From memory-based decisions to decision-based movements: A model of interval discrimination followed by action selection

Prashant Joshi

Institute for Theoretical Computer Science, Technische Universität Graz, Inffeldgasse, 16-h/I, A-8010 Graz, Austria

Abstract

The interval discrimination task is a classical experimental paradigm that is employed to study working memory and decision making and typically involves four phases. First, the subject receives a stimulus, then holds it in the working memory, then makes a decision by comparing it with another stimulus and finally acts on this decision, usually by pressing one of the two buttons corresponding to the binary decision. This article demonstrates that simple linear readouts from generic neural microcircuits that send feedback of their activity to the circuit, can be trained using identical learning mechanisms to perform quite separate tasks of decision making and generation of subsequent motor commands. In this sense, the neurocomputational algorithm presented here is able to integrate the four computational stages into a single unified framework. The algorithm is tested using two-interval discrimination and delayed-match-to-sample experimental paradigms as benchmarks.

Keywords: Working memory; Decision making; Action selection; Tuned feedback

1. Introduction

‘Decision making followed by action selection’ is one of the most demanding and recurring event in our day to day lives (Latham & Dayan, 2005). For example, to choose a drink from a vending machine, you need to browse through the possible choices, hold the interesting ones in your working memory, decide which of these choices you want to buy, and finally press the button corresponding to your choice. The interval discrimination task (Machens, Romo, & Brody, 2005; Rainer, Lee, & Logothetis, 2004) is one of the classical experimental paradigms that is employed to study working memory and decision making. The experiment typically involves four phases, viz. the initial loading (L) of the first stimulus, maintaining (M) this stimulus in working memory till the subsequent stimulus is presented, making a binary decision (D), and finally acting (A) on this decision, usually by pressing one of the two buttons corresponding to the binary choice.

The precise computational and biophysical mechanism(s) through which the brain is able to execute this load-maintain-decide-act (LMDA) sequence is not understood. Several theoretical and modelling studies have tried to look at segregated phases of this sequence and give possible explanations for their working (Mensh, Aksay, Lee, Seung, & Tank, 2004; Seung, Lee, Reis, & Tank, 2000b; Singh & Eliasmith, 2006). Typically neurocomputational models of working memory fail to address how the decisions made by neurons in the prefrontal cortex (PFC), are converted into motor commands, which are executed by the sensorimotor system (Durstewitz, Seamans, & Sejnowski, 2000; Machens et al., 2005; Miller & Wang, 2006). Similarly, models for computational motor control tend to ignore the first three phases (LMD), that are responsible for generation of motor commands (Joshi & Maass, 2004, 2005; Todorov, 2000).

Several interesting modelling approaches for tasks involving working memory and decision making have been proposed recently (Machens et al., 2005; Miller, Brody, Romo, & Wang, 2003; Miller & Wang, 2006). These models propose different mechanisms e.g. precise tuning of mutual inhibition (Machens et al., 2005), fine-tuning of a heterogeneous recurrent network (Miller et al., 2003), using an integral feedback signal for inhibitory control (Miller & Wang, 2006), to obtain the persistent neural activity which in turn stores information in the working memory. Despite existing evidence that shows synaptic learning as a responsible mechanism for working
memory related tasks (Rainer et al., 2004), all the models described above use static (no learning involved) neural circuits.

This article proposes a neurocomputational architecture that uses synaptic learning mechanisms (simply linear regression), and is able to integrate the four phases (LMDA) involved in the process of action selection in presence of a decision, into a unified computational framework. Essentially the neural model described here integrates two distinct cortical functions viz., working memory and decision making carried out by the neurons in PFC, and subsequent action selection executed by the sensorimotor system. More precisely, it is demonstrated that delayed-decision tasks that are followed by action selection can be solved, if feedback from trained linear readouts is provided to generic neural microcircuits whose internal dynamics have not been optimized for any particular computational task. Two classical experimental paradigms for interval-discrimination task are modeled using different mechanisms for encoding external sensory inputs. For comparison with earlier models of working memory, the unified framework is used to build a spiking neural network model of two-interval discrimination (Machens et al., 2005). Additionally, to demonstrate that this computational paradigm is task-independent, robust to how the external sensory inputs are encoded, and is capable of integrating the A phase, another spiking neural network model is presented for the delayed match-to-sample task (Rainer et al., 2004), followed by an arm movement to the decided goal position.

The core principles behind the working of this model make the assumption that the cortex can be thought of as an ultra-high dimensional dynamic system, where the afferent inputs arriving from the thalamus and the recurrent cortical feedbacks are churned in a nonlinear way to obtain a high-dimensional projection of the low-dimensional input space. Preceding work has demonstrated that such high dimensional transient dynamics endows the neural circuit with analog fading memory (see Appendix) that can provide the circuit with enough computational power for performing open-loop sensory processing tasks (Buonomano & Merzenich, 1995; Maass & Markram, 2004; Maass, Natschlager, & Markram, 2002).

Analog fading memory by itself is not powerful enough to render the circuits the power to hold information in working memory. The obvious reason being that such memory has an upper limit in the order of tens of milliseconds, depending on the time constants of synapses and neurons in the neural circuit (Maass et al., 2002), whereas typically working memory holds information in the order of seconds. Recent results show that precisely tuned synaptic feedback can be a possible mechanism for maintaining persistent memory (Seung, 1996; Seung, Lee, Reis, & Tank, 2000a), and that feedback from trained readout neurons can induce multiple coexisting “partial attractors” in the circuit dynamics (Maass, Joshi, & Sontag, 2006, 2007). These results are further extended here to demonstrate that even in the presence of feedback noise, such “partial attractor” states can be maintained by generic neural circuits on the time scales of several seconds, that is obviously a requirement for tasks involving working memory. The results presented in this article indicate that simple linear readouts from generic neural microcircuit models, that send their output as a feedback signal to the circuit, can be a plausible model of how the interval discrimination task is executed and a subsequent action is chosen.

A preliminary version of some results from this article (a model for just working memory and decision making without subsequent action selection) was presented at a conference (Joshi, 2006).

