LETTER Communicated by Elham Ghazizadeh

Heterogeneous Forgetting Rates and Greedy Allocation in Slot-Based Memory Networks Promotes Signal Retention

BethAnna Jones
bethannajones@wustl.edu
Department of Electrical and Systems Science, Washington University in St. Louis, St. Louis, MO 63130, U.S.A.

Lawrence Snyder
lsnyder@wustl.edu
Department of Neuroscience, Washington University in St. Louis, St. Louis, MO 63130, U.S.A.

ShiNung Ching
shinung@wustl.edu
Department of Electrical and Systems Science, Washington University in St. Louis, St. Louis, MO 63130, U.S.A.

A key question in the neuroscience of memory encoding pertains to the mechanisms by which afferent stimuli are allocated within memory networks. This issue is especially pronounced in the domain of working memory, where capacity is finite. Presumably the brain must embed some "policy" by which to allocate these mnemonic resources in an online manner in order to maximally represent and store afferent information for as long as possible and without interference from subsequent stimuli. Here, we engage this question through a top-down theoretical modeling framework. We formally optimize a gating mechanism that projects afferent stimuli onto a finite number of memory slots within a recurrent network architecture. In the absence of external input, the activity in each slot attenuates over time (i.e., a process of gradual forgetting). It turns out that the optimal gating policy consists of a direct projection from sensory activity to memory slots, alongside an activity-dependent lateral inhibition. Interestingly, allocating resources myopically (greedily with respect to the current stimulus) leads to efficient utilization of slots over time. In other words, later-arriving stimuli are distributed across slots in such a way that the network state is minimally shifted and so prior signals are minimally "overwritten." Further, networks with heterogeneity in the timescales of their forgetting rates retain stimuli better than those that are more homogeneous. Our results suggest how online, recurrent networks working on temporally localized objectives without high-level supervision can nonetheless implement efficient allocation of memory resources over time.

Neural Computation **36, 1022–1040** (2024)
https://doi.org/10.1162/neco_a_01655

1 Introduction

Working memory is the cognitive system responsible for the temporary storage, processing, and integration of new information. Central to learning, decision making, language, and long-term memory, working memory forms the foundation for higher cognitive function (Baddeley, 1992) and is thus crucial to understanding how the brain functions overall. Studies of auditory and visual working memory reveal persistent neural activity in the prefrontal cortex (PFC) during memory trial delay periods (Funahashi & Kubota, 1994; Joseph et al., 2016; Lara & Wallis, 2015; Shafi et al., 2007), thought to be the neural substrate for encoding or processing memoranda. However, many questions persist regarding the dynamical mechanisms embedded within these circuits and the precise nature by which memory representations are allocated and transformed. Additionally, while methods of measuring memory capacity in neural circuits are still debated, studies show there are limits to capacity (Rouder et al., 2011), and neuropsychiatric illnesses have been shown to be correlated with impairments in working memory capacity as well as the accuracy of memory recall (Lee & Park, 2005; Glahn et al., 2006). Several works have hypothesized that limitations in capacity are due to a "slot"-based resource schema in which each independent slot comprising, say, a subnetwork of neurons of finite size is able to store one or more discrete memoranda (Cowan et al., 2005; Luck & Vogel, 2013). In these models, slots can be shared across memoranda, but at a cost to the ambiguity of the stored information. These works and others suggest that higher cognitive function is dependent on neural circuits allocating finite resources effectively for the accurate storage of information, avoiding redundancy, and budgeting for new demands (Bialek et al., 2007; Ye et al., 2017; Zaccarian, 2009).

Furthermore, our ability to remember and process information from continuously encountered stimuli demands that any memory encoding choices be made in an online manner, requiring presumably prompt and tactful choices of which of hundreds of thousands of different stimuli (or various dimensions of those stimuli) to store, to what degree, and how. Such massive spatiotemporal coordination in and among neural circuits would necessitate the existence of underlying neural mechanisms in order for effective and efficient encoding to be enacted locally within the network and still be globally beneficial. In this vein, theoretical neuroscience works employing formal modeling and analysis have been instrumental in generating hypotheses regarding how memory function is enacted in the brain (Chung & Abbott, 2021). Work by Spaak et al. (2017), Murray et al. (2017), Ghazizadeh and Ching (2021), and Wojtak et al. (2023), among numerous others, proposes that memoranda are encoded and maintained via the formation of asymptotically stable dynamical attractors or oscillatory dynamics (Pina et al., 2018). Further work proposes that both stable and dynamic neural activity work together to balance recency and primacy of

information (Lee et al., 2020; Stokes et al., 2013; Spaak et al., 2017), incorporating new information over time and without disrupting current encodings. An enigmatic issue in the above theories pertains to how heterogeneity in the intrinsic timescales of memory networks (i.e., their rates of decay or "forgetting") contributes to memory function, in light of observations that neuronal activity can ramp up and down during delay periods (Zhang et al., 2015; Murray et al., 2017).

Our goal in this letter is to examine the issue of resource allocation in the encoding and storage phases of working memory. As noted, working memory exhibits finite capacity. We propose here that the fundamental global characteristics of memory circuits—maintaining accurate encodings of information while allowing for new information to be stored and processed in an online manner—engender innate encoding strategies in neural circuits that allow for online and resource-efficient allocation of these resources. We propose a simplified dynamical model of resource allocation in a mathematical instantiation of a slot-based network setting, when memoranda are encoded in discrete subnetworks. Using this model, we perform formal control-theoretic optimization to understand how afferent stimuli would be allocated to these subnetworks under different prioritizations of encoding accuracy versus frugality with respect to the proportion of slots used for any given memoranda. Within this framework, we provide new analysis suggesting (1) that greedy allocation of resources at the moment of encoding may intrinsically promote the retention of signals/information over time and (2) that heterogeneous timescales of forgetting embedded across slots carry benefits in terms of allocation efficiency and performance.

2 Model and Control-Theoretic Formulation

We begin by formulating a mathematical model within which we will explore the theory of how network dynamics can embed memory resource allocation objectives.

2.1 Slot-Based Network. We begin with a canonical linear dynamical system composed of N subnetworks, each conceptualized as a memory resource or slot,

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t), \tag{2.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^N$ is the state of each slot within the network, whose time evolution is determined by $\mathbf{A} \in \mathbb{R}^{N\times N}$. It is important to emphasize here that $\mathbf{x}(t)$ represents neural activity at an abstract, population level and is not meant to reflect biophysical dynamics of single neurons. By this definition, in the absence of any stimulus and under the assumption that $\mathbf{A}$ is Hurwitz, then $\mathbf{x}(t)$ will decay to zero asymptotically, that is, the subnetworks "forget"

asymptotically. Understanding the policies by which $\mathbf{x}(t)$ could be activated in response to incoming memory demand is the central theoretical issue in this letter.

