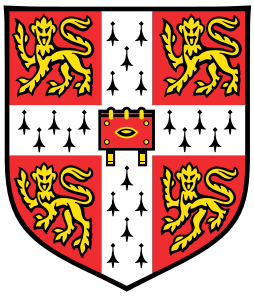
Visual Working Memory in Neural Networks



Derek Jinyu Dong

Supervisors: Prof. Yashar Ahmadian

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Master of Engineering

Emmanuel College June 2025

Table of contents

Nomenclature

Other Symbols

Maximum set size M: The maximum number of items can be received by a model

Mean-field population Z: Approximate number of biological neurons represented by a single unit in the model.

Number of units N: The number of artifitial neurons in the recurrent neural network

Set size m: The number of items to be remembered

Acronyms / Abbreviations

VWM Visual Working Meemory

1 Introduction

Context and motivation: Visual Working Memory (VWM) is the brain's limited-capacity buffer that allows us to maintain visual information over short delays and use it to guide behaviour [5, 16]. Capacity limits in VWM correlate strongly with general intelligence and academic performance, earning it the reputation of a fundamental "cognitive bottleneck"[11, 9]. Behavioral tasks such as change detection, sequential recall, and (most relevant here) single-report delayed estimation reveal that recall precision declines and the distribution of errors acquires heavy tails as the number of to-be-remembered items increases[34, 3, 31].

Classical slot[34] and continuous-resource models[3] quantify how errors scale with set size but stop short of identifying their neural origins. Variable-precision[31] and psychophysical-similarity frameworks[25, 32] go further, yet both remain *descriptive*: they assume either hand-crafted resource limits or a static representational geometry without explaining how such constraints emerge from circuit dynamics.

Mechanistic proposals such as ring-attractor networks [33, 8]or multi-ring architectures [6] capture persistent activity, but either enforce hand-crafted architectures, leaving open questions about metabolic cost and mixed selectivity[22, 30].

Objectives: This project aims to develop a biologically plausible recurrent neural network (RNN) model that performs VWM tasks under metabolic constraints, thereby offering mechanistic insight into the origins of working memory limitations. Specifically, we ask:

- Can known cortical principles, such as excitatory-only external drive and Dale's law, be integrated into task-optimized RNNs without sacrificing performance?
- Does divisive normalization emerge spontaneously in trained networks, both at the population and single-neuron levels?
- Can a single, unified network recapitulate key behavioral patterns—such as error variance scaling and heavy-tailed distributions—while also exhibiting emergent mixed selectivity and continuous attractor dynamics?

Approaches: To this end, we construct and train RNNs on a biologically relevant variant of the Delayed Estimation Task. The networks are constrained by architectural principles (e.g., sign-consistent synaptic weights, heterogeneous time constants) and optimized using gradient-based learning. We systematically evaluate how hyperparameters such as network size, input strength, and process noise influence behavior. Finally, we analyze the internal dynamics of the trained networks to examine the emergence of divisive normalization, continuous attractors, and high-dimensional population codes.

In doing so, this work contributes a unified framework for studying VWM that bridges cognitive theories and biological mechanisms, providing both functional performance and mechanistic interpretability.

Report Organisation: Chapter 2 reviews psychological and computational accounts of VWM. Chapter 3 details the task design, network architecture, and training procedure. Chapter 4 reports behavioural fits, dynamical analyses, and emergent computations. Chapter 5 discusses implications, limitations, and future directions, followed by appendices on theoretical derivations and code availability.

2 Background and Motivations

2.1 VWM from Psychological Perspectives

Visual Working Memory (VWM) refers to the brain's capacity to actively maintain visual information over brief intervals for ongoing cognitive tasks. It underpins everyday cognition and behavior by allowing us to retain critical information when our gaze shifts or objects are occluded, thereby supporting dynamic tasks such as driving, cooking, reading, or grasping [15, 23]. Individual VWM capacity correlates moderately to highly with intelligence, reasoning ability, and academic achievement, earning it the label of a "cognitive bottleneck" [11, 9].

