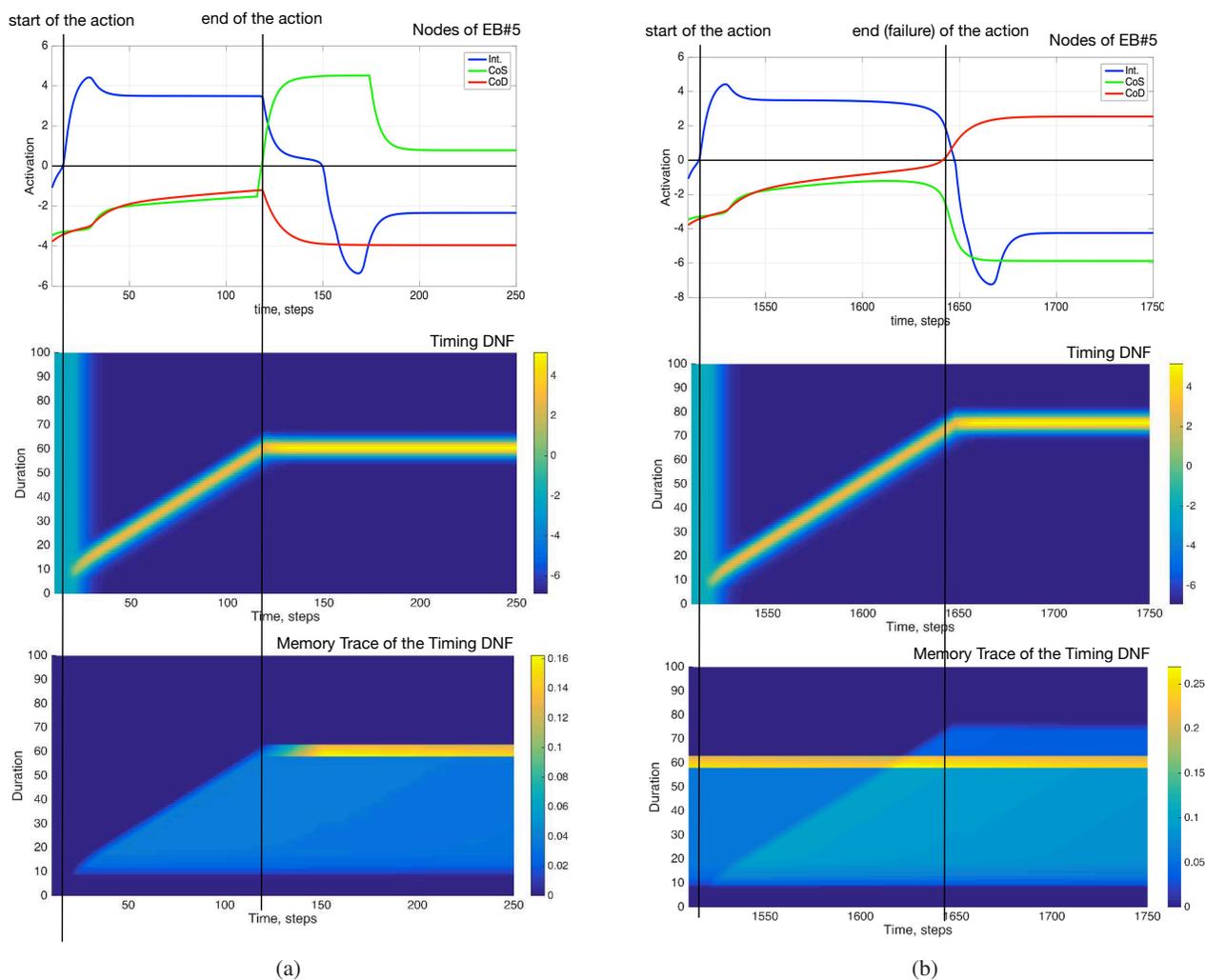


Fig. 5: Activity of the intention, CoS, and CoD nodes (top), the timing DNF (middle), and the memory trace field (bottom) for the EB5 on two different trials. (a) Trial 1: the robot successfully accomplished the action (the CoS node is activated, green line in the top plot), the action's duration is stored in the memory trace of the Timing DNF (bottom plot). (b) Trial 4: an obstacle caused a delay in reaching the target, leading to activation of the CoD node (red line in the top plot) the memory trace is weakened for the previously learned duration of EB5, small activation for a alter time is visible in the bottom plot.