Thus, crucial in our network formulation is the input stimulus $\mathbf{u}(t) \in \mathbb{R}^d$ that impinges on the network via the time-varying gain matrix $\mathbf{B}(t) \in \mathbb{R}^{N\times d}$. Formulation 2.1 represents a quite typical linear time-invariant system formulation that is ubiquitous in control theory. However, a key distinction is that here, we will consider not the design of the external stimulus to direct the network but rather the gain matrix that gates the stimulus onto the network dynamics and hence memory slots. To enable this analysis, we define inputs as an impulse train,

$$\mathbf{u}(t) \triangleq \sum_{k\in\mathbf{Z}^+} \delta(t - t_k)\beta_k, \tag{2.2}$$

where $\delta(\cdot)$ is the Dirac delta function and β_k is the kth stimuli in sequence, occurring at time t_k. Each stimulus $\beta_k \in \mathbb{R}^d$ acts as an abstract feature vector, representing characteristics of the stimulus in d dimensions, where we assume these representations are concise and avoid excessive redundancy. In an effort to maintain the generalizability of our work, we do not specify feature types for each dimension. However, we do make the crucial assumption that $d < N$: the number of nonredundant features of a stimulus will always be smaller than the size of the network.

Thus, formulation 2.2 encapsulates two assumptions regarding stimuli: that they occur as discrete events in a sequence with potentially random timing and that each individual stimulus is described by a finite-dimensional vector. Such a formulation is sometimes referred to in the literature as a marked point process. For the purposes of our study into encoding strategies, we assume each individual stimulus within $\mathbf{u}(t)$ is relevant to the network. That is, we do not include a distinction at this stage regarding the salience or importance of a given stimulus in a sequence (see also section 4). In this manner, our formulation embeds the notion of a sequence of temporally discrete stimuli whose identity must be distributed onto the network. Figure 1 schematizes the basic setup of our formulation.

2.2 Mnemonic Resource Objectives. Encoding a series of stimuli within a finite network involves resource allocation. The slots within the network must be regulated in order to meet the demands of newly arriving stimuli while also retaining previously held ones. How this regulation is enacted will determine the degree to which new stimuli are faithfully stored versus prior stimuli overwritten. In our formulation, the primary regulator of incoming stimuli is the gating matrix $\mathbf{B}(t)$, which must project stimuli onto the network in a manner that balances the retention of new and old signals (Carrillo-Reid et al., 2015; Kessler & Meiran, 2006).

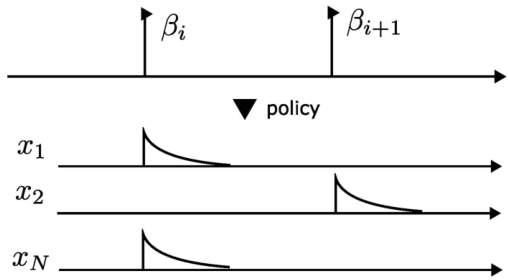

Figure 1: We consider a slot-based model where stimuli are gated onto distinct memory subnetworks according to a policy. The optimization of the policy is the central question considered.

To promote analysis of this problem, we will decompose $\mathbf{B}(t)$ as

$$\mathbf{B}(t) \triangleq \mathbf{b}(t)\mathbf{w}_\mathbf{x}, \tag{2.3}$$

where $\mathbf{w}_\mathbf{x} \in \mathbb{R}^{1\times d}$ compresses the stimulus into a univariate, scalar signal. However, we will later see that this assumption is largely for analytical convenience, and the ultimate gating policy derived will not meaningfully rely on $\mathbf{w}_\mathbf{x}$ and will in fact be sensitive to all stimulus dimensions. It follows from the above definitions that

$$\mathbf{x}(t_k^+) \triangleq \mathbf{x}(t_k) + \mathbf{b}(t_k)\mathbf{w}_\mathbf{x}\beta_k, \tag{2.4}$$

where t_k^+ indicates that the kth stimulus has been gated onto the network. Our formulation proceeds by defining an optimization problem whose solution will specify the $\mathbf{b}(t) \in \mathbb{R}^N$, thus yielding a gating "policy."

2.3 Top-Down Optimization of Resource Gating Mechanisms. We now proceed to define the core top-down optimization problem considered. Specifically, we postulate that any policy for the gating vector $\mathbf{b}(t)$ must balance two factors: (1) the accurate encoding and hence allocation of stimuli as they are received and (2) the minimization of overwriting in currently occupied memory slots, hence promoting memory retention. In a resource-constrained network setting, these objectives may be in opposition (Gorgoraptis et al., 2011).

2.3.1 Encoding Accuracy. In order to model the encoding phase of memory and measure how accurately a stimulus is stored within the network dynamics, we define a linear decoder,

$$\mathbf{z}(t) \triangleq \mathbf{C}\mathbf{x}(t), \tag{2.5}$$

where $\mathbf{C} \in \mathbb{R}^{d \times N}$ is rank d (and $d < N$). The variable $\mathbf{z}(t)$ provides a lower-dimensional read-out of the system slots at any given time, creating a means to translate between the network and stimulus spaces. With the decoder in hand, the first objective (accurate encoding of stimuli as they are received) is readily quantified via the instantaneous error between the input and the decoded network activity:

$$J_{enc}(t) \triangleq \|\mathbf{z}(t) - \mathbf{u}(t)\|_2^2. \tag{2.6}$$

It follows from equation 2.2 that at the time of a specific stimulus, t_k, the input is $\mathbf{u}(t_k) = \beta_k$ and the encoding error for this kth stimulus is given by

$$J_{enc}(t_k^+) \triangleq \|\mathbf{z}(t_k^+) - \beta_k\|_2^2, \tag{2.7}$$

where again t_k^+ indicates the kth stimulus has been gated onto the network. We note that the definition of the decoder here is made primarily to establish a surrogate quantity that embeds a reference for how any given stimulus *could* be allocated to the network slots. Indeed, since the number of slots is assumed to exceed the dimension of the stimuli in $\mathbf{u}(t)$ (i.e., $d < N$), the problem of encoding as per equation 2.5 is underdetermined. Thus, as formulated, the encoding maps to a typical least-squares problem, and there are infinitely many possible configurations of the neural state $\mathbf{x}(t)$ that could be associated with a given β_k, that is, achieving $J(t_k^+) = 0$ in equation 2.7. Thus, there could be a range of possible policies that preserve encoding accuracy.