Beyond its tight coupling with higher-order cognitive functions, VWM exhibits several key features that have made it a focus of study in both psychology and neuroscience. **Highly controllable experiments.** Researchers employ simple visual elements (e.g. colored squares or oriented bars) and precisely manipulate core variables (set size, delay interval, distractor presence) to obtain clear behavioral metrics (accuracy rates or continuous error distributions), ensuring high reproducibility and statistical power. **Cross-scale theoretical integration.** VWM performance can be directly linked across levels of analysis—from psychophysical measurements to neural circuit dynamics and computational frameworks (e.g. variable-precision models, the TCC framework)—making it a gold standard for testing mechanisms that span from neural activity to observable behavior.

2.1.1 Behavioral Tasks in VWM Research

This section introduces several behavioral tasks that probe different facets of VWM. Among these, the single-report, no-cue version of the Delayed Estimation Task is the focus of this project.

- **Change Detection Task** examines capacity limits: observers view an array of m items, followed by a delay, then report whether any item has changed [15].
- **Sequential Recall Task** investigates temporal coding and inter-item interference: participants sequentially report the orientations of m items in the order presented [20]. Some variant of this task is known as the "whole report task".
- **Delayed Estimation Task (single-report, no-cue)** assesses recall precision, error distributions, and resource allocation. In the classic implementation by Bays [2, 1], participants memorize the orientations of m items; after a delay, a test item appears and its location indicates which orientation should be recalled. Participants then adjust the test item to match the remembered orientation (Fig. 2.1A).. The angular recall error is calculated from reported orientation $\hat{\theta}$ and the true orientation θ:

$$\Delta\theta = \text{wrap}(\hat{\theta} - \theta) \in (-\pi, \pi], \tag{2.1}$$

  The error distribution broadens as set size m increases, indicating higher average error, and develops heavier tails, reflecting an increased likelihood of large errors (Fig. 2.1B).

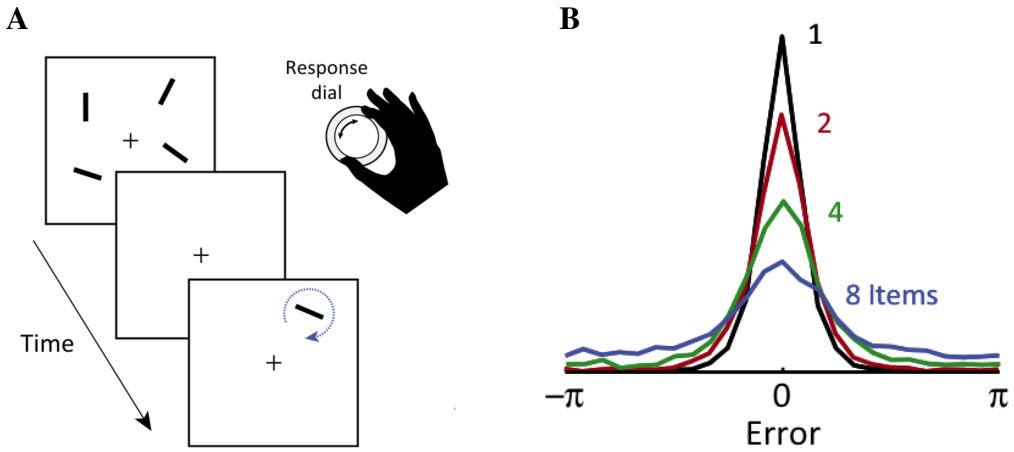


Fig. 2.1 **Delayed-estimation paradigm and empirical results** (adapted from [1]). **A**: Schematic of the trial sequence used to probe visual working-memory precision. **B**: Aggregate error distribution across trials, showing characteristic heavy tails and increased variance with set size.

2.2 Models of VWM: From Abstract to Mechanistic

According to Marr's framework[17], understanding visual working memory (VWM) requires analysis at three distinct but complementary levels. The *computational level* defines what the system does and why—namely, the functional goal of maintaining visual information to guide behavior. The *algorithmic level* specifies how this goal is achieved, through internal representations and rules for processing and transforming information. Finally, the *implementational level* concerns how these processes are physically realized in the brain, in terms of neural circuits and synaptic dynamics.