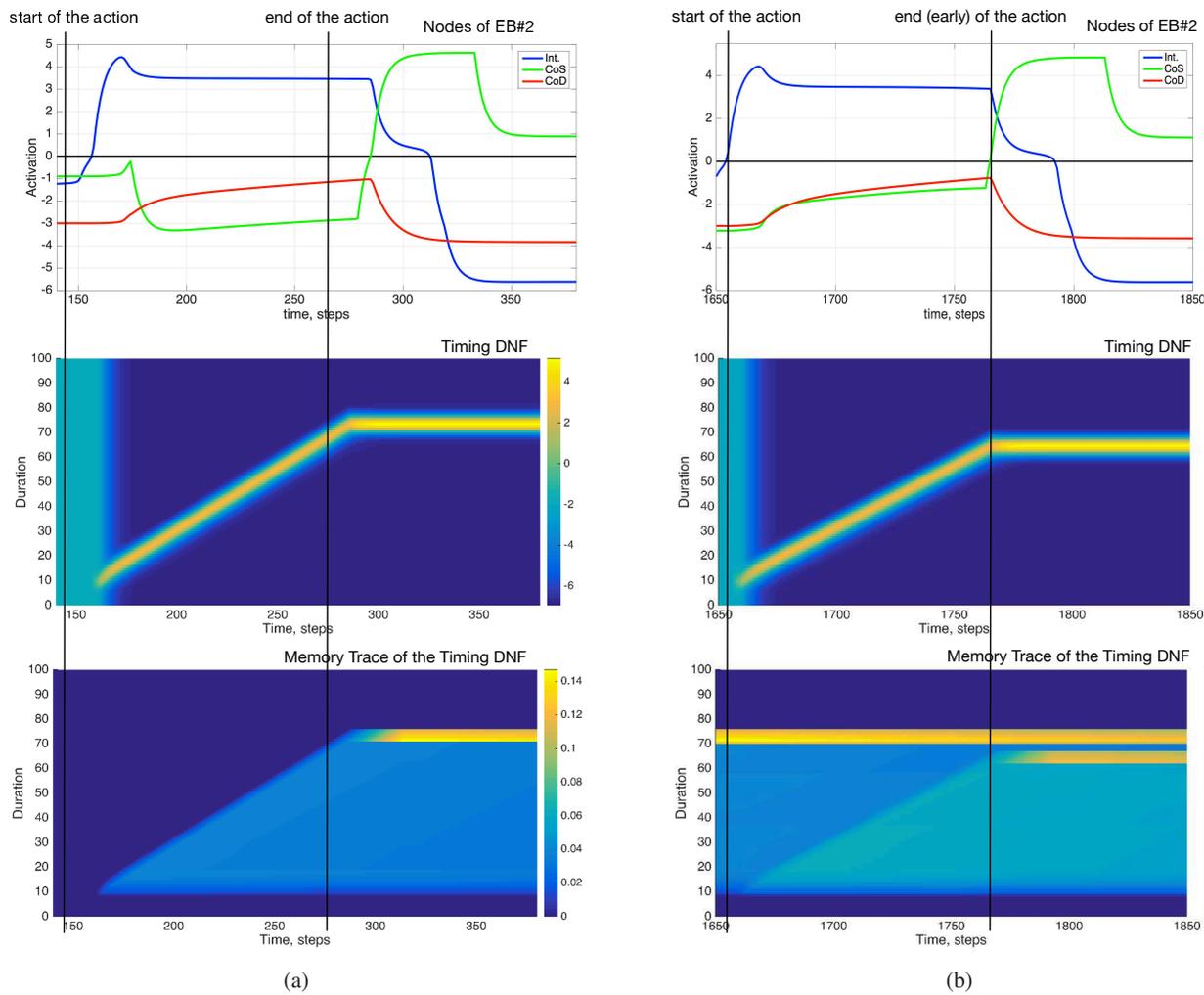


Fig. 6: Activity of the intention, CoS, and CoD nodes (top), the timing DNF (middle), and the memory trace field (bottom) for the EB2 on two different trials. (a) Trial 1: the robot successfully accomplished the action (the CoS node is activated, green line in the top plot), the action's duration is stored in the memory trace of the Timing DNF (bottom plot). (b) Trial 4: abortion of the action of EB5 makes the route to the target #2 shorter; a shorter duration is stored in the memory trace layer of the timing DNF (bottom plot).

plot), the memory trace on trial 4 becomes bi-modal, since a (much) shorter duration was experienced and stored. After only one such experience, the strength of this memory trace won't allow it to be read out by the read-out DNF ( $u^C(x, t)$ ), although because of the noise in the system this might happen. When in trial 5 the usual, longer, duration is experienced again, the strength of the "unusual" memory trace is further reduced (bottom line in the middle plot in Fig. 7).

An example of the evolution of the memory trace for EB5 is shown in Fig. 8. It is possible to see that already after two trials, a well defined peak around  $x \approx 245$  is created. A delayed stable input from the non-moving peak in the timing DNF leaves a memory trace in trial 4 after  $x \approx 245$ , but this experience alone is not strong enough to reshape the stable peak. A new "normal" trial adds activity around the stable peak in the memory trace. The memory trace is continuously updated and one can observe its slow decay from trial to trial (in times when no positive update happens, i.e. other EBs are

active).

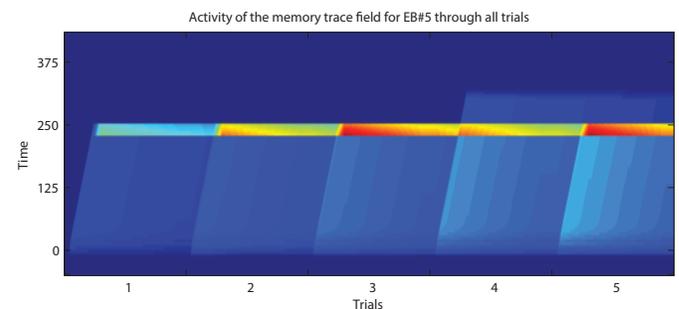


Fig. 8: Activity of memory trace layer for EB5 throughout all trials. Trial 4 shows a decrease in the activity of its maximum due to the memory formation from the delayed reaching of target 5.

Video of a simulated experiment can be seen at <https://www.youtube.com/watch?v=DZ6WuirA4eY>, Matlab

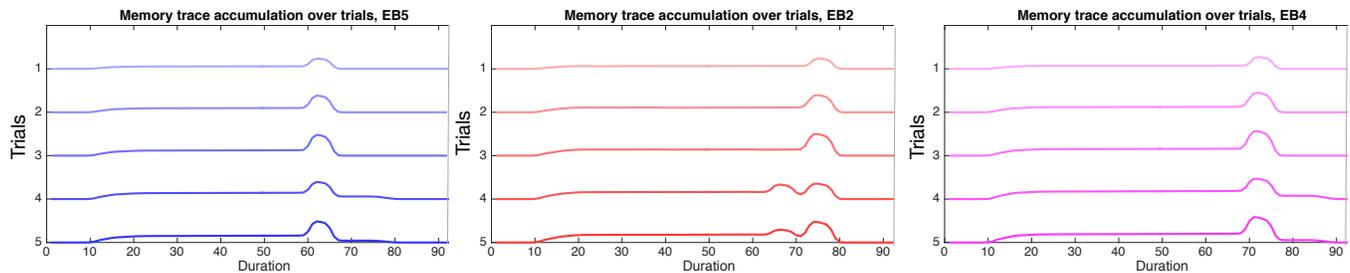


Fig. 7: Accumulation of the memory trace of the Timing DNF over all five trials for three EBs (EB5, EB2, and EB4). See text for details.

code of the experiment can be shared on request.