2.3.2 Encoding Frugality. In addition to encoding accuracy, we also consider the storage phase of memory via the extent to which slots are overwritten by newly arrived stimuli. For this purpose, we use the distance between the neural state prior and subsequent to a stimulus as a measure of frugality of the policy:

$$J_{fru}(t_k) \triangleq \left\|\mathbf{x}(t_k^+) - \mathbf{x}(t_k)\right\|_2^2. \tag{2.8}$$

A frugal policy is one that achieves a low value of J_{fru}, thus implying that new stimuli are encoded using slots that are already being used. However, such a policy also likely incurs a high degree of overwriting.

2.3.3 Optimization Problem. Consolidating the above principles together, we define the memory resource allocation optimization problem as

$$\begin{aligned} \min_{\mathbf{b}(t_k)} \quad & J(t_k) = \lambda_e J_{enc}(t_k) + \lambda_f J_{fru}(t_k) \\ \text{subject to} \quad & \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}(t)\mathbf{w}_{\mathbf{x}}\mathbf{u}(t), \end{aligned} \tag{2.9}$$

$$\mathbf{z}(t) = \mathbf{C}\mathbf{x}(t),$$
$$\mathbf{x}(t_k^+) = \mathbf{x}(t_k) + \mathbf{b}(t_k)\mathbf{w}_\mathbf{x}\beta_k,$$
$$\lambda_e \geq 0, \lambda_f > 0,$$
$$\mathbf{w}_\mathbf{x}\beta_k > 0.$$

Here the parameters λ_f and λ_e respectively weight the relative importance of frugality and minimizing error, and we assume $\mathbf{w}_\mathbf{x}\beta_k > 0$ to guarantee strict convexity of $J(t)$ (see the appendix). In this way, any solution(s) to equation 2.9 outline(s) dynamical means to store and maintain signals and information in the slot-based network model.

2.4 Slot Utilization as a Metric for Memory Retention. In addition to J_{fru}, we use a second quantity to characterize how well a policy is retaining stimuli over time. Specifically, we define the *slot overlap*,

$$R(x, y) \triangleq \frac{|x|^T |y|}{\|x\|_2 \left\|y\right\|_2}, \tag{2.10}$$

which is simply the cosine similarity between the entry-wise absolute value of x and y. For example, $R(x(t_k^+), x(t_j^+)) \approx 0$ would imply that the kth and jth stimuli were gated onto a distinct set of network slots. We would interpret such an occurrence to imply that the memory of the kth stimulus has not been disrupted by the gating of the jth stimulus.

3 Results

3.1 Resource-Efficient Gating Is Achieved by Lateral Inhibition. From the definition of our optimization of mnemonic principles, we can now derive the optimal gating policy.

Proposition 1. *Given the model network with N slots with dynamics specified in equation 2.1, input train (see equation 2.2) of stimuli with dimension $d < N$ and decoder matrix $\mathbf{C}$, the policy to optimize the cost function (see equation 2.9) is given by*

$$\mathbf{b}(t) = \mathbb{1}_T(t)\mathbf{b}(t_k), \tag{3.1}$$

where $\mathbb{1}_T(t)$ is the indicator function over the set of stimulus arrival times $T = \{t_1, t_2, \ldots\}$ and

$$\mathbf{b}(t_k) = -\frac{\lambda_e}{\mathbf{w}_\mathbf{x}\beta_\mathbf{k}}\Gamma(\lambda_f, \lambda_e)^{-1}\mathbf{C}^T\big(\mathbf{C}\mathbf{x}(t_k) - \beta_k\big),$$
$$\Gamma(\lambda_f, \lambda_e) = \lambda_f\mathbf{I} + \lambda_e\mathbf{C}^T\mathbf{C}.$$

Proof. The proof proceeds by first establishing that cost function $J(t)$: $\mathbb{R}^N \to \mathbb{R}^+$ is a convex function, and hence a unique minimizer $\mathbf{b}(t)$ of $J(t)$ exists. Subsequently, the minimizer is obtained analytically from the gradient of the cost with respect to $\mathbf{b}(t)$. We note that the crucial assumption $N > d$ implies the existence of $\Gamma(\lambda_f, \lambda_e)^{-1}$. Further details are in the appendix. □

Of note in proposition 1, *inaction error* $\mathbf{C}\mathbf{x}(t_k) - \beta_k$ explicitly measures the difference between the original stimulus β_k and that which is decoded from the network at time t_k. Translating this error from the stimulus domain into the state domain via $\Gamma(\lambda_f, \lambda_e)^{-1}\mathbf{C}^T$, the network is prescribed with the exact action needed to accurately encode β_k, modulated by the relative importance of frugality as well as the embedded decoder.

Clearly, the decoder is central to the gating policy, and there presumably exists some sensitivity to the choice of $\mathbf{C}$, as discussed below. Equipped with this solution, we now provide the closed-loop dynamics for our memory network.

Proposition 2. *Given a time-varying system described by a stable network (see equation 2.1) and decoder (see equation 2.5), stimulus train (see equation 2.2) with arrival times* $T = \{t_1, t_2, \ldots\}$, *and objective parameters* $\lambda_f > 0, \lambda_e \geq 0$, *the state evolution,*

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbb{1}_T(t)\lambda_e \xi(t), \tag{3.2}$$

is the optimal solution to the resource allocation problem (see equation 2.9), where

$$\xi(t) = \Gamma(\lambda_f, \lambda_e)^{-1}\mathbf{C}^T\big[\beta_k - \mathbf{C}\mathbf{x}(t)\big],$$
$$\Gamma = \lambda_f \mathbf{I}_N + \lambda_e \mathbf{C}^T\mathbf{C}.$$

Proof. The proof is straightforward using theorem 1 and the network definition, equation 2.1, and is further outlined in the appendix. □

The above derivations carry several analytical points and interpretations. First, $\mathbf{b}(t)$ is explicitly dependent on the current stimulus β_k as well as the state of the network, adjusting the burden of the new encoding based on feedback. This feedback can be conceptualized in a network motif depicted in Figure 2. Here, a new stimulus triggers a form of lateral inhibition prescribed by $\mathbf{A}$ via translation error $\Gamma(\lambda_f, \lambda_e)^{-1}\mathbf{C}^T$, where λ_e, λ_f regulate the degree of inhibition. Importantly, this network feedback interaction occurs only at the time of encoding. Thus, the mechanism here amounts to one of stimulus-triggered gain control from sensory afferents onto memory units.