### 2. Generic neural microcircuit models

In contrast to artificial neural networks, neural microcircuits in biological organisms are composed of diverse components such as different types of spiking neurons and dynamic synapses, each endowed with an inherently complex dynamics of its own. This poses a challenge to construct neural circuits out of biologically realistic computational units that solve specific computational problems, e.g. decision making or motor control. In fact, decision making and motor control are particularly challenging, since these tasks occur on the time scales of seconds, where as the inherent dynamics of spiking neurons takes place on a much faster timescale. This article demonstrates that this problem can be solved, if feedback from trained readouts is available to generic neural microcircuits whose internal dynamics has not been adjusted or specialized for any particular computational task, by taking at any time \( t \), a weighted sum \( w \times y(t) \) of the vector \( y \) that describes the current firing activity of all neurons in the neural circuit. The weight vector \( w \), which remains fixed after training, is the only part that needs to be optimized or specialized for the generation of any particular computational task (decision making or motor control). Each component of \( y(t) \) models the impact that a particular neuron \( v \) may have on the membrane potential of a generic linear readout neuron, which is trained for some specific computational task. The value of \( y(t) \) is obtained by applying a low-pass filter to the spike trains emitted by neurons in the generic neural microcircuit model. Note that this is not biologically unrealistic as it has been shown previously that such weighted sums contain behaviour-relevant information (Wessberg et al., 2000).

For the simulations described in this article, generic cortical microcircuit models consisting of integrate-and-fire neurons were used, with a high level of noise that reflects experimental data. Biologically realistic models of dynamic synapses were used whose individual mixture of pair-pulsed depression and facilitation (depending on the type of pre- and postsynaptic neuron) was based on experimental data (Gupta, Wang, & Markram, 2000; Markram, Wang, & Tsodyks, 1998). These circuits were not created for any specific computational task. Sparse synaptic connectivity between neurons was generated (with a biologically realistic bias towards short-range connections) by a probabilistic rule, and synaptic parameters were chosen randomly from distributions that depended on the type of pre- and postsynaptic neurons (in accordance with empirical data from Markram et al. (1998), Gupta et al. (2000)). The neurons in the generic cortical microcircuit models were placed on the integer-points of a 3-D grid, and 20% of these...
neurons were randomly chosen to be inhibitory. The probability of a synaptic connection from neuron $a$ to neuron $b$ (as well as that of a synaptic connection from neuron $b$ to neuron $a$) was defined as $C \exp(-D^2/(a,b)/\lambda^2)$, where $D(a,b)$ is the Euclidean distance between neurons $a$ and $b$, and $\lambda$ is a parameter which controls both the average number of connections and the average distance between neurons that are synthaptically connected. Depending on whether the pre- and postsynaptic neurons were excitatory ($E$) or inhibitory ($I$), the value of $C$ was set according to Gupta et al. (2000) to 0.3 ($E,E$), 0.2 ($E,I$), 0.4 ($I,E$), 0.1 ($I,I$).

2.1. Neuron model

A standard leaky integrate-and-fire neuron model was used, where the membrane potential $V_m$ of a neuron is given by

$$\tau_m \frac{dV_m}{dt} = -(V_m - V_{\text{resting}}) + R_m \cdot (I_{\text{syn}} + I_{\text{inject}} + I_{\text{noise}}) \quad (1)$$

where $\tau_m$ is the membrane time constant (30 ms), which subsumes the time constants of synaptic receptors as well as the time constant of the neuron membrane. Other parameters: absolute refractory period 3 ms (excitatory neurons), 2 ms (inhibitory neurons), threshold 15 mV (for a resting membrane potential $V_{\text{resting}}$, assumed to be 0), reset voltage drawn uniformly from the interval [13.8, 14.5 mV] for each neuron, input resistance $R_m$, 1 MΩ, constant non-specific background current $I_{\text{inject}}$ uniformly drawn from the interval [13.5, 14.5 nA] for each neuron, an additional time-varying noise input current $I_{\text{noise}}$ was drawn every time step from a Gaussian distribution with mean 0 and SD chosen for each neuron randomly from the uniform distribution over the interval [4.0, 5.0 nA]. For each simulation, the initial condition of each I&F neuron, i.e. its membrane voltage at time $t = 0$, was drawn randomly (uniform distribution) from the interval [13.5, 14.9 mV]. Finally, $I_{\text{syn}}(t)$ is the sum of input currents supplied by the explicitly modeled synapses.

2.2. Synaptic parameters

The short-term dynamics of synapses was modeled according to the model proposed in Markram et al. (1998), with the synaptic parameters $U$ (use), $D$ (time constant for depression), $F$ (time constant for facilitation) randomly chosen from Gaussian distributions that model empirically found data for such connections. This model predicts the amplitude $A_k$ of the EPSC for the $k$th spike in a spike train with interspike intervals $\Delta_1, \Delta_2, \ldots, \Delta_{k-1}$ through the equations

$A_k = w \cdot u_k \cdot R_k$

$u_k = U + u_{k-1}(1 - U) \exp(-\Delta_{k-1}/F)$

$R_k = 1 + (R_{k-1} - u_{k-1}R_{k-1} - 1) \exp(-\Delta_{k-1}/D)$

with hidden dynamic variables $u \in [0,1]$ and $R \in [0,1]$ whose initial values for the first spike are $u_1 = U$ and $R_1 = 1$ (see Maass and Markram (2002) for a justification of this version of the equations, which corrects a small error in Markram et al. (1998)). Depending on whether $a$ and $b$ were excitatory ($E$) or inhibitory ($I$), the mean values of the three parameters $U, D, F$ (with $D, F$ expressed in seconds, s) were chosen according to Gupta et al. (2000) to be 0.5, 1.1, 0.05 ($E,E$), 0.05, 0.125, 1.2 ($E,I$), 0.25, 0.7, 0.02 ($I,E$), 0.32, 0.144, 0.06 ($I,I$). The SD of each of these parameters was chosen to be 50% of its mean. The mean of the scaling parameter $w$ (in nA) was chosen to be 70 ($E,E$), 150 ($E,I$), $-47$ ($I,E$), $-47$ ($I,I$).

In the case of input synapses the parameter $w$ had a value of 70 nA if projecting onto an excitatory neuron and $-47$ nA if projecting onto an inhibitory neuron. The SD of the parameter $w$ was chosen to be 70% of its mean and was drawn from a gamma distribution. The postsynaptic current was modeled by an exponential decay $\exp(-t/\tau_v)$ with $\tau_v = 3$ ms ($\tau_v = 6$ ms) for excitatory (inhibitory) synapses. The transmission delays between neurons were chosen uniformly to be 1.5 ms ($E,E$), and 0.8 ms for the other connections.