## V. DISCUSSION

This paper presented a neural-dynamic model for learning durations of actions. The model details the neural-dynamic mechanisms, which are, in our opinion, required to learn durations of actions and to use these learned duration representations to influence behavior of an embodied agent. These mechanisms include: (1) The concept of an elementary behavioral unit, which comprises an intention, condition of satisfaction (CoS) and condition of dissatisfaction (CoD; failure) neuronal populations (“nodes” in our model). In order to measure and store duration of a state, or an action, its beginning and the end need to be detected by the neuronal system. In a complex neuronal architecture, where different neuronal populations are activated and deactivated at every moment in time, detecting the temporal “boarders” of an action is a non-trivial task and such structured representation of the behavioral states is required. (2) A time-representing substrate (a timing DNF in our model), in which time can be measured (by a running activity peak in our model) and the measured durations can be stored (in the memory trace layer) and retrieved (through the read-out DNF). (3) Finally, when the stored duration is read-out, it shall influence the action production. In our model, the simplest impact on the action production is modelled: the system detects when the stored usual duration of an action is exceeded and aborts the action in such case. This influence could take a more sophisticated form, for instance, coordinating different effectors that should accomplish an action at the same time.

The architecture described in this article builds on our previous work on serial order [53], [30], [46] and hierarchical sequences [33] with DNFs. The tools and methodology of DNFs have been used for learning durations here. We have demonstrated the functionality of the model in a simulated experimental setup, in which a mobile robot navigates to a number of pre-defined locations in a two-dimensional space, avoiding an obstacle on one of the trials. This setup demonstrates how durations of a sequence of actions can be stored, updated, and used to detect an unusual situation, in which an action takes longer. This simulated setup can be easily transferred to a physical robot and the model can be implemented in neurally inspired hardware [34] and it can be scaled-up for more complex behavioral scenarios.

We used neurally-based dynamic neural fields as the building blocks of our model and can state that the implemented mechanisms for measuring, storing, and reading-out the temporal intervals can be realised by biological neuronal populations. However, in this work we did not aim to account for any neuronal or behavioral findings, although we believe that our architecture is supported by a number of such findings (briefly reviewed next) and we aim to study biological relevance of our model in the future.

A multitude of studies, e.g. listed in [54], [6], [11], [55], [10], [56] establish that timing of motor responses undergoes learning and adaptation. This is true for temporal intervals in the range of hundreds of milliseconds to a few seconds [57], [12] and means that the variability of a timed action reduces with practice: the requested durations can be reproduced and discriminated with increasing accuracy. The mechanism for storing durations in our architecture – the memory trace dynamics of the DNF framework – is a memory mechanism of a graded nature, in which graded values can be stored together with their uncertainties (expressed in the width of the stored activity distributions). The memory traces are summed up over experiences and have their intrinsic dynamics (i.e., decay). Importantly, the new memory traces interact with memory traces of previous experiences in a non-linear way in the timing DNF: for a short distance (on the dimension of duration) between the memory traces, they are merged while being pulled closer to each other, for a larger distance between the new and the old memory trace, a new memory trace is stored. The accumulated memory traces make activity stronger as more consistent evidence for a given time interval is accumulated. This leads to a more reliable reproduction of durations with training. With inconsistent experience, the memory traces are weak and do not influence the behavior considerably, but may lead to categorisation of the newly perceived temporal intervals. Thus, our model in principle accounts for learning and adaptation of timing. In contrast to more descriptive models of memory formation, e.g. Bayesian inference, our model offers a *neuronal mechanism* for this process.

Our model can be considered an instantiation of a well-accepted principle of encoding time – the state-dependent network (SDN) model [26] – with a particular choice of the population dynamics that alters the neuronal state with passage of time (the assymmetric kernel of the timing DNF). The SDN

model proposes that temporal information is encoded in a neural population with short-term plasticity, in which passage of time changes the system's state, implicitly encoding time along with representations of other stimulus attributes [58], [59], [60], [26]. Another model that proposes that timing may be encoded in dynamically changing patterns of neuronal activity was developed in the context of the cerebellum modelling [61]. In this population encoding model, it was proposed that a stimulus triggers a changing pattern of neural activity as a result of negative feedback in the network [62], [63], [56]. Such timing mechanism has been referred to as a population clock, in which changing populations of active neurons encode time [58], [60]. The shifting activity peak in our timing DNF is a possible interpretation of the population clock representation of the elapsed time, in which the underlying substrate is modelled in a convenient topological fashion, but could be realised by neuronal populations without spatial alignment along the timing dimension.

Another notion, which has considerable experimental support, is that time may be encoded in the linear changes of the firing rate of neuronal population [64]. Such ramping activity has been observed in the neurons in different areas during timing tasks [65], [66], [67], [68], [69], [70], [71]. Ramping dynamics of the timing node in our model reflects this line of neuronal findings.

In hippocampus, neuronal substrate that represents timing in a distributed way has been determined, in which an evolving temporal signal takes the form of a succession of briefly firing neurons, termed "time cells" [72]. These time cells are activated sequentially, irrespective of the content of the task or action and represent the course of time during an action sequence. The behavioral evidence for a distributed neuronal substrate for representing duration was established in studies which postulate that learning effects are duration-selective, i.e., training increases duration sensitivity and reduces performance variability exclusively for the trained interval. This selectivity seems to suggest the existence of neurons tuned to specific temporal intervals and indicate duration-sensitive tuning as a possible mechanism underlying the active encoding of time in the millisecond/second range. We take inspiration from these findings and postulate a timing-specific substrate (the timing DNF) in our model.