In this way, the policy equips the network to make optimal encoding and allocation decisions in an online manner via, in essence, an optimal network architecture and connectivity weights. In particular, there is no need for explicit preallocation of resources or centralized knowledge of each subnetwork's memory burden at all times. However, the optimal solution is clearly

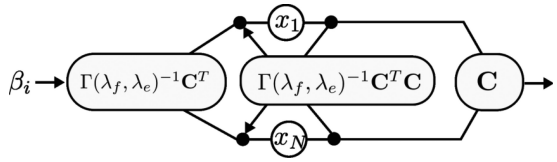

Figure 2: From equation 3.2, the optimal policy can be interpreted as feedback, lateral inhibition from memory units that in essence modulate the gain from sensory projections.

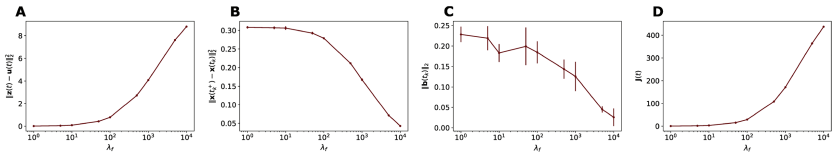

Figure 3: Gating policy optimizes cost objectives. As a function of λ_f, (A) encoding error (J_{enc}), (B) frugality cost (J_{fru}), (C) policy gain $\|\mathbf{b}\|$, and (D) total cost. As networks are constrained to be more frugal (as $\lambda_f \to \infty$), the frugality (see equation 2.8) and policy gain decrease, as expected. Furthermore, the accuracy of encoded stimuli is maximized, approaching 0, under the edge case of the accuracy-myopic policy.

established here offline with knowledge of the overall model. Hence, the issue of how such connectivity could be learned remains unaddressed (see also section 4).

3.2 Greedy Encoding of Stimuli for Accuracy Distributes Resources. Equipped with an analytically optimal gating policy, we sought to understand the behavior of the network as a function of key model parameters. We first considered the extent to which the frugality regularizer (and, by extension, lateral inhibition in the network interpretation) affected slot utilization. In this regard, it is useful to consider two edge cases. When $\lambda_f \to 0$, the policy will prioritize immediate encoding of each stimulus accurately, without regard for any prior stimuli currently being encoded in the network. We refer to this as the greedy or *accuracy-myopic* policy. But when $\lambda_e \to 0$, the policy is biased toward using only those slots that are already being used. Figure 3 verifies the effacy of the policy as the frugality regularizer is modulated. By the nature of our formulation, λ_e and λ_f effectively work inversely from each other, such that $\lambda_f \to 0$ is akin to $\lambda_e \to \infty$ and vice versa. Thus, we illustrate only results of modulating λ_f.

Before proceeding, it is important to note that the derived policy has a degeneracy for the limiting case, $\lambda_f = 0$. In fact, the cost is no longer strictly convex in this case, and therefore there is no unique policy when $\lambda_f = 0$.

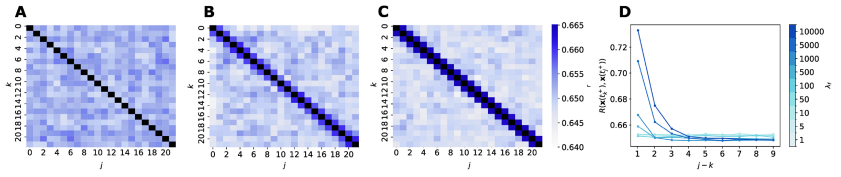

Figure 4: Network slot utilization modulated by frugality weight. The relative importance of encoding frugality and accuracy was modulated via weights $\lambda_e = 1$ and varying λ_f. Information retention in paired postdecision states $\mathbf{x}(t_k^+), \mathbf{x}(t_j^+)$ for all j, k were measured via slot overlap (see equation 2.10) for the (A) accuracy-myopic policy ($\lambda_f = 1$), (B) moderately frugal policy ($\lambda_f = 500$), and (C) highly frugal policy ($\lambda_f = 1000$). With the introduction of the frugality term, immediately adjacent states ($|j - k| = 1$) have similar slot utilization, but demands of competing stimuli ultimately shift the state further from $\mathbf{x}(t_k^+)$. (D) The effect on information retention via slot overlap for $\lambda_f \in \{1, 100, \ldots, 10000\}$, averaged across $j - k$, as well as samples of $\mathbf{u}(t)$. As $\lambda_f \to \infty$, the policy distributes stimuli across slots. Any overwriting of a stimulus that occurs under a frugal policy is more likely immediately after that stimulus. Black diagonal elements ($j = k$) indicate values of 1.

This is readily understood in terms of the previous discussion about J_{enc} in equation 2.7, corresponding to an overdetermined least squares problem. See the appendix for more details.

We sweep over 10 different values of λ_f in the range $[1, 10{,}000]$, where for each value of λ_f, the corresponding policy was assigned to 6 networks of size $N = 80$ for a total of 60 networks. The performance of the networks under each policy was averaged across unique samples of 100 different $\mathbf{u}(t)$ where stimuli were of dimension $d = 30 < N$, pulled from a uniform distribution $\mathrm{U}_{[-100,100]}$, and with stimulus arrival times $T = \{t_1, t_2, \ldots\}$ kept constant across $\mathbf{u}(t)$. For the purposes of this simulation, the time constants of forgetting were homogeneous across slots and set to $\mu = -25$. Here, since the dynamics of the autonomous network are linear, the timescales of forgetting are governed by the eigenvalues of $\mathbf{A}$, that is, $\mathbf{A} \sim \mathrm{diag}(\mu, \ldots, \mu)$, assuming slots are decoupled in the autonomous regime.