In this section, we survey existing models of visual working memory and categorize them into two broad classes based on Marr's framework. **Non-mechanistic models**, such as the *discrete slot model*, *continuous resource model*, *variable precision model* and the *nonlinear psychological scaling model (i.e. TCC model)*, primarily address the computational and algorithmic levels. They focus on describing behavioral performance and quantifying memory limitations without committing to specific neural implementations. In contrast, **mechanistic models**—including *ring attractor networks* and *eight-ring networks*—aim to bridge the algorithmic and implementational levels by proposing biologically plausible circuits that instantiate cognitive-level representations and operations.

2.2.1 Slot/Resource Models

Discrete Slot Model

The slot model believes that VWM consists of a fixed number K of discrete storage "slots", each holding one item with perfect fidelity. Classic behavioral estimates place K around seven items[18] or four items[9]. In 2008, Zhang and Luck [34] formalized this account by fitting a mixture model in which each memory item is either stored with fixed precision or not encoded at all (i.e. guessed). The resulting error distribution is a mixture of a von Mises distribution and a uniform distribution:

$$P(\hat{\theta} \mid \theta) = (1-\gamma)\,\mathscr{V}(\hat{\theta}-\theta;\kappa) \;+\; \gamma \mathrm{Uniform}(\hat{\theta}), \tag{2.2}$$

where $\mathscr{V}(\cdot;\kappa)$ is the von Mises distribution with constant concentration parameter κ. γ is the guess rate, which increases as set size m exceeds K. As m grows beyond K, the proportion of guesses γ increases, leading to heavier-tailed error distributions beyond the core von Mises component.

Continuous Resource Model (Equal-precision)

A continuously divisible and limited resource model of VWM, allowing flexible allocation across m items, was first introduced in 2008 (Bays and Husain [4] and 2009 (Bays et al. [3]). A total resource R_{total} is divided equally among all items, and the precision P_i with which item i is stored follows a power-law relationship with its resource

$$R_i = \frac{R_{\text{total}}}{m}, \quad P_i = \alpha R_i^{\beta}, \tag{2.3}$$

where α and β are fitted parameters. The inverse-variance of recall error satisfies

$$\sigma_i^{-2} \propto R_i. \tag{2.4}$$

As set size m increases, R_i (and thus P_i) decreases smoothly, producing broader error distributions. However, a guessing component is still required to capture heavy-tailed error distribution, yielding a combined model similar to equation 2.2:

$$P(\hat{\theta} \mid \theta) = (1-\gamma)\,\mathscr{V}(\hat{\theta}-\theta;\, \kappa(R_i)) \;+\; \gamma \text{Uniform}(\hat{\theta}), \tag{2.5}$$

where $\kappa(R_i)$ now varies with set size, in contrast to the slot model in which it is fixed.

While differing in assumptions about memory architecture—discrete vs. continuous—both models ultimately rely on similar mixture formulations and introduce a uniform guessing component to account for heavy-tailed error distributions. However, both models are abstract cognitive models and lack mechanistic grounding: the uniform component does not arise naturally from a neural process, and neither model specifies how noise or precision emerges from population dynamics.

Variable Precision Model(s)

The classic Variable Precision (VP) model was introduced by van den Berg et al. [31] in 2012. It extends continuous-resource accounts by allowing the precision assigned to each item to fluctuate across items and trials, eliminating the need for an explicit guessing component to explain heavy-tailed errors. The concentration parameter κ of the von Mises distribution is treated as a random variable:

$$\kappa \sim \text{Gamma}(\alpha, \beta), \tag{2.6}$$

$$P(\hat{\theta} \mid \theta) = \int_0^{\infty} \mathscr{V}(\hat{\theta}-\theta;\, \kappa)\,\text{Gamma}(\kappa;\, \alpha, \beta)\,\mathrm{d}\kappa, \tag{2.7}$$

By marginalizing over κ, the VP model produces a mixture of von Mises distributions with variable width, naturally generating heavy-tails. However, the cognitive model does not specify the nature of the underlying resource or the neural origin of variability.

At the neural-population level, Bays [1, 2] reproduced variable precision by combining (i) Poisson spiking noise, whereby the precision of an item scales with its spike count (Appendix A.1.4), and (ii) divisive normalization [7], which caps the total spike budget across items (Appendix A.1.1). As illustrated in Fig. 2.2A–C, this simple mechanism not only generates heavy-tailed recall errors (panel A) but also captures the empirical rise in variance (panel B) and the concomitant drop in kurtosis (panel C) as set size increases.