Our model cannot be directly compared with the mechanistic biological models at this point, since we use more abstract continuous dynamic neural fields to describe activity of neuronal populations. Computational neuroscience models emphasize importance of timing in neuronal dynamics both on the microscopic scale (the scale on which neuronal plasticity mechanisms, such as spike timing dependent plasticity, operate) and on the behavioral scale [73], [74], [75]. These models show that both using precise timing of neuronal spikes and using rates of neuronal firing, neuronal states can be associated with a delayed reward [76]. Dopamine has been found to play a crucial role both in learning these associations and in learning the timing between the rewarded stimulus and the reward [77]. The latter model is particularly close to our architecture, since it uses rate-based dynamics that is conceptually close to the dynamics of the DNFs and neuronal nodes, used in

our work. The model details several neuronal structures and learning processes, involved in learning associations between the conditioned and unconditioned stimuli that also include learning the duration of the time interval between the stimuli. However, a neuronally implausible "shortcut" is used precisely at this point – the time in the simulation of the model when the first stimulus arrives is stored in a variable ( $t_0$ ), which influences neuronal equations to produce the timing effect (along with a ramping variable and a bank of oscillators, synchronised through the introduced variable  $t_0$ ).

The neuronal-dynamic architecture proposed in this paper focuses on the process of learning timing of neuronal states, which is involved, among other processes, in such reward-related learning. While we don't specify the details of the neuronal mechanisms and brain areas involved, we propose a functional model that uses neurally-plausible dynamics and can be considered a particular instantiation of the SDN model, in which the state of a neuronal population changes with the mere passage of time (shifting the position of activity peak in the timing DNF). Our neural-dynamic architecture shows how such model can be embedded in a closed behavioral loop, driven by sensory events and influencing the dynamics of action initiation. The model can also be realised in neuromorphic hardware, using well-established link between the DNFs and the winner-take-all architecture that is one of the building blocks for cognitive systems in neuromorphic hardware [78], [34].

Our model presents just one of possibly many mechanisms to learn, store, and use temporal intervals during action and perception. The characteristic aspect of this model is that it accounts for the process of learning durations in an *embodied* and *situated* setting, when the system has to detect relevant events and coordinate its dynamics in time based on its own sensory information, perceived in real time. This coordination includes, for instance, temporal organisation on different scales: on the scale of individual behaviors, which have to be initiated and which completion needs to be detected, on the scale of a single behavior for which its timing must be tracked (measured), stored, and used to control behaviors on subsequent instances, as well as for storage of durations of several actions in a row. Timing is only one of possibly many aspects of actions and perceptual states in our architecture, all modelled using DNFs.

## VI. CONCLUSIONS

We presented a neurally-plausible mechanism to learn and store the temporal duration of actions that effectively integrates different experienced durations of the same action, averaging over intervals that are close to each other (making the respective representation broader and thus "less certain") and creating a bi-modal representation if the experienced instances are far apart. This learning mechanism presents a dynamical systems alternative to Bayesian filter and other statistical models of long-term memory formation or creation of representations and will be exploited further in other examples of online sensorimotor learning. Our approach to learning durations can be applied in neuromorphic robotic systems that

require a neuronal timing mechanisms, in order to avoid using a digital clock to measure time.