Most other models of working memory need slow time constants in the order of 100 ms, to enhance the time constant of the entire network (Miller et al., 2003). These slow time constants are based on the simplified assumption that recurrent activity is transmitted purely through the slower NMDA receptors (NMDARs). It is worth noting that no such assumption was made in this article and the long time constants of NMDARs were not used at all.

3. Results

The experiments for tasks described in this article consisted of two distinct phases. In the first phase linear readouts that received inputs from the circuit, were trained in an open-loop fashion to perform diverse computational tasks (e.g. to make decisions, to generate a motor command or to predict the joint angles during the arm movement). During open-loop training, the feedback from readouts performing diverse computational tasks were simulated by a noisy version of their target output (“teacher forcing”). More precisely, at each time-step $t$, a different noise value of $0.0001 \times \rho \times f(t)$ was added, where $\rho$ is a random number drawn from a gaussian distribution with mean 0 and SD 1, and $f(t)$ is the current value of the feedback signal (signal-dependent noise). Note that teacher forcing with noisy versions of target feedback values trains these readouts to correct errors resulting from imprecision in their preceding feedback (rather than amplifying errors). Each readout neuron was trained by linear regression to output at any time $t$, a particular target value $f(t)$. Linear regression was applied to a set of data points of the form $(y(t), f(t))$, for many time points $t$, where $y(t)$ is the output of low-pass filters applied to the spike trains of presynaptic neurons, and $f(t)$ is the target output. Note that this form of training keeps the internal dynamics of the neural circuit intact, and only modifies the weight vector $w$, that corresponds to the projection of the circuit dynamics onto a linear readout for the generation of any particular computational task. Performance of a readout was measured in terms of the correlation value of its target and observed signals during any trial for all experiments reported in this article.

The training phase was followed by a subsequent closed-loop validation phase. During validation, teacher signals to
the generic neural microcircuits were replaced by actual feedback from trained readouts. Both the teacher feedback signal and the actual feedback signal was analog streams fed into the neural microcircuit using a standard form of population coding (Pouget & Latham, 2003): each signal was mapped onto an array of 50 symbolic neurons with bell-shaped tuning curves. Thus the value of the signal is encoded at any time by the output values of the associated 50 symbolic neurons (of which at least 43 neurons output at any time the value 0). It is to be noted that the teacher feedback signal was only used during the open-loop training and was absent in the closed-loop validation phase, where it was replaced by the actual feedback from the trained readout.

In principle, if significantly larger circuits are used, thereby providing the neural microcircuits with more robust kernels, then one can also send this feedback directly to the neurons in the generic neural microcircuit. Nevertheless, population coding of feedback variables is suitable in context of this article as previous studies demonstrate that neurons in PFC process information via a population code (Fuster, 1973; Fuster, Bodner, & Kroger, 2000; Miller, Erickson, & Desimone, 1996). Also population coding in M1 neurons has been shown to predict movement direction in 2-D (Georgopoulos, Caminiti, Kalaska, & Massey, 1983), as well as 3-D (Georgopoulos, Kettner, & Schwartz, 1988; Georgopoulos, Schwarz, & Kettner, 1986; Kettner, Schwarz, & Georgopoulos, 1988; Schwartz, Kettner, & Georgopoulos, 1988) space, movement trajectories (Schwartz, 1993, 1994; Schwartz & Moran, 1999), and several other variables crucial for motor control (Ashe et al., 1993; Georgopoulos, Lurito, Petrides, Schwartz, & Massey, 1989; Kruse, Dannenberg, Kleiser, & Hoffmann, 2002; Smyrnis, Kettner, & Georgopoulos, 1992).

3.1. The two-interval discrimination task

In the actual experimental protocol for two-interval discrimination task, a pretrained subject is presented with two frequencies \( f_1 \) and \( f_2 \), separated by a certain delay interval. Initially the frequency \( f_1 \) is loaded into working memory, and its value is maintained during the delay phase, and on presenting the \( f_2 \) frequency, the subject is required to decide whether \( f_1 > f_2 \). A key feature of these vibrotactile sensory inputs arriving from secondary somatosensory cortex (S2) to PFC is that they arrive through identical neural sensory inputs composed of two frequencies \( f_1 \) and \( f_2 \) which are fed back into the neural microcircuit using a simple form of population coding as discussed earlier (see Section 3). The generic neural microcircuit used in this experiment consisted of 300 integrate-and-fire neurons arranged on the grid-points of a \( 20 \times 5 \times 3 \) cube in 3D. Fig. 1A shows the spatial arrangement of neurons in the neural microcircuit model considered for this experiment. The neurons in the three columns on the left-hand side encode the values of the external input and feedback variables. Each of these three columns of neurons was composed of 50 symbolic input neurons, with the leftmost and rightmost column of neurons being used to encode the values of \( r_{o+}(t) \) and \( r_{o-}(t) \) respectively using population coding. Note that in the open-loop training phase, these neurons were encoding the teacher feedback signal, while in the closed-loop validation phase these same neurons were encoding the output of the trained readouts.

In addition to the feedback, the circuit receives external sensory input composed of two frequencies \( f_1 \) and \( f_2 \). These vibrotactile frequencies \( f_1 \) and \( f_2 \) are presented during the L and D phases (see Fig. 1 and also Fig. 2A) through the same input lines, and are coded using simple monotonic coding. In the training phase the circuit was trained for 100 trials with the training data composed of 10 noisy versions of each of the 10 input pairs \( \{f_1, f_2\} \) (see Fig. 2B). The target functions of “+” and “−” readouts are as described above (see Appendix for details). During each trial, \( f_1 \) and \( f_2 \) are presented for 0.5 s each, during the L and D phases. In the closed-loop validation phase, the performance was validated for 100 trials, with the external sensory inputs for these validation trials being 10 noisy versions of each of the 10 input frequency pairs (not seen during training).