Figure 4 illustrates the slot overlap $R\big(x(t_k^+), x(t_j^+)\big)$ as a function of increasing temporal proximity of stimuli, $j - k$, and for several values of λ_f with fixed $\lambda_e = 1$. We can discern two observations from this figure. First, the λ_f regularizer has the desired effect of encoding adjacent (in time) stimuli in similar slots. However, this effect has an upper plateau at $R\big(x(t_k^+), x(t_j^+)\big) \approx 0.25$ (for $j - k = 1$), suggesting a fundamental limit in the extent of frugality possible for fixed λ_e. Perhaps more interesting is what happens in the myopic policy ($\lambda_f \to 0$). As expected, this policy is less frugal in terms of slot utilization than the regularized policies. However, the extent to which this is true is quite significant, with near-zero correlation

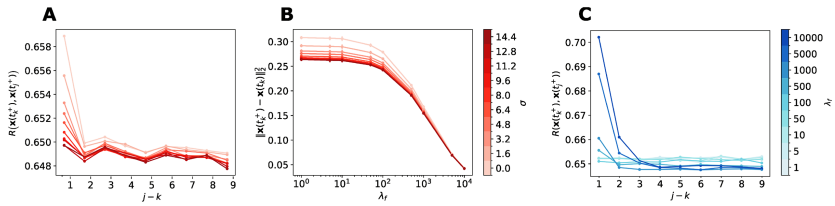

Figure 5: Slot utilization is further modulated by heterogeneity of network timescales. The eigenvalues of system matrix **A** determine the decay rates of the corresponding slots, so we consider the variance σ of the distribution from which we sample these eigenvalues. Generating sample sets of system eigenvalues from uniform distributions with fixed mean $\mu = -25$, we sweep σ in $[0, \sigma_{\max})$ where $\sigma_{\max} = -2\mu\sqrt{1/12}$ was found using the mean and variance of a continuous uniform distribution. All systems also use a fixed decoder matrix **C**. (A) Slot overlap for varying degrees of eigenvalue heterogeneity σ, averaged across $j - k$ and samples of $\mathbf{u}(t)$. A greater spread of eigenvalues promotes the distribution of slots to stimuli, as observed from a decrease in the overlap of adjacent stimuli. (B) Heterogeneity also better minimizes frugality cost than in the singular case where there is no variation of decay rates. Yet with large λ_f, sensitivity to eigenvalue spread decreases. (C) Slot overlap under λ_f modulation with fixed $\sigma = 1.6$, again averaged across k and samples $\mathbf{u}(t)$.

of slots for even $j - k = 1$. Thus, under this policy, stimuli are well distributed over the network. What this means, somewhat paradoxically, is that through focusing on encoding the immediate stimulus at hand, this policy nonetheless mitigates the overwriting of prior stimuli already encoded in the network. In other words, myopically encoding at the current time is also conducive to retaining stimuli from the past.

3.3 Heterogeneity of Memory Slot Timescales Encourages Retention. In a second numerical study, we investigated the impact of network slot timescale heterogeneity on optimal resource allocation. While **A** does not explicitly appear in equation 3.1, the effect of the timescales is nonetheless manifest via the presence of $\mathbf{x}(t_k)$ in the specification of $\mathbf{b}(t_k)$.

With this in mind, we generated more network instances with $N = 80$ subnetwork slots, modulating the spread of subnetwork timescales—the variance σ of the distribution from which we sample eigenvalues of **A**. Specifically, we generated eigenvalues from a uniform distribution with mean $\mu = -25$ and varying $\sigma \geq 0$ (rejecting any samples resulting in positive eigenvalues) and fix cost weights $\lambda_e = \lambda_f = 1$. As before, each network had a fixed **C** and bias parameter $\mathbf{w}_\mathbf{x}$, and we co-evolved them against several $\mathbf{u}(t)$ of dimension $d = 30$.

As observed in Figure 5, networks with heterogeneous timescales will distribute stimuli to a greater degree than those without, as evidenced by a

drop in $R(x(t_k^+), x(t_j^+))$ for temporally adjacent stimuli. That is, heterogeneity of timescales equips the network to be more frugal. Of note, this effect occurs independent of the frugality regularizer, including the case of $\lambda_f = 1$, implying that the variability in decay rates across the network will implicitly spread the encoding workload even when the objective is effectively myopic with respect to error. In the case where the network is constrained to change the state as minimally as possible (as $\lambda_f \to \infty$), the flexibility offered by varying timescales is overshadowed by the cost accrued and the sensitivity of the network to the spread of its timescales diminishes.

4 Discussion and Conclusion

We examined the problem of resource allocation in large networks in the face of intermittent novel afferent stimuli. In particular, we outlined key concepts underlying resource allocation principles of memory and defined corresponding mathematical objectives. Building from a canonical linear dynamical system, we derived an online policy that optimizes memory objectives and is able to be enacted in large networks via a network lateral inhibition-based architecture. The effectiveness of this policy in terms of information retention and overall memory, as well as the degree and form of lateral inhibition, are regulated by the parameterization of the network via cost weights λ_f, λ_e and timescales of slot decay. Our work highlights how online, dynamical mechanisms in brain networks may enact specific memory objectives toward higher-level functions.

4.1 Greedy Policy Retains Memories More Steadily into the Future. Compared to greedy or myopic policies, the most frugal policies produce states that are much more highly correlated to the original encoding state for the first few subsequent memoranda. This effect is expected, since it is in essence the mathematical purpose of the frugality regularizer. However, the slot overlap associated with the frugal policy does cross over the greedy policy several stimuli into the future (see Figure 4), ultimately yielding encoding states that are less correlated to the original after only about four subsequent stimuli. So there is a subtle effect of attenuated overwriting at long latencies for the frugal policies. On the other hand, the greedy policies yield encoding states that do not correlate strongly at all to the original state that stays within a relatively small range going forward, again implying that this policy tends to distribute stimuli across mnemonic resources.

4.2 Intrinsic Benefits to Timescale Heterogeneity. Neural activity in memory areas such as dorsolateral prefrontal cortex displays elevated activity during delay periods, though there is variability in the timescales with which such activity is maintained (Miller et al., 1996; Shafi et al., 2007).

However, despite this variability, stable memory representations can be decoded with ensuing behavioral robustness. Our results here indicate a potential benefit to variable timescales of local representations in terms of longitudinal resource management, as opposed to acutely in the encoding of a single item. That is, variability promotes the distribution of resources to incoming stimuli, thus enabling function in sequential, multi-item scenarios.

4.3 Stimulus Relevance and Encoding Bias. By construction, we assume every stimulus in an input signal $\mathbf{u}(t)$ is task relevant, that is, the information represented via each β_k is meaningful to some functional goal and thus must be encoded within the network. Thus, we do not explicitly consider higher-level attention-like processes that might dictate the relative importance of different stimuli in a sequence. Nonetheless, our formulation could be generalized to interface with such processes via the weighting parameter λ_e. For instance, λ_e could act as an online filter such that the magnitude of λ_e dictates the relative relevance of a stimulus and the respective degree to which the network prioritizes its encoding. Such online updates of accuracy weights would further enable the network to utilize and balance different aspects of accuracy- versus frugality-focused policies.

4.4 Mathematical Interpretations. We note that the cost function is strictly convex; hence, a unique, global policy exists when we assume $\lambda_f \neq 0$. In this scenario, the cost bears some resemblance to problems in efficient neural coding, with an L_2 regularizer on a least-squares cost. The mathematical tractability of this problem enables the analytical specification of $\mathbf{b}(t)$. Thus, $\lambda_f = 0$ implies $J(t)$ is only convex, resulting in a nonunique optimal policy (see the appendix for more details).