## REFERENCES

- [1] M. R. Bailey and P. D. Balsam, "Memory reconsolidation: Time to change your mind," *Current Biology*, vol. 23, no. 6, pp. 243–245, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.cub.2013.02.006>
- [2] G. Indiveri and S.-C. Liu, "Memory and information processing in neuromorphic systems," *Proceedings of the IEEE*, vol. 103, no. 8, pp. 1379–1397, 2015.
- [3] S. B. Furber, D. R. Lester, L. A. Plana, J. D. Garside, E. Painkras, S. Temple, and A. D. Brown, "Overview of the SpiNNaker System Architecture," *IEEE Transactions on Computers*, vol. 62, no. 12, pp. 2454–2467, 2012.
- [4] B. V. Benjamin, P. Gao, E. McQuinn, S. Choudhary, A. R. Chandrasekaran, J.-M. Bussat, R. Alvarez-Icaza, J. V. Arthur, P. a. Merolla, and K. Boahen, "Neurogrid: A Mixed-Analog-Digital Multichip System for Large-Scale Neural Simulations," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 699–716, may 2014.
- [5] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, D. Sumislawska, G. Indiveri, and G. Indiveri, "A Re-configurable On-line Learning Spiking Neuromorphic Processor comprising 256 neurons and 128K synapses," *Frontiers in neuroscience*, vol. 9, no. February, 2015.
- [6] D. V. Buonomano and U. R. Karmarkar, "How do we tell time?" *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry*, vol. 8, no. 1, pp. 42–51, 2002.
- [7] R. B. Ivry and J. E. Schlerf, "Dedicated and intrinsic models of time perception," *Trends in cognitive sciences*, vol. 12, no. 7, pp. 273–80, 2008.
- [8] D. W. Tank and J. J. Hopfield, "Neural computation by concentrating information in time." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 84, no. April, pp. 1896–1900, 1987.
- [9] E. W. Large and C. Palmer, "Perceiving temporal regularity in music," *Cognitive Science*, vol. 26, pp. 1–37, 2002.
- [10] A. Goel and D. V. Buonomano, "Timing as an intrinsic property of neural networks: evidence from in vivo and in vitro experiments." *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, vol. 369, no. 1637, p. 20120460, 2014.
- [11] R. Laje, K. Cheng, and D. V. Buonomano, "Learning of Temporal Motor Patterns: An Analysis of Continuous Versus Reset Timing," *Frontiers in Integrative Neuroscience*, vol. 5, no. October, pp. 1–11, 2011.
- [12] J. X. O'Reilly, M. M. Mesulam, and A. C. Nobre, "The cerebellum predicts the timing of perceptual events." *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 28, no. 9, pp. 2252–60, 2008.
- [13] C. V. Buhusi and W. H. Meck, "What makes us tick? Functional and neural mechanisms of interval timing," *Nature Reviews Neuroscience*, vol. 6, no. 10, pp. 755–765, 2005.
- [14] W. H. Meck, "Neuropsychology of timing and time perception." *Brain and cognition*, vol. 58, no. 1, pp. 1–8, jun 2005.
- [15] R. B. Ivry and R. M. C. Spencer, "The neural representation of time." *Current opinion in neurobiology*, vol. 14, no. 2, pp. 225–32, 2004.
- [16] D. L. Harrington, K. Y. Haaland, and R. T. Knight, "Cortical networks underlying mechanisms of time perception." *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 18, no. 3, pp. 1085–95, 1998.
- [17] R. B. Ivry, "The representation of temporal information in perception and motor control," *Current Opinion in Neurobiology*, vol. 6, pp. 851–857, 1996.
- [18] D. Buetti and D. V. Buonomano, "Temporal Perceptual Learning," *Timing & Time Perception*, vol. 2, no. 3, pp. 261–289, 2014.
- [19] R. B. Ivry and T. C. Richardson, "Temporal control and coordination: the multiple timer model," *Brain and Cognition*, vol. 48, pp. 117–132, 2002.
- [20] J. Heron, C. Aaen-Stockdale, J. Hotchkiss, N. W. Roach, P. V. McGraw, and D. Whitaker, "Duration channels mediate human time perception," *Proceedings of the Royal Society B: Biological Sciences*, vol. 279, no. 1729, pp. 690–698, 2012.
- [21] A. Mita, H. Mushiaki, K. Shima, Y. Matsuzaka, and J. Tanji, "Interval time coding by neurons in the presupplementary and supplementary motor areas." *Nature neuroscience*, vol. 12, no. 4, pp. 502–7, 2009.
- [22] D. Z. Jin, N. Fujii, and A. M. Graybiel, "Neural representation of time in cortico-basal ganglia circuits." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, pp. 19 156–19 161, 2009.
- [23] H. R. Wilson and J. D. Cowan, "A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue." *Kybernetik*, vol. 13, no. 2, pp. 55–80, sep 1973.
- [24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [25] H. Jaeger, W. Maass, and J. Principe, "Special issue on echo state networks and liquid state machines," *Neural Networks*, vol. 20, no. 3, pp. 287–289, 2007.
- [26] D. V. Buonomano and W. Maass, "State-dependent computations: spatiotemporal processing in cortical networks," *Nature Reviews Neuroscience*, vol. 10, no. 2, pp. 113–125, 2009.
- [27] G. Schöner and J. P. Spencer, Eds., *Dynamic Field Theory: A Primer on Dynamic Field Theory*. Oxford University Press, 2015. [Online]. Available: <https://books.google.ch/books?id=iLVPAAQBAJ{&}printsec=frontcover{&}hl=de{&}>
- [28] Y. Sandamirskaya, S. K. U. Zibner, S. Schneegans, and G. Schöner, "Using Dynamic Field Theory to extend the embodiment stance toward higher cognition," *New Ideas in Psychology*, vol. 31, no. 3, pp. 322–339, 2013.
- [29] W. Erlhagen and E. Bicho, "The dynamic neural field approach to cognitive robotics." *J Neural Eng*, vol. 3, no. 3, pp. R36–54, 2006.
- [30] M. Richter, Y. Sandamirskaya, and G. Schöner, "A robotic architecture for action selection and behavioral organization inspired by human cognition," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 2012.
- [31] Y. Sandamirskaya, M. Richter, and G. Schöner, "A neural-dynamic architecture for behavioral organization of an embodied agent," in *IEEE International Conference on Development and Learning and on Epigenetic Robotics (ICDL EPIROB 2011)*, 2011.
- [32] Y. Sandamirskaya and G. Schöner, "An embodied account of serial order: How instabilities drive sequence generation," *Neural Networks*, vol. 23, no. 10, pp. 1164–1179, December 2010.
- [33] B. Durán, Y. Sandamirskaya, and G. Schöner, "A dynamic field architecture for the generation of hierarchically organized sequences," in *Artificial Neural Networks and Machine Learning ICANN 2012*, ser. Lecture Notes in Computer Science, A. Villa, W. Duch, P. rdi, F. Masulli, and G. Palm, Eds. Springer Berlin Heidelberg, 2012, vol. 7552, pp. 25–32. [Online]. Available: [http://dx.doi.org/10.1007/978-3-642-33269-2\\_4](http://dx.doi.org/10.1007/978-3-642-33269-2_4)
- [34] Y. Sandamirskaya, "Dynamic Neural Fields as a Step Towards Cognitive Neuromorphic Architectures," *Frontiers in Neuroscience*, vol. 7, p. 276, 2013.
- [35] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures," *Neural Networks*, vol. 1, no. 1, pp. 17–61, jan 1988.
- [36] S. Amari, "Dynamics of pattern formation in lateral-inhibition type neural fields," *Biological Cybernetics*, vol. 27, pp. 77–87, 1977.
- [37] H. R. Wilson and J. D. Cowan, "A mathematical theory of the functional dynamics of cortical and thalamic nervous tissue," *Kybernetik*, vol. 13, pp. 55–80, 1973.
- [38] G. Schöner and E. Thelen, "Using Dynamic Field Theory to Rethink Infant Habituation," *Psychological Review*, vol. 113, no. 2, pp. 273–299, 2006.
- [39] E. Thelen, G. Schöner, C. Scheier, and L. Smith, "The dynamics of embodiment: A field theory of infant perseverative reaching." *Brain and Behavioral Sciences*, vol. 24, pp. 1–33, 2001.
- [40] C. Faubel and G. Schöner, "Fast learning to recognize objects: Dynamic Fields in label-feature spaces," in *Proceedings of the 5th IEEE 2006 International Conference on Development and Learning (ICDL'06)*, 2006.
- [41] C. Wilimzig, S. Schneider, and G. Schöner, "The time course of saccadic decision making: Dynamic Field Theory." *Neural Networks*, vol. 19, pp. 1059–1074, 2006.
- [42] W. Erlhagen and G. Schöner, "Dynamic field theory of movement preparation." *Psychological Review*, vol. 109, no. 3, pp. 545–572, 2002. [Online]. Available: <http://doi.apa.org/getdoi.cfm?doi=10.1037/0033-295X.109.3.545>
- [43] A. R. Schutte, J. P. Spencer, G. Schöner, and I. Neuroinformatik, "Testing the Dynamic Field Theory : Working Memory for Locations Becomes More Spatially Precise Over Development Testing the Dynamic Field Theory : Working Memory for Locations Becomes," 2003.
- [44] V. R. Simmering, A. R. Schutte, and J. P. Spencer, "Generalizing the dynamic field theory of spatial cognition across real and developmental time scales." *Brain research*, vol. 1202, pp. 68–86, apr 2008. [Online]. Available: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2593104{&}tool=pmcentrez>
- [45] M. Richter and Y. Sandamirskaya, "Neural dynamics for behavioral organization of an embodied agent." in *16th International Conference on Cognitive and Neural Systems (ICNS)*, 2012.