1 The following convention was applied. Each signal is first scaled into the range \([0, 1]\). This range is linearly mapped onto an array of 50 symbolic neurons. At each time step, one of these 50 neurons, whose number \( n(t) \in [1, \ldots, 50] \) reflects the current value \( f(t) \in [0, 1] \) which is the normalized value of the analog signal \( f(t) \) (e.g. \( n(t) = 1 \) if \( f(t) = 0 \), \( n(t) = 50 \) if \( f(t) = 1 \)). The neuron \( n(t) \) then outputs at time \( t \) the value \( f(t) \). In addition the three closest neighbours on both sides of neuron \( n(t) \) in this linear array get activated at time \( t \) by a scaled down amount according to a Gaussian function (the neuron number \( n \) outputs at time step \( t \) the value \( f(t) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(n-n(t))^2}{2\sigma^2}} \), where \( \sigma = 0.8 \)).

2 The external input composed of \( f_1 \) and \( f_2 \) was encoded by a simple monotonic code by a set of 50 input neurons (the middle column of the three columns on the left-hand side of Fig. 1A) with variable transmission delays \( \Delta \), uniformly drawn from the interval \([0, 10]\) ms.

3 Each trial lasted for 3.5 s, with a simulation time-step of 10 ms. The duration of L, M, and D phases were 0.5, 2.0, and 0.97 s respectively. The cue stimulus \( f_1 \) was presented for 0.5 s during the L phase (starting at \( t = 0.03 \) s), which was followed by the M phase (starting at \( t = 0.53 \) s). The probe stimulus was presented for 0.5 s from the start of D phase (starting at \( t = 2.53 \) s).
Fig. 1. (A) Spatial arrangement of neurons in the neural microcircuit model considered for the two-interval discrimination task. Each of the three columns on the left-hand side is composed of 50 symbolic input neurons encoding the values of $r_{o+}(t)$, the external input composed of the two frequencies $\langle f_1, f_2 \rangle$ and $r_{o-}(t)$. The values of $r_{o+}(t)$ and $r_{o-}(t)$ are encoded using a simple form of population coding, and the external input frequencies $\langle f_1, f_2 \rangle$, are encoded using a simple monotonic code. Connections from the input neurons to the neurons in the model PFC circuit and also the recurrent connections in the model PFC circuit are chosen randomly according to a probability distribution discussed in the text (a few typical examples are shown). The model PFC circuit is made of 300 integrate-and-fire neurons arranged on the integer points of a $20 \times 5 \times 3$ cube and is shown on the right-hand side. (B) Closed-loop setup for the two-interval discrimination task. The weight vectors of the “+” and “−” readouts from this circuit, that produce the decision signals, are the only parameters that are adjusted during training. The task is to answer the question if “$f_1 > f_2$”? The “+” (“−”) neurons show an increase in their activity when the answer to the above question is “yes” (“no”). The notation $z^{-1}$ denotes a unit time-step delay in the feedback signal. Note that during training a teacher feedback signal is provided (shown as solid lines) which is replaced by the actual feedback from the trained readouts during the validation phase (dashed lines).

Fig. 2E shows a 200 ms blowup of the circuit response of 100 randomly chosen neurons (activity of inhibitory neurons shown in red) during one of the closed-loop validation runs ($f_1 = 18$ Hz, $f_2 = 26$ Hz). The panels C and D of Fig. 2 show the target (black) and observed (red) values of the “+” and “−” readouts during this run. Panel F and G show the closed-loop validation response of the “+” and “−” readouts for the 10 pairs of input frequencies, $\langle f_1, f_2 \rangle$ (note the similarity to Fig. 1, panels C and D in Machens et al. (2005), which show the actual data from “+” and “−” neurons in PFC during the two-interval discrimination task).

Robustness. To test the robustness of the neural model, experiments were done where, after the readouts were trained, a subset $\kappa_n$ ($\kappa_n$ progressively increased from 0.5% to 5% in 10 trials such that $\kappa_n \subset \kappa_{n+1}$) of synapses converging onto the “+” readouts were randomly chosen and pruned (synaptic weight set to 0). The resulting set-up was tested for 100 trials using 10 noisy versions of each of the 10 frequency pairs $\langle f_1, f_2 \rangle$. Panels H and I of Fig. 2 show the result of these robustness tests. Panel H shows the mean and standard error of correlation values between the target and observed values of “+” neuron for progressively higher levels of pruned synapses. Panel I shows the resulting matrix of same correlation values, where each square shows the mean correlation value for 10 validation runs using noisy versions of a particular frequency pair $\langle f_1, f_2 \rangle$, for a particular pruning percentage. The control correlation values (no pruning) are shown in the row on the

4 With the correlation values between target and observed signals being 0.98 for the “+” neuron and 0.95 for the “−” neuron.

5 Using the same 100 pairs of input frequencies $\langle f_1, f_2 \rangle$ for all levels of pruning.

6 For 100 validation runs for each level of pruning.
Results indicate that the model shows graceful degradation in presence of suboptimal inputs, and the performance of this model is comparable to other more traditional models of attractor-based neural computation (Seung, Lee, Reis, & Tank, 2000b).

Fig. 2. (A) The external frequencies $f_1$ (18 Hz), and $f_2$ (26 Hz) presented during the L and D phase for one of the closed-loop validation trials. (B) Stimulus frequencies used in this study. The target (black) and observed (red) values for the (C) “+” readout, and the (D) “−” readout. (E) A blowup of 200 ms of resulting firing activity of 100 randomly chosen neurons in the circuit. Inhibitory neurons are shown in red. Responses of the (F) “+” and (G) “−” readouts for each of the frequency pairs. The colorbar at upper left indicates the value of $f_1$ used in each trial. (H) Mean and standard error of correlation values (between the target and observed values) of “+” readout for trials with progressively higher pruning percentage. (I) The resulting matrix of correlation values where each square shows the mean correlation value over 10 runs for a particular frequency pair, for a particular pruning percentage. The control correlation values (no pruning) are shown in the row on the top.