4.5 Network Performance Preserved across Dimensions. A crucial assumption in our work is the underdetermined nature of the decoding problem specified by $\mathbf{z}(t) = \mathbf{C}\mathbf{x}(t)$ (i.e., $d < N$). In the analysis, we have fixed these parameters, though it is also of interest to consider the robustness of the results to different signal dimensions and network sizes. However, a direct comparison across different values of d or N brings about certain mathematical limitations. Foremost, it is important to note that the slot overlap $R\big(x(t_j^+), x(t_k^+)\big)$ is inherently sensitive to the similarity of the original stimuli, since our gating policy is, fundamentally, a linear transformation. Thus, with all other variables equal, small-dimension stimuli will tend to produce higher overlap with one another on average than those with higher dimension. Furthermore, the value of the norm in J_{fru} grows as the number of elements (i.e., N) in $\mathbf{x}(t)$ increase, thus dampening the effect of λ_f. Thus, networks of different sizes will require different frugality regularizers in order

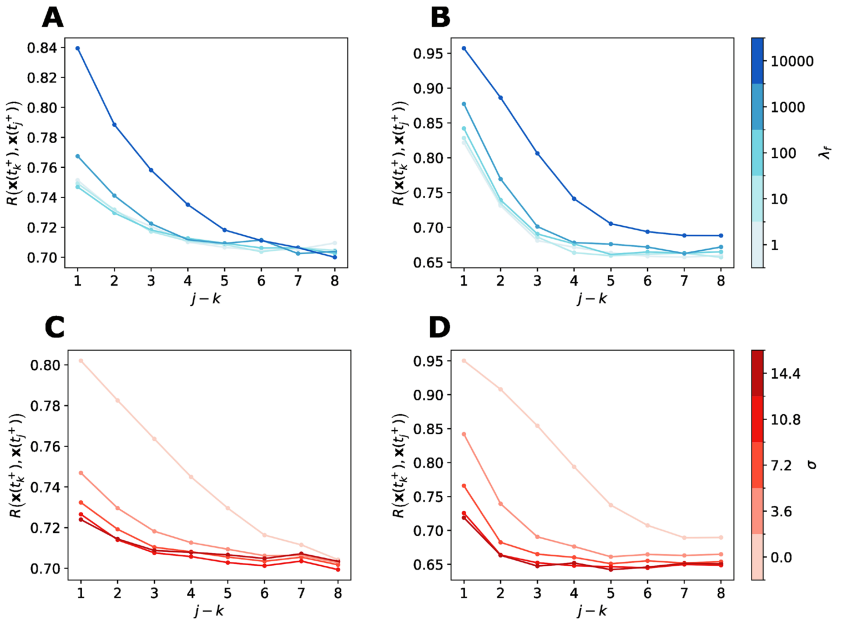

Figure 6: Slot utilization is consistent for different ratios of dimension. Generating sample sets of $\mathbf{u}(t)$ for signal dimensions $d \in \{4, 30\}$, we run simulations with $N \in \{80, 160\}$ with fixed eigenvalue mean $\mu = -25$ and compare network performance for the different degrees of encoding freedom prescribed by the ratio of d to N. We average within each sample set and again across subsequent encoding steps $j - k$ for each stimulus within each $\mathbf{u}(t)$, comparing the effect of d and N modulated by spread in subnetwork decay rates σ and relative prioritization of frugality via λ_f. (A) Slot overlap across frugality with moderate spread of decay rates $\sigma = 3.6$ for the same network size $N = 80$ as the simulations outlined above but with small $d = 4$, as well as for (B) the same signal dimension $d = 30$ as above but with large $N = 160$. (C) We similarly consider slot overlap across decay rate heterogeneity σ with moderate frugality weight $\lambda_f = 100$ for $N = 80$ and small $d = 4$ and (D) $d = 30$ and large $N = 160$ again. Distribution of stimuli across network slots is still encouraged by large λ_f and σ, and thus the relative network behavior is consistent for different d and N.

to achieve comparable overlap relationships. With these points in mind, we can nonetheless establish that the relative behavior of the policy with respect to λ_f and σ is robust to changing d and N. Figure 6 replicates our previous figures for both a larger N and a smaller d, where we again see, as expected, that large λ_f promotes frugality of immediate stimuli and greater heterogeneity reduces overlap. Further expected is the attenuation of λ_f's influence for larger networks (see Figures 6B and 6D), particularly in terms of retention through successive encodings.

4.6 Network Interpretations and Limitations. The policy is enacted as a fixed mechanism across the network. As described, there is correspondence between this mechanism and certain canonical motifs that are believed to be overexpressed in neural circuits, namely, those of lateralized inhibition and gain control. In essence, memory units feed back their state to modulate the projection of sensory afferent signals onto slots. The question, of course, is how such connectivity could be learned in biologically plausible ways. Certainly the gradient of the cost could be used to define a descent rule for incremental update of network parameters through canonical learning methods such as backpropogation through time (Rumelhart et al., 1985). However, a conceptual issue would still arise regarding the extent to which global information would be required to enact such a rule, particularly if connectivity is to be updated in an online manner. Work by Kafashan and Ching (2017) and Murray (2019) among others has explored local learning implementations (i.e., those dependent only on local or pairwise interaction between neural units) for cost functions similar to the ones we consider here. However, these usually assume approximation of global information or large timescale separation in learning the connectivity versus enacting the optimized functions in question. For now, we leave learning the network connectivity as an open question, with our primary findings extending to an interpretation of the network in its optimized form.

A related limitation in our model is the presumption of a fixed decoder $\mathbf{z}(t)$ in the specification of $J(t)$. This assumption manifests in the matrix $\mathbf{C}$ directly parameterizing the allocation policy, ultimately influencing how network slots evolve and interact with one another. We have used $\mathbf{C}$ in our study as a theoretical construct that implies memory slots are "read out" into a lower-dimensional space, such that there is some slot redundancy. The exact specification of $\mathbf{C}$ and degree of correlation with the stimuli to encode is arbitrary here, modulo rank considerations. Relaxing these assumptions to consider more general decoding schemes and neuronal-level encoding is left for future study.

In our framework, network state $\mathbf{x}(t)$ is defined as some abstract representation of the network's activity. Specifically, values in the subnetwork slots represent activity to be decoded into information, with such activity eventually decaying to zero without further input in a sense of "forgetting" the stored information. In this way, $\mathbf{x_i}(t) = 0$ is interpreted as a lack of decodable information within the ith slot, or a return to background activity. True, spontaneous background activity as seen in empirical studies is not modeled.