- [46] E. Billing, R. Lowe, and Y. Sandamirskaya, "Simultaneous planning and action: Neural-dynamic sequencing of elementary behaviours in robot navigation," *Adaptive Behavior*, pp. 1–22, 2015.
- [47] Kazerounian S., Luciw M., M. Richter, and Y. Sandamirskaya, "Autonomous Reinforcement of Behavioral Sequences in Neural Dynamics," in *Proceedings of the Joint IEEE International Conference on Development and Learning & Epigenetic Robotics (ICDL-EPIROB)*, 2012.
- [48] W. Erihagen, A. Mukovskiy, E. Bicho, G. Panin, C. Kiss, A. Knoll, H. Vanschie, and H. Bekkering, "Goal-directed imitation for robots: A bio-inspired approach to action understanding and skill learning," *Robotics and Autonomous Systems*, vol. 54, no. 5, pp. 353–360, may 2006. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0921889006000157>
- [49] E. Bicho and G. Schöner, "Target position estimation, target acquisition, and obstacle avoidance," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE'97)*. IEEE, Piscataway, NJ, 1997, pp. SS13—SS20.
- [50] Y. Sandamirskaya and G. Schöner, "An Embodied Account of Serial Order: How Instabilities Drive Sequence Generation," *Neural Netw.*, vol. 23, no. 10, pp. 1164–1179, dec 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.neunet.2010.07.012>
- [51] B. Duran, Y. Sandamirskaya, and G. Schöner, "A Dynamic Field Architecture for the Generation of Hierarchically Organized Sequences," in *Artificial Neural Networks and Machine Learning ICANN 2012*, ser. Lecture Notes in Computer Science, A. Villa, W. Duch, P. Érdi, F. Masulli, and G. Palm, Eds. Springer Berlin Heidelberg, 2012, vol. 7552, pp. 25–32.
- [52] E. Bicho, P. Mallet, and G. Schöner, "Target representation on an autonomous vehicle with low-level sensors." *The International Journal of Robotics Research*, vol. 19, no. 5, pp. 424–447, May 2000. [Online]. Available: <http://dx.doi.org/10.1177/02783640022066950>
- [53] Y. Sandamirskaya and G. Schöner, "Serial order in an acting system: a multidimensional dynamic neural fields implementation," in *Development and Learning, 2010. ICDL 2010. 9th IEEE International Conference on*, 2010.
- [54] M. J. Allman and W. H. Meck, "Pathophysiological distortions in time perception and timed performance," *Brain*, vol. 135, no. 3, pp. 656–677, 2012.
- [55] D. V. Buonomano, "The biology of time across different scales," *Nature Chemical Biology*, vol. 3, no. 10, pp. 594–597, 2007.
- [56] J. F. Medina, K. S. Garcia, W. L. Noes, N. M. Taylor, and M. D. Mauk, "Timing Mechanisms in the Cerebellum: Testing Predictions of a Large-Scale Computer Simulation," *Journal of Neuroscience*, vol. 20, no. 14, pp. 5516–5525, 2000.
- [57] S. W. Keele, R. Ivry, U. Mayr, E. Hazeltine, and H. Heuer, "Cognitive and neural architecture of sequence Representation," *Psychological Review*, vol. 110, no. 2, pp. 316–339, 2003.
- [58] U. R. Karmarkar and D. V. Buonomano, "Timing in the Absence of Clocks: Encoding Time in Neural Network States," *Neuron*, vol. 53, no. 3, pp. 427–438, 2007.
- [59] D. V. Buonomano and D. V. Buonomano, "Decoding Temporal Information: A Model Based on Short-Term Synaptic Plasticity," *Journal of Neuroscience*, vol. 20, no. 3, pp. 1129–1141, 2000.
- [60] D. V. Buonomano and R. Laje, "Population clocks: motor timing with neural dynamics," *Trends in Cognitive Sciences*, vol. 14, no. 12, pp. 520–527, 2010. [Online]. Available: <http://dx.doi.org/10.1016/j.tics.2010.09.002>
- [61] J. L. Raymond, S. G. Lisberger, and M. D. Mauk, "The Cerebellum: A neuronal learning machine," *Science*, vol. 272, pp. 1126–1131, 1996.
- [62] T. Yamazaki and S. Nagao, "A computational mechanism for unified gain and timing control in the cerebellum." *PLoS one*, vol. 7, no. 3, p. e33319, jan 2012. [Online]. Available: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3305129>
- [63] M. D. Mauk and N. H. Donegan, "A model of Pavlovian eyelid conditioning based on the synaptic organization of the cerebellum." *Learning & memory (Cold Spring Harbor, N.Y.)*, vol. 4, no. 1, pp. 130–158, 1997.
- [64] D. Durstewitz, "Self-Organizing Neural Integrator Predicts Interval Times through Climbing Activity," *J. Neurosci.*, vol. 23, no. 12, pp. 5342–5353, 2003. [Online]. Available: <http://www.jneurosci.org/cgi/content/abstract/23/12/5342>
- [65] C. D. Brody, A. Hernández, A. Zainos, and R. Romo, "Timing and Neural Encoding of Somatosensory Parametric Working Memory in Macaque Prefrontal Cortex," *Cerebral Cortex*, vol. 13, no. 11, pp. 1196–1207, 2003.
- [66] M. A. Lebedev, J. E. O. Doherty, and M. A. L. Nicolelis, "Decoding of Temporal Intervals From Cortical Ensemble Activity," pp. 166–186, 2008.
- [67] M. I. Leon and M. N. Shadlen, "Exploring the neurophysiology of decisions," *Neuron*, vol. 21, pp. 669–672, 1998.
- [68] B. A. Schneider and G. M. Ghose, "Temporal Production Signals in Parietal Cortex," *PLoS Biology*, vol. 10, no. 10, 2012.
- [69] D. Bullock, J. C. Fiala, and S. Grossberg, "A neural model of timed response learning in the cerebellum," *Neural Networks*, vol. 7, no. 6-7, pp. 1101–1114, 1994.
- [70] S. Grossberg and N. A. Schmajuk, "Neural dynamics of adaptive timing and temporal discrimination during associative learning," *Neural Networks*, vol. 2, no. 2, pp. 79–102, 1989.
- [71] S. Grossberg and J. W. Merrill, "The hippocampus and cerebellum in adaptively timed learning, recognition, and movement," *Journal of Cognitive Neuroscience*, vol. 8, no. 3, pp. 257–277, 1996.
- [72] H. Eichenbaum, "Time cells in the hippocampus: a new dimension for mapping memories," *Nature Reviews Neuroscience*, vol. 15, no. October, pp. 1–13, 2014. [Online]. Available: <http://dx.doi.org/10.1038/nrn3827>
- [73] E. M. Izhikevich, "Pychronization: computation with spikes." *Neural computation*, vol. 18, no. 2, pp. 245–82, 2006. [Online]. Available: <http://math.rice.edu/~mjf8/research/networks/LearningComputation/Izhikevich2006>
- [74] E. M. Izhikevich, J. Jay, H. Drive, and S. Diego, "Solving the Distal Reward Problem through Linkage of STDP and Dopamine Signaling," *Cerebral Cortex*, vol. 17, pp. 2443–2452, 2007.
- [75] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *Journal of Machine Learning Research*, vol. 3, pp. 115–143, 2002.
- [76] A. Soltoggio and J. J. Steil, "Solving the distal reward problem with rare correlations," *Neural Computation*, vol. 25, no. 4, pp. 940–38, 2013.
- [77] J. Vitay and F. H. Hamker, "Timing and expectation of reward: A neuro-computational model of the afferents to the ventral tegmental area," *Frontiers in Neuroinformatics*, vol. 8, no. JAN, pp. 1–25, 2014.
- [78] E. Neftci, J. Binas, U. Rutishauser, E. Chicca, G. Indiveri, and R. J. Douglas, "Synthesizing cognition in neuromorphic electronic systems." *Proc Natl Acad Sci U S A*, vol. 110, no. 37, pp. E3468–76, 2013.
- [79] B. Duran and Y. Sandamirskaya, "Neural Dynamics of Hierarchically Organized Sequences: a Robotic Implementation," in *Proceedings of 2012 IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2012.