3.2. Delayed-match-to-sample followed by a goal-directed arm movement

The decisions taken in the PFC region are acted upon by the motor commands issued by the M1 neurons of the sensorimotor system. To demonstrate that the neurocomputational paradigm
Fig. 3. Block diagram depicting the setup used in the delayed match-to-sample experiment. The circuit modelling PFC function consisted of 500 integrate-and-fire neurons arranged on the grid points of a $20 \times 5 \times 5$ cube and the circuit modelling M1 was made of 1000 integrate-and-fire neurons arranged on the grid points of a $20 \times 5 \times 10$ cube. The model PFC circuit received three color stimuli ($C_{\text{cue}}$, $C_{\text{left}}$, $C_{\text{right}}$) as external inputs, and 2 feedback signals ($r_{\text{left}}(t)$, $r_{\text{right}}(t)$) from the “left” and “right” readouts. The “left” (“right”) neurons show an increase in their activity when $C_{\text{cue}} = C_{\text{left}}$ ($C_{\text{right}}$). In addition to the response of the “left” and “right” readouts, the model M1 circuit receives four additional external inputs ($X_{\text{left}}$, $Y_{\text{left}}$, $X_{\text{right}}$, $Y_{\text{right}}$; the X and Y coordinates of the centre of the left and right probe regions), and four feedback signals ($\hat{\theta}_1(t)$, $\hat{\theta}_2(t)$, $\tau_1(t)$, $\tau_2(t)$). The task is to move the arm to the centre of the probe region that matches in color to the cue stimuli.

Presented in this article is capable of integrating the A phase, and is task independent, delayed-match-to-sample was chosen as an alternative interval discrimination paradigm. In this task, a pretrained subject is presented with a cue visual stimulus during the L phase (for example a small colored square on a predetermined region of the screen), which is followed by a delay period (M phase), and subsequently in the D phase two simultaneous probe stimuli are presented at two different locations on the workspace of the arm (see Fig. 5 panels A and B). The neurons in the model PFC circuit made projections to two linear readouts (called “left” and “right” readouts from now on) which had similar functionality as the “+” and “−” readouts in the two-interval discrimination task. The “left” readout showed high activity during the D phase, if the answer to the question “$C_{\text{cue}} = C_{\text{left}}$?” was positive. The “right” readout behaved exactly opposite to this and showed increased activity in the D phase, if the probe stimulus shown on the right matched the cue stimulus. Please see Appendix for details about target functions for these readouts.

The outputs of the “left” and “right” readouts were sent as inputs to the model M1 circuit. In addition, the circuit also received four external inputs ($X_{\text{left}}$, $Y_{\text{left}}$, $X_{\text{right}}$, $Y_{\text{right}}$; the X and Y coordinates of the centre of the left and right probe regions). The model PFC circuit received three external inputs ($C_{\text{cue}}$, the cue color; $C_{\text{left}}$, the left probe color; $C_{\text{right}}$, the right probe color), and two feedback signals (from the “left” and “right” readouts). The cue stimulus was presented during the L phase and the probe stimuli were presented during the D phase, at three different locations on the workspace of the arm (see Fig. 5 panels A and B). The neurons in the model PFC circuit made projections to two linear readouts (called “left” and “right” readouts from now on) which had similar functionality as the “+” and “−” readouts in the two-interval discrimination task. The “left” readout showed high activity during the D phase, if the answer to the question “$C_{\text{cue}} = C_{\text{left}}$?” was positive. The “right” readout behaved exactly opposite to this and showed increased activity in the D phase, if the probe stimulus shown on the right matched the cue stimulus. Please see Appendix for details about target functions for these readouts.
It is to be noted that the neurons in the model M1 circuit made projections to four linear readouts. Two of these readouts \( \langle \hat{\theta}_1 \rangle, \langle \hat{\theta}_2 \rangle \rangle \) were trained to predict the current values of the shoulder and elbow joint angles. The other two readouts \( \langle \tau_1 \rangle, \langle \tau_2 \rangle \rangle \) were trained to generate joint torques to drive the arm from its resting position to the centre of the probe region that matched the stimuli (see Fig. 5 panels A and B).

During the open-loop training, the circuit was trained for 100 trials\(^8\) using 10 noisy versions of each of the five input color triplets \( (C_{\text{cue}}, C_{\text{left}}, C_{\text{right}}) \) (see Fig. 4A, right side). The target trajectory followed by the tip of the arm was generated using the minimum-jerk model (for details see Appendix) described in Flash and Hogan (1965). For the closed-loop validation phase, 100 trials were performed using 20 noisy versions of each of the five input color triplets.

Fig. 4 shows the result of a closed loop validation run. The target (black) and observed (red) response of the “left” and “right” readouts are shown in panel B and D respectively. The panels C and E of Fig. 4 show the response of the “left” and “right” readouts for the five input stimuli (each line drawn in the color of corresponding cue stimulus). Panels F and G show the target (black) and observed (red) motor commands (joint torques) generated in the A phase. Panels H and I show the target (black), predicted (red) and actual (blue dashed) values for the joint angles. The movement of the tip of the robot arm from the resting phase to the decided end-point occurred with a biologically realistic bell-shaped velocity profile shown in panel J.

**Robustness.** The columns in panel K of Fig. 4 demonstrate the performance of this setup over 100 closed-loop validation runs. More precisely, these results show the histogram of correlation values between the target and observed signals for the “left” (mean = 0.996, SD = 0.002) and “right” (mean = 0.994, SD = 0.002) readouts, the readouts that computed the joint torques \( \langle \tau_1 \rangle, \langle \tau_2 \rangle \rangle \) mean = 0.92, SD = 0.02; \( \langle \tau_1 \rangle, \langle \tau_2 \rangle \rangle \) mean = 0.96, SD = 0.002, and the readouts that predicted the joint angles \( \langle \hat{\theta}_1 \rangle, \langle \hat{\theta}_2 \rangle \rangle \) mean = 0.63, SD = 0.44; \( \langle \hat{\theta}_1 \rangle, \langle \hat{\theta}_2 \rangle \rangle \) mean = 0.99, SD = 0.005. The last two plots from the right in panel K also show the histogram of correlation for the actual joint angles with the target (shown in blue, for \( \theta_1 \) mean = 0.97, SD = 0.03, for \( \theta_2 \) mean = 0.99, SD = 0.004).

The panels A B of Fig. 5 shows typical movement performed by the arm when the cue stimulus matches the stimulus shown on the left (right) probe location. In both panels, O denotes the position of the end point of arm before movement initiation. L, R and C denote the position of the left, right and cue stimulus. Panel C of Fig. 5 shows the target trajectories of the tip of the arm during a movement to the left or right probe location (solid), and the actual trajectories (dashed) for one validation run to the left and one to the right. The open (closed) circles represent the location of the end-point for other trials, at the end of movement, when the target end point was the centre of the left (right) probe region. Panel D shows the histogram of error values for the 100 trials, where the error was defined as the Euclidean distance between the target and observed end-point of the arm, for each trial (mean = 0.11 m, SD = 0.08 m).