Appendix

A.1 Convexity of $J(t)$. To guarantee a global solution to the optimization problem, equation 2.9, the cost function $J(t_k)$ must be convex. We now proceed to show via second-order methods that $J(t_k)$ is indeed convex and

is in fact strictly convex, implying the global solution is unique. Due to the discrete nature of the input signal $\mathbf{u}(t)$, we first specify that

$$\begin{aligned}\frac{d}{d\mathbf{b}(t)}\mathbf{x}(t_k^+) &= \frac{d}{d\mathbf{b}(t)}\mathbf{b}(t_k)\mathbf{w}_\mathbf{x}\beta_k \\ &= \mathbf{w}_\mathbf{x}\beta_k,\end{aligned}$$

and can now compute the gradient of $J(t_k)$ with respect to $\mathbf{b}(t_k)$. To begin, consider the gradients of the two cost terms:

$$\begin{aligned}\nabla J_{enc}(t_k) &= \frac{d}{d\mathbf{b}(t)}\|\mathbf{C}\mathbf{x}(t_k^+) - \beta_k\|_2^2 \\ &= 2\big[\mathbf{C}\mathbf{x}(t_k^+) - \beta_k\big]^T \frac{d}{d\mathbf{b}(t)}\big[\mathbf{C}\mathbf{x}(t_k^+) - \beta_k\big] \\ &= 2\big[\mathbf{C}\mathbf{x}(t_k) - \beta_k\big]^T \mathbf{C}\mathbf{w}_\mathbf{x}\beta_k + 2\mathbf{b}(t_k)^T\mathbf{C}^T\mathbf{C}(\mathbf{w}_\mathbf{x}\beta_k)^2, \\ \nabla J_{fru}(t_k) &= \frac{d}{d\mathbf{b}(t)}\|\mathbf{x}(t_k^+) - \mathbf{x}(t_k)\|_2^2 \\ &= \frac{d}{d\mathbf{b}(t)}\|\mathbf{b}(t_k)\mathbf{w}_\mathbf{x}\beta_k\|_2^2 \\ &= 2\mathbf{b}(t_k)^T(\mathbf{w}_\mathbf{x}\beta_k)^2.\end{aligned}$$

Therefore, the full gradient of $J(t_k)$ is given by

$$\begin{aligned}\nabla J(t_k) &= \lambda_e\frac{d}{d\mathbf{b}(t)}J_{enc}(t_k) + \lambda_f\frac{d}{d\mathbf{b}(t)}J_{fru}(t_k) \\ &= 2\lambda_e\big[\mathbf{C}\mathbf{x}(t_k) - \beta_k\big]^T\mathbf{C}(\mathbf{w}_\mathbf{x}\beta_k) + 2\mathbf{b}(t_k)^T\big[\lambda_e\mathbf{C}^T\mathbf{C} + \lambda_f\mathbf{I}_N\big](\mathbf{w}_\mathbf{x}\beta_k)^2, \quad \text{(A.1)}\end{aligned}$$

and the Hessian easily follows:

$$\mathrm{H}J(t_k) = 2(\mathbf{w}_\mathbf{x}\beta_k)^2\big[\lambda_e\mathbf{C}^T\mathbf{C} + \lambda_f\mathbf{I}_N\big].$$

When we assume $\mathbf{w}_\mathbf{x}\beta_k > 0$, we see that $\mathrm{H}J(t_k)$ is strictly convex, as $\mathbf{C}^T\mathbf{C}$ and $\mathbf{I}_N$ are positive semidefinite and positive-definite matrices, respectively. However, when we consider the cost function without the regularizer (i.e., assume $\lambda_f = 0$), the gradient and Hessian respectively become

$$\begin{aligned}\nabla J(t_k)|_{\lambda_f=0} &= 2\lambda_e\big[\mathbf{C}\mathbf{x}(t_k) - \beta_k\big]^T\mathbf{C}(\mathbf{w}_\mathbf{x}\beta_k) + 2\lambda_e\mathbf{b}(t_k)^T\mathbf{C}^T\mathbf{C}(\mathbf{w}_\mathbf{x}\beta_k)^2, \\ \mathrm{H}J(t_k)|_{\lambda_f=0} &= 2\lambda_e\|\mathbf{C}\mathbf{w}_\mathbf{x}\beta_k\|_2^2,\end{aligned}$$

causing the cost function to be simply convex if no further assumptions are given, guaranteeing a global but not necessarily unique solution to the optimization problem.

A.2 Derivation of Policy and Closed-Loop System. To find an analytical solution to the optimization problem, equation 2.9, we seek the minimizer of $J(t_k)$. Setting $\nabla J(t_k) = 0$ from equation A.1, we easily arrive at the solution in equation 2.9,

$$\mathbf{b}(t_k) = -\frac{\lambda_e}{\mathbf{w_x}\beta_\mathbf{k}}\left[\lambda_e \mathbf{C}^T\mathbf{C} + \lambda_f \mathbf{I}_N\right]^{-1}\mathbf{C}^T\left(\mathbf{C}\mathbf{x}(t_k) - \beta_k\right).$$

A.3 Closed-Loop Dynamics. We can now simply plug the derived policy into our original network expression, equation 2.1, noting that $\delta(t - t_k)$ and $\mathbb{1}_T(t)$ are effectively redundant and conveniently collapse into $\mathbb{1}_T(t)$:

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \\
&= \mathbf{A}\mathbf{x}(t) + \mathbb{1}_T(t)\mathbf{b}(t_k)(\mathbf{w_x}\beta_k) \\
&= \mathbf{A}\mathbf{x}(t) + \mathbb{1}_T(t)\lambda_e\left[\lambda_f \mathbf{I}_N + \lambda_e \mathbf{C}^T\mathbf{C}\right]^{-1}\mathbf{C}^T\left[\beta_k - \mathbf{C}\mathbf{x}(t_k)\right] \\
&= \mathbf{A}\mathbf{x}(t) + \mathbb{1}_T(t)\lambda_e\Gamma(\lambda_f, \lambda_e)^{-1}\mathbf{C}^T\left[\beta_k - \mathbf{C}\mathbf{x}(t_k)\right].
\end{aligned}$$

Acknowledgments

This work was partially supported by grants 1653589 from the National Science Foundation, W911NF-21-1-0312 from the U.S. Department of Defense, and R01EB028154 from the U.S. National Institutes of Health.

References

Baddeley, A. (1992). Working memory. *Science*, *255*, 556–559. 10.1126/science.1736359

Bialek, W., Steveninck, R., & Tishby, N. (2007). *Efficient representation as a design principle for neural coding and computation*. arXiv:0712.4381.