## VII. APPENDIX

### A. A formal description of the model

Mathematically, the neural-dynamic architecture, depicted in Fig. 2, can be described by the following set of differential equations: for the dynamics neural nodes, Eqs. 10, and dynamic neural fields, Eqs. 11:

$$\tau^i \dot{v}^i(t) = \mathcal{F}(v^i(t)) - c^{i,s} f(v^s(t)) - c^{i,d} f(v^d(t)) + I^i(t) \quad (10a)$$

$$\tau^s \dot{v}^s(t) = \mathcal{F}(v^s(t)) + c^{s,\tau} f(v^\tau(t)) + c^{s,i} f(v^i(t)) - c^{s,d} f(v^d(t)) + I_{CoS} \quad (10b)$$

$$\tau^d \dot{v}^d(t) = \mathcal{F}(v^d(t)) + c^{d,\tau} f(v^\tau(t)) + c^{d,i} f(v^i(t)) - c^{d,s} f(v^s(t)) \quad (10c)$$

$$\tau^\tau \dot{v}^\tau(t) = f(v^i(t)) (-v^\tau(t) + f(v^i(t))) \quad (10d)$$

The intention node's activation,  $v^i(t)$ , Eq. 10a, follows the neural field attractor dynamics ( $\mathcal{F}$ ) with two inhibitory terms: one from the CoS node ( $v^s(t)$ ) and one from the CoD node ( $v^d(t)$ ).  $I^i(t)$  is an external (motivational) input that comes from other EBs and context ("precondition") nodes of the overall architecture for behavioral organisation [79].