### 3.3. Memory and decision signals from untrained neurons in neural microcircuits

Recordings from randomly chosen task-related neurons in PFC show a strong correlation to memory and decision signals during a working memory task (Brody, Hernandez, Zainos, & Romo, 2003; Machens et al., 2005). The results shown in Fig. 6 show that randomly chosen neurons from generic neural microcircuits used in both the tasks discussed in this article, also contain strong correlation to the working memory and decision signals. Panel A of Fig. 6 shows the target (solid) and observed (dashed) performance of the “+” readout during one of the closed-loop validation runs\(^9\) of the two-interval discrimination task (for the case when \( f_1 \) was less than \( f_2 \)). Panel B shows a mean peristimulus time histogram (PSTH) derived from responses of 50 randomly chosen neurons from the layer of 100 neurons that received the feedback signal \( r_{o+}(t) \), in the neural microcircuit used in this experiment. It was observed that the PSTH signal has strong correlation\(^10\) with the observed value of the “+” readout indicating that the neurons in the generic circuit contained information about the decision signal. Panel C and D show similar results for a closed-loop validation run during the delayed match-to-sample task (for the case when the color of the left probe matched the cue stimulus) for the “left” readout.\(^11\) It is to be noted that unlike the response of trained readouts, the PSTH’s shown in panel B and D show the recordings from randomly chosen untrained neurons from the layers of neural microcircuit that were receiving the corresponding signals from the trained readouts.

### 4. Discussion

This article describes a new neurocomputational paradigm that uses synaptic learning mechanisms to present a unified model for decision making followed by action selection. The model is unified in the sense that a single learning algorithm (simple linear regression) is used to train readouts that make decisions based on the activity of a model PFC circuit, and the

---

\(^{8}\) Each trial lasted for 2.5 s, with a simulation time step of 10 ms. The duration of L, M, D and A phases were 0.32, 1.0, 0.6, and 0.5 s respectively. The cue stimulus \( C_{\text{cue}} \) was presented for 0.32 s during the L phase (starting at \( t = 0.08 \) s), which was followed by the M phase (starting at \( t = 0.44 \) s). \( C_{\text{left}} \) and \( C_{\text{right}} \) are presented simultaneously for 0.6 s from the start of D phase (starting at \( t = 1.4 \) s). The A phase (starting at \( t = 2.0 \) s) consisted of moving the arm from the initial resting position to the decided position in 0.5 s (see Fig. 4, panel A, left side).

\(^{9}\) With the correlation between target and observed signal being 0.98.

\(^{10}\) With the correlation between the PSTH signal and the observed value of the “+” readout being 0.90.

\(^{11}\) With the correlation value between the target and observed signal being 0.99, and the correlation between the observed value and the PSTH being 0.91.
Fig. 4. (A) The left column shows external stimuli to the model PFC circuit during one of the closed loop validation runs. The cue stimulus is presented during the L phase and the left-and-right probe stimuli are presented during the D phase. The right column shows the colors of the cue, left and right stimuli used in trials. The target (black) and observed (red) values for the (B) “left” readout, and the (D) “right” readout. Responses of the (C) “left” and (E) “right” readouts for each of the color triplets. Target (black) and observed (red) motor commands (joint torques) generated in the A phase for the (F) shoulder and (G) elbow joints. The target (black), predicted (red) and actual (blue dashed) values for the joint angles for the (H) shoulder and (I) elbow joints. (J) The movement of the tip of the robot arm from the resting phase to the decided end-point occurred with a biologically realistic bell-shaped velocity profile. (K) Histogram of correlation values between target and observed signals for readouts that made decisions (“left” and “right”), issued motor commands \( \langle \tau_1(\cdot), \tau_2(\cdot) \rangle \), and predicted the joint angles \( \langle \hat{\theta}_1(\cdot), \hat{\theta}_2(\cdot) \rangle \). The last two plots from the right in panel (K) also show the histogram of correlation for between the actual joint angles and their target values (shown in blue).

readouts that generate motor commands or predict the joint angles based on the activity of the model M1 circuit. Biologically realistic neural microcircuit models composed of spiking neurons and dynamic synapses were used to generate models for two different interval discrimination paradigms. Initially, for comparison with other models of working memory, a model was presented in Section 3.1 for the two-interval discrimination task that incorporated the L, M and D phases involved in working memory and decision making. Additionally to show that this paradigm can be extended to incorporate the A phase,
is robust how external inputs are encoded and task independent; another model was presented in Section 3.2 for the delayed match-to-sample task with a goal directed movement of a two-jointed robot arm during the A phase. Note that the A phase for the two-interval discrimination task could easily have been implemented in an identical manner to the delayed match-to-sample task, but was avoided to prevent redundancy.

A prominent feature of this neural model is that synaptic learning only occurs at synapses that project the activity in the neural microcircuit models onto the linear readout neurons, leaving the circuit dynamics intact. A key advantage of this learning mechanism is that since only the readouts, and not the neural circuit itself, have to adapt to specific computational tasks, the same circuit can support several other open-loop computations in parallel. In principle, one can of course also view various parameters within the circuit as being subject to learning or adaptation, for example, in order to optimize the dynamics of the circuit for a particular range of computational tasks. However, this has turned out to be not necessary for the tasks described in this article, although it remains an interesting open research problem how unsupervised learning could optimize a circuit for some particular computational task.

It is to be noted, however, that although spiking neural circuits were used to model the interval-discrimination tasks for added biological realism, it is not a requirement, and any recurrent circuit would give similar results, as long as it has the kernel property. Readouts make a binary decision by reaching one of the states corresponding to the decision made by them. The actual point of time when the readout makes a decision can be thought of as a threshold crossing event, i.e. the first time when the readout crosses a threshold after the presentation of the probe stimulus in the D phase.

It should be noted that, in principle, the task described in Section 3.2 can also be performed by a single large neural microcircuit. Instead, two separate neural microcircuits were used to model the functionality of PFC and M1 individually for added biological realism, with the synaptic input from PFC to M1 being optimized by linear regression. Note that the sparse, long-range synaptic projections from M1 to PFC were not modeled, as they were not required for this task. More precisely, these recurrent projections were not modeled since the decision making process is not influenced by the current action performed, so this feedback had no computational relevance.