Carrillo-Reid, L., Miller, J., Hamm, J., Jackson, J., & Yuste, R. (2015). Endogenous sequential cortical activity evoked by visual stimuli. *Journal of Neuroscience*, *35*(6), 8813–8828. 10.1523/JNEUROSCI.5214-14.2015

Chung, S., & Abbott, L. (2021). Neural population geometry: An approach for understanding biological and artificial neural networks. *Current Opinion in Neurobiology*, *70*(10), 137–144. 10.1016/j.conb.2021.10.010

Cowan, N., Elliott, E., Scott Saults, J., Morey, C., Mattox, S., Hismjatullina, A., & Conway, A. (2005). On the capacity of attention: Its estimation and its role in working memory and cognitive aptitudes. *Cognitive Psychology*, *51*(8), 42–100. 10.1016/j.cogpsych.2004.12.001

Funahashi, S., & Kubota, K. (1994). Working memory and prefrontal cortex. *Neuroscience Research*, *21*(11). 10.1016/0168-0102(94)90063-9

Ghazizadeh, E., & Ching, S. (2021). Slow manifolds within network dynamics encode working memory efficiently and robustly. *PLOS Computational Biology*, *17*(9), e1009366. 10.1371/journal.pcbi.1009366

Glahn, D., Bearden, C., Cakir, S., Barrett, J., Najt, P., Serap Monkul, E., . . . Soares, J. (2006). Differential working memory impairment in bipolar disorder and schizophrenia: Effects of lifetime history of psychosis. *Bipolar Disorders*, *8*(2006), 117–123. 10.1111/j.1399-5618.2006.00296.x

Gorgoraptis, N., Catalao, R., Bays, P., & Husain, M. (2011). Dynamic updating of working memory resources for visual objects. *Journal of Neuroscience 31*(6), 8502–8511. 10.1523/JNEUROSCI.0208-11.2011

Joseph, S., Teki, S., Kumar, S., Husain, M., & Griffiths, T. (2016). Resource allocation models of auditory working memory. *Brain Research*, *1640*(6), 183–192. 10.1016/j.brainres.2016.01.044

Kafashan, M., & Ching, S. (2017). Recurrent networks with soft-thresholding nonlinearities for lightweight coding. *Neural Networks*, *94*(10), 212–219. 10.1016/j.neunet.2017.07.008

Kessler, Y., & Meiran, N. (2006). All updateable objects in working memory are updated whenever any of them are modified: Evidence from the memory updating paradigm. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *32*(5), 570–585. 10.1037/0278-7393.32.3.570

Lara, A., & Wallis, J. (2015). The role of prefrontal cortex in working memory: A mini review. *Frontiers in Systems Neuroscience*, *9*(12). 10.3389/fnsys.2015.00173

Lee, H., Choi, W., Park, Y., & Paik, S. (2020). Distinct role of flexible and stable encodings in sequential working memory. *Neural Networks*, *121*(1), 419–429. 10.1016/j.neunet.2019.09.034

Lee, J., & Park, S. (2005). Working memory impairments in schizophrenia: A meta-analysis. *Journal of Abnormal Psychology*, *114*, 599. 10.1037/0021-843X.114.4.599

Luck, S., & Vogel, E. (2013). Visual working memory capacity: From psychophysics and neurobiology to individual differences. *Trends in Cognitive Sciences*, *17*(8), 391–400. 10.1016/j.tics.2013.06.006

Miller, E., Erickson, C., & Desimone, R. (1996). Neural mechanisms of visual working memory in prefrontal cortex of the macaque. *Journal of Neuroscience*, *16*, 5154–5167. 10.1523/JNEUROSCI.16-16-05154.1996

Murray, J. (2019). Local online learning in recurrent networks with random feedback. *eLife*, *8* (5).

Murray, J., Bernacchia, A., Roy, N., Constantinidis, C., Romo, R., & Wang, X. (2017). Stable population coding for working memory coexists with heterogeneous neural dynamics in prefrontal cortex. *Proceedings of the National Academy of Sciences*, *114*(1), 394–399. 10.1073/pnas.1619449114

Pina, J., Bodner, M., & Ermentrout, B. (2018). Oscillations in working memory and neural binding: A mechanism for multiple memories and their interactions. *PLOS Computational Biology*, *14*(11), e1006517. 10.1371/journal.pcbi.1006517

Rouder, J., Morey, R., Morey, C., & Cowan, N. (2011). How to measure working memory capacity in the change detection paradigm. *Psychonomic Bulletin Review*, *18*(4), 324–330. 10.3758/s13423-011-0055-3

Rumelhart, D., Hinton, G., & Williams, R. (1985). *Learning internal representations by error propagation*. Defense Technical Information Center.

Shafi, M., Zhou, Y., Quintana, J., Chow, C., Fuster, J., & Bodner, M. (2007). Variability in neuronal activity in primate cortex during working memory tasks. *Neuroscience*, *146*, 1082–1108. 10.1016/j.neuroscience.2006.12.072

Spaak, E., Watanabe, K., Funahashi, S., & Stokes, M. (2017). Stable and dynamic coding for working memory in primate prefrontal cortex. *Journal of Neuroscience*, *37*(7), 6503–6516. 10.1523/JNEUROSCI.3364-16.2017

Stokes, M., Kusunoki, M., Sigala, N., Nili, H., Gaffan, D., & Duncan, J. (2013). Dynamic coding for cognitive control in prefrontal cortex. *Neuron*, *78*, 364–375. 10.1016/j.neuron.2013.01.039

Wojtak, W., Coombes, S., Avitabile, D., Bicho, E., & Erlhagen, W. (May 2023). Robust working memory in a two-dimensional continuous attractor network. *Cognitive Neurodynamics*. 10.1007/s11571-023-09979-3

Ye, C., Hu, Z., Li, H., Ristaniemi, T., Liu, Q., & Liu, T. (2017). A two-phase model of resource allocation in visual working memory. *Journal of Experimental Psychology: Learning Memory and Cognition*, *43*(10), 1557–1566. 10.1037/xlm0000376

Zaccarian, L. (2009). Dynamic allocation for input redundant control systems. *Automatica*, *45*(6), 1431–1438. 10.1016/j.automatica.2009.01.013

Zhang, X., Yi, H., Bai, W., & Tian, X. (2015). Dynamic trajectory of multiple single-unit activity during working memory task in rats. *Frontiers in Computational Neuroscience*, *9*.

Received September 7, 2023; accepted January 10, 2024.