The CoS and CoD neural nodes, in their turn, follow the dynamics of Eq. 10b and Eq. 10c, respectively. These nodes receive a positive input from the intention node,  $v^i(t)$ , and from

the timing node,  $v^\tau(t)$ . When input from the intention node is positive, the growing input of the timing node determines, when the CoS and CoD nodes reach the activation threshold. Both nodes mutually inhibit each other, thus only one of them is eventually activated: either the CoS node, driven by the sensory input that signals a successful accomplishment of the action,  $I_{CoS} = c \int f(u^S(x, t)) dx$ , or the CoD node, which is activated otherwise.

Finally, Eq. 10d describes the dynamics of the timing node,  $v^\tau(t)$  that doesn't have the self-excitatory part of the neural fields dynamics, but follows a simplified memory trace dynamics instead. In particular, the  $v^\tau$  node develops a ramping (monotonically increasing) activity if the intention node,  $v^i(t)$ , is activated. The temporal constant of this dynamics,  $\tau^\tau$  is set to be large in the beginning of a learning session and is influenced by the activity of the timing DNF through its memory trace (Eqs. 9b and 9c).

By modifying the integration time of the timing node,  $\tau^\tau$ , according to Eqs. (9), the system "learns to wait" for a CoS signal and activates a CoD node if the CoS node is not activated in the expected time. In Eq. (9c), the variable  $\phi$  reads out the position of the activity peak in the field  $u^C(x, t)$  (which, in its turn, reads out the memory trace layer): the activity value of the variable  $\phi$  is proportional to the position of the activity peak in the field  $u^C(x, t)$  (from left to right) and thus represent this position by a "rate code". This variable is linearly controlling the time constant of the  $\tau$  node (Eq. (9b)).

The dynamics of the DNFs that constitute the architecture for timing can be described by the following set of equations:

$$\tau^I \dot{u}^I(x, t) = \mathcal{F}(u^I(x, t)) + f(v^i(t)) W^I(x) \quad (11a)$$

$$\begin{aligned} \tau^S \dot{u}^S(x, t) = & \mathcal{F}(u^S(x, t)) + \\ & + c^{S,I} \int w^{S,I}(x - x') f(u^I(x', t)) dx' + I^S(x, t) \end{aligned} \quad (11b)$$

$$\tau^A \dot{u}^A(x, t) = \mathcal{F}(u^A(x, t)) + I^A \quad (11c)$$

$$\begin{aligned} \tau^B \dot{u}^B(x, t) = & \lambda_{build} (-u^B(x, t) + f(u^A(x, t))) f(u^A(x, t)) - \\ & - \lambda_{decay} (1 - f(u^A(x, t))) u^B(x, t) \end{aligned} \quad (11d)$$

$$\begin{aligned} \tau^C \dot{u}^C(x, t) = & \mathcal{F}(u^C(x, t)) + \\ & + c^{C,B} \int \omega^{C,B}(x - x') u^B(x', t) dx' \end{aligned} \quad (11e)$$

Here, the dynamics of the *intention field*,  $u^I(x, t)$  (Eq. 11a) and the *CoS field*,  $u^S(x, t)$  (Eq. 11b) follow the generic DNF equation and are interconnected in an EB structure, as described in [33]. The CoS field receives a "one-to-one" input from the intention field, convolved with a Gaussian kernel,  $w^{S,I}(x - x')$ , and an external, sensory input  $I^S(x, t)$ . The *intention field* receives input from the associated *intention node*,  $v^i(t)$ , through a synaptic weights function,  $W^I(x)$ , that encodes the "content" of the action's intention. The fields  $u^A$ ,  $u^B$ , and  $u^C$  are described in the main text (Section III-B).

### B. Scalar property of timing

Although we don't aim to account for biological findings about learning durations, we would like to demonstrate that the

model can account for one important characteristics, found in many experiments on learning timing of actions and perceived signals. In particular, it is the so called scalar property of timing: the precision of storing (and reproducing) shorter intervals seems to be higher than precision of storing (and reproducing) longer intervals, at least in some range. Fig. 9 shows the dynamics of a running-peak representation of time in a timing DNF. Note that for some time, the peak's width grows. This is the consequence of lateral interaction in the DNF that make the peak "lag behind" its moving front, which is driven by the asymmetrical kernel. Thus, as the peak moves along the DNF dimension, it becomes more diffuse for some time, which is determined by the parameters of the DNF.

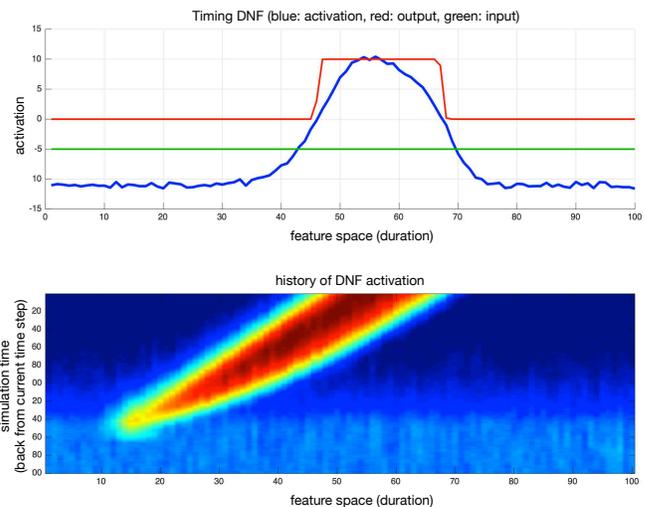


Fig. 9: Widening of the travelling peak in a DNF in the beginning of its propagation as a possible mechanism to account for the scalar property of timing.

### C. The code

Matlab code of the model is freely available at <https://github.com/sandayci/LearningTiming.git>

### ACKNOWLEDGEMENTS

This work was supported by the EU FP7-ICT-2009-6 grant 270247 "NeuralDynamics", the EU H2020-MSCA-IF-2015 grant 707373 "ECogNet", and the UZH Forschungskredit grant FK-16-106.