The feedback from the trained readouts played an apparently important role in the neural model described above. Although the generic neural microcircuits used to model the PFC and M1 circuits are endowed with fading memory due to the kernel property of the circuits, this is not sufficient for holding information in working memory for longer timespans...
Fig. 6. Memory and decision signals from untrained neurons in neural microcircuits. (A) The target (solid) and observed (dashed) performance of the “+” readout during one of the closed-loop validation runs for the two-interval discrimination task. (B) The PSTH obtained by recording the activity of 50 randomly chosen neurons from the layer of 100 neurons that received this feedback signal in the neural microcircuit used in this experiment. Strong correlation was observed between the PSTH signal and the observed value of the “+” readout indicating that the neurons in the generic circuit contained information about the decision signal. Panel (C) and (D) show similar results for the “left” readout, during a closed-loop validation run for the delayed match-to-sample task.

ranging in the order of seconds. Apparently the feedback from trained readouts provides the circuit with additional needed information that falls outside the window of fading memory, hence enhancing the information present in the circuit dynamics.

Obviously closed-loop applications of generic neural microcircuit models like the ones discussed in this article present a harder computational challenge than open-loop sensory processing tasks, since small imprecisions in their output are likely to be amplified by the plant (e.g. the arm model) to yield even larger deviations in the feedback, which is likely to further enhance the imprecision of subsequent outputs. This problem can be solved by teaching the readouts from the neural microcircuit during training to ignore smaller recent deviations reported by feedback, thereby making the target trajectory of output torques an attractor in the resulting closed-loop dynamic system. It is not claimed, however, that such teacher signals are biologically realistic, specially in the context of decision making. However, they can be interpreted, for example, in the context of movement generation, as the movements shown by an instructor during learning of a new motor skill (“imitation learning”). In a related vein, it should be noted that the exact shape of the target signals for the decision making readouts (as described in the Appendix), was determined empirically with a motivation to keep it qualitatively close to the actual data from Romo’s lab.

One of the potential drawbacks of earlier models of working memory (Machens et al., 2005; Miller & Wang, 2006) is their inability to capture the temporal dynamics of real PFC neurons during the delay period (Latham & Dayan, 2005). While the real PFC neurons show a monotonic increase or decrease in their firing rate during the D phase, the neurons in these models have a stationary firing rate in this phase. The neural algorithm presented in this article is able to overcome this challenge, as the readouts can successfully “learn” to decode the value of the original cue stimuli from its temporally decayed value.

According to the “internal model” hypothesis (Bizzi & Mussa-Ivaldi, 1998; Craik, 1943), there exists an internal model for each of the tasks that we have learned throughout our lives. One outcome of such a hypothesis would be that a given neuron may participate with a different synaptic weight in a number of neural assemblies, each supporting a different internal model. Interestingly, this is reflected in our set-up too, as neurons in the generic neural circuit make synaptic projections to the set of readouts with different synaptic weights assigned for each task.

This study also demonstrates the ability of generic neural microcircuit models to hold “partial attractor” states in their circuit dynamics for significantly longer and biologically relevant time scales ranging in the order of a couple of seconds, even in the presence of noise. The role of feedback in enhancing the inherent fading memory of a neural circuit is also demonstrated. Further it also shows the potential of generic neural circuits to integrate the mechanisms involved in the sense-think-decide-act sequence involved in decision making followed by action selection, that happen at significantly longer time scales. The idea of a uniform modus operandi across cortical regions is not only attractive, but also intuitively
plausible. Further work is needed to explore the ideas presented here in detail.

Acknowledgments

Helpful comments from Wolfgang Maass, Herbert Jaeger, Carlos Brody and anonymous reviewers on the draft version of this manuscript are gratefully acknowledged. Written under partial support by the Austrian Science Fund FWF, project #P17229-N04 and project #FP6-015879 (FACETS) of the European Union.

Appendix

A.1. Fading memory

A map (or filter) \( F \) from input-to-output streams is defined to have fading memory if its current output at time \( t \) depends (up to some precision \( \epsilon \)) only on values of the input \( u \) during some finite time interval \([t – T, t]\). Formally, a filter \( F \) has fading memory if there exists for every \( \epsilon > 0 \) some \( \delta > 0 \) and \( T > 0 \) so that \(|\{F(u)(t) - (Fv)(t)\} < \epsilon\) for any \( t \in \mathbb{R} \) and any input functions \( u, v \) with \(|u(\tau) - v(\tau)| < \delta\) for all \( \tau \in [t – T, t] \).

Note that fading memory is just a continuity property (that is characteristic for filters whose output is defined via one or several integrals over the input functions) which implies graceful decay of the output precision when there is noise on the input: the most relevant bits of the output value \( (Fv)(t) \) depend only on the most relevant bits of \( u(s) \) for arguments \( s \) from some finite domain \([t – T, t]\). It is easy to see that any linear filter, and each higher order term of a Volterra series, is time invariant and has fading memory.

A.2. Model for the two-joint robot arm

The model can be described using the well-known Lagrangian equation in classical dynamics. The dynamic equations for this arm model are given by Eq. (2)

\[
\begin{bmatrix}
H_{11} & H_{12} \\
H_{21} & H_{22}
\end{bmatrix}
\begin{bmatrix}
\ddot{\theta}_1 \\
\ddot{\theta}_2
\end{bmatrix}
+ \begin{bmatrix}
-k\dot{\theta}_1 - h(\dot{\theta}_1 + \dot{\theta}_2) \\
h\dot{\theta}_1
\end{bmatrix}
= \begin{bmatrix}
\tau_1 \\
\tau_2
\end{bmatrix}
\]

with \( \theta = [\theta_1, \theta_2]^T \) being the two joint angles, \( \tau = [\tau_1, \tau_2]^T \) being the joint input torques to the two joints, and

\[
\begin{align*}
H_{11} &= m_1l_1^2 + I_1 + m_2[l_1^2 + l_2^2 + 2l_1l_2 \cos \theta_2] + I_2 \\
H_{12} &= H_{21} = m_2l_1l_2 \cos \theta_2 + m_2l_2^2 + I_2 \\
H_{22} &= m_2l_2^2 + I_2 \\
h &= m_2l_1l_2 \sin \theta_2.
\end{align*}
\]

Eq. (2) can be compactly written as

\[
H(\dot{\theta})\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} = \tau
\]

where \( H \) represents the inertia matrix, and \( C \) represents the matrix of Coriolis and centripetal terms. \( I_1, I_2 \) are the moments of inertia of the two joints. The values of the parameters that were used in simulations were: \( m_1 = 1, m_2 = 1, l_1 = 0.25, l_2 = 0.25, I_1 = 0.03, I_2 = 0.03. \)

A.3. The minimum-jerk model

For a given start point \( (x_{\text{start}}, y_{\text{start}}) \) and target end point \( (x_{\text{dest}}, y_{\text{dest}}) \) of a movement (both given in Cartesian coordinates), an interpolating trajectory of the tip of the arm was generated according to the following equation given in Flash and Hogan (1965):

\[
\begin{align*}
x(t) &= x_{\text{start}} + (x_{\text{dest}} - x_{\text{start}}) \cdot (15\xi^4 - 6\xi^5 - 10\xi^3) \\
y(t) &= y_{\text{start}} + (y_{\text{dest}} - y_{\text{start}}) \cdot (15\xi^4 - 6\xi^5 - 10\xi^3)
\end{align*}
\]

where \( \xi = t/MT \) and \( MT \) is the target movement time (in this case \( MT = 500 \text{ ms} \)). From this target trajectory for the endpoint of the robot arm, the target trajectories of the joint angles \( \theta_1, \theta_2 \) of the robot arm were generated by applying standard equations from geometry (see e.g. Craig (1989)). From these the target trajectories of the torques were generated according to Eq. (2).

A.4. Target values for the “+” and “−” neurons

The target values for these readouts were 0 before the onset of L phase. Following this, the target values for these readouts during L, M and D phases were modeled as simple scaled sigmoidal. More precisely, for a pair of input frequencies, \( (f_1, f_2) \), the target value for the “+” readout, \( f^+ (t) \) is given by

\[
f^+(t) = \begin{cases} 
0 & \text{if } 0 \leq t < 0.03 \text{ s} \\
\frac{0.7 \cdot (f_1 + 20)}{1 + e^{-\frac{5 - 10t - 0.3}{2}}} & 0.03 \leq t \leq 0.53 \text{ s} \\
\frac{60 - 0.7 \cdot (f_1 + 20)}{1 + e^{-\frac{6 + 5.5t - 0.3}{3}}} & 2.53 \leq t \leq 3.5 \text{ s}, f_1 < f_2 \\
0.7 \cdot (f_1 + 20) & 2.53 \leq t \leq 3.5 \text{ s}, f_1 < f_2
\end{cases}
\]

And in similar fashion, the target value for the “−” readout, \( f^- (t) \) is given by

\[
f^-(t) = \begin{cases} 
0 & \text{if } 0 \leq t < 0.03 \text{ s} \\
\frac{64 - f_1}{1 + e^{-\frac{5 - 0.03}{0.9}}} & 0.03 \leq t \leq 0.53 \text{ s} \\
\frac{0.7 \cdot (64 - f_1)}{1 + e^{-\frac{5 - 10t - 0.3}{2}}} & 0.53 \leq t \leq 2.53 \text{ s} \\
0.7 \cdot (64 - f_1) & 2.53 \leq t \leq 3.5 \text{ s}, f_1 > f_2 \\
\frac{60 - 0.7 \cdot (64 - f_1)}{1 + e^{-\frac{6 + 5.5t - 0.3}{3}}} & 2.53 \leq t \leq 3.5 \text{ s}, f_1 < f_2
\end{cases}
\]
A.5. Target functions for the “left” and “right” neurons

The target values for these readouts were 0 before the onset of L phase. Following this the target values for these readouts during L and D phases were modeled as simple scaled sigmoids. The target values during the M phase was modeled as a linear decay. The target value during the A phase was obtained by extrapolating the value reached during the D phase. The sample and probe colors were encoded by their hue values that were scaled in the range [0, 1]. More precisely, for a color triplet \((C_{\text{cue}}, C_{\text{left}}, C_{\text{right}})\), the target value for the “left” readout, \(f_{\text{left}}(t)\) is given by

\[
f_{\text{left}}(t) = \begin{cases} 
0 & 0 < t < 0.08 \\
\frac{1}{1 + e^{-(5+10 \cdot t / 0.08)}} & 0.08 < t \leq 0.40 \\
(1 - C_{\text{cue}}) \cdot (1 - 0.3 \cdot (t - 0.40)) & 0.40 < t < 1.40 \\
0.7 \cdot (C_{\text{cue}} + 0.1) & 1.40 < t < 2.0, C_{\text{cue}} = C_{\text{left}} \\
0.7 \cdot (C_{\text{cue}} + 0.1) + \frac{1 - 0.7 \cdot (C_{\text{cue}} + 0.1)}{1 + e^{-(5+10 \cdot t / 0.08)}} & 1.40 < t < 2.0, \ C_{\text{cue}} = C_{\text{right}} \\
1 + e^{-(5-10 \cdot t / 0.08)} & t > 2.0. 
\end{cases}
\]

And in similar fashion, the target value for the “right” readout, \(f_{\text{right}}(t)\) is given by

\[
f_{\text{right}}(t) = \begin{cases} 
0 & 0 < t < 0.08 \\
\frac{1}{1 + e^{-(5+10 \cdot t / 0.08)}} & 0.08 < t \leq 0.40 \\
(1 - C_{\text{cue}}) \cdot (1 - 0.3 \cdot (t - 0.40)) & 0.40 < t < 1.40 \\
0.7 \cdot (1 - C_{\text{cue}}) & 1.40 < t < 2.0, \ C_{\text{cue}} = C_{\text{left}} \\
0.7 \cdot (1 - C_{\text{cue}}) + \frac{1 - 0.7 \cdot (1 - C_{\text{cue}})}{1 + e^{-(5+10 \cdot t / 0.08)}} & 1.40 < t < 2.0, \ C_{\text{cue}} = C_{\text{right}} \\
1 + e^{-(5-10 \cdot t / 0.08)} & t > 2.0. 
\end{cases}
\]

**References**