More recently, Schneegans et al. [24] proposed a Stochastic Sampling model that also assumes neural variability and divisive normalization, but by generalizing Poisson variability, it successfully unifies the classic VP, slot, and continuous resource models.

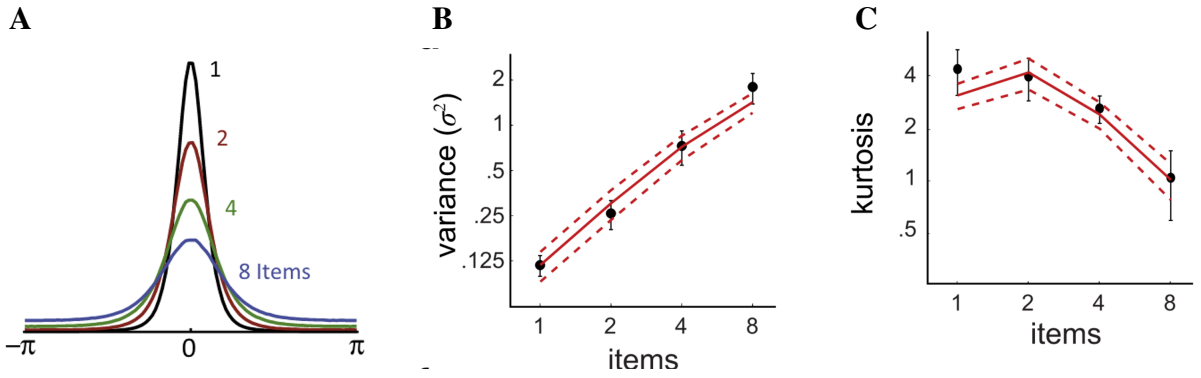


Fig. 2.2 **Performance of the [1, 2] population-coding model**. **A**: The simulated error distribution exhibits pronounced heavy tails, closely matching behavioural data. **B**: Error variance rises with set size, consistent with empirical observations. **C**: Kurtosis decreases as set size increases, reflecting the flattening of distribution tails. Together, these results demonstrate that Poisson variability combined with divisive normalization suffices to reproduce key signatures of variable precision in human recall.

2.2.2 Psychophysical Similarity and Representational Geometry

Schurgin et al. [25] showed that perceived similarity between stimuli decays non-linearly with their physical separation. This psychophysical scaling offers an empirical foundation for the representational-geometry account proposed by Wei and Woodford [32]: in a high-dimensional neural manifold, identical Gaussian noise is warped by non-linear representational distances (RD), producing heavy-tailed error distributions in physical space (see Fig. 2.3). In other words, high-dimensional manifold $\Rightarrow$ non-linear RD $\Rightarrow$ heavy-tailed recall errors, without recourse to discrete guessing or variable resource limits.

Research Gaps

Both the Variable Precision (VP) model and the Psychophysical Similarity framework, have provided valuable insights but remain fundamentally *non-mechanistic*. For instance, the VP model imposes a hand-crafted constraint on the total available firing rate. However, it remains unclear how such a constraint might arise from biological processes within the brain. Similarly, while the Psychophysical Similarity framework captures empirical patterns of memory errors via non-linear similarity functions, it lacks a neurobiological substrate.

Although these two models are theoretically compatible, they originate from distinct assumptions: the VP model attributes behavioral variability to limited resources, whereas the non-linear similarity-based account explains it through representational geometry in a perceptual manifold. A natural question thus emerges: Can a single mechanistic model both implements variable-precision internally and has the curved geometry implied by psychophysical similarity? Although the two have been connected mathematically, our goal is to observe this connection emerge within a task-optimized, biologically grounded neural network.

2.2.3 Bouchacourt's Eight-Rings Model

Bouchacourt and Buschman [6] proposed a mechanistic model of visual working memory (VWM) based on an excitatory-inhibitory (EI) balanced spiking recurrent neural network. As illustrated in Figure 2.4A, the model consists of multiple ring attractors[8] (see appendix A.1.2), each designated to store one target item. These modular rings interact bidirectionally with a central, randomly connected network, which acts as a convergence hub. Within each ring, neurons form a continuous attractor network that maintains item identity via sustained bump activity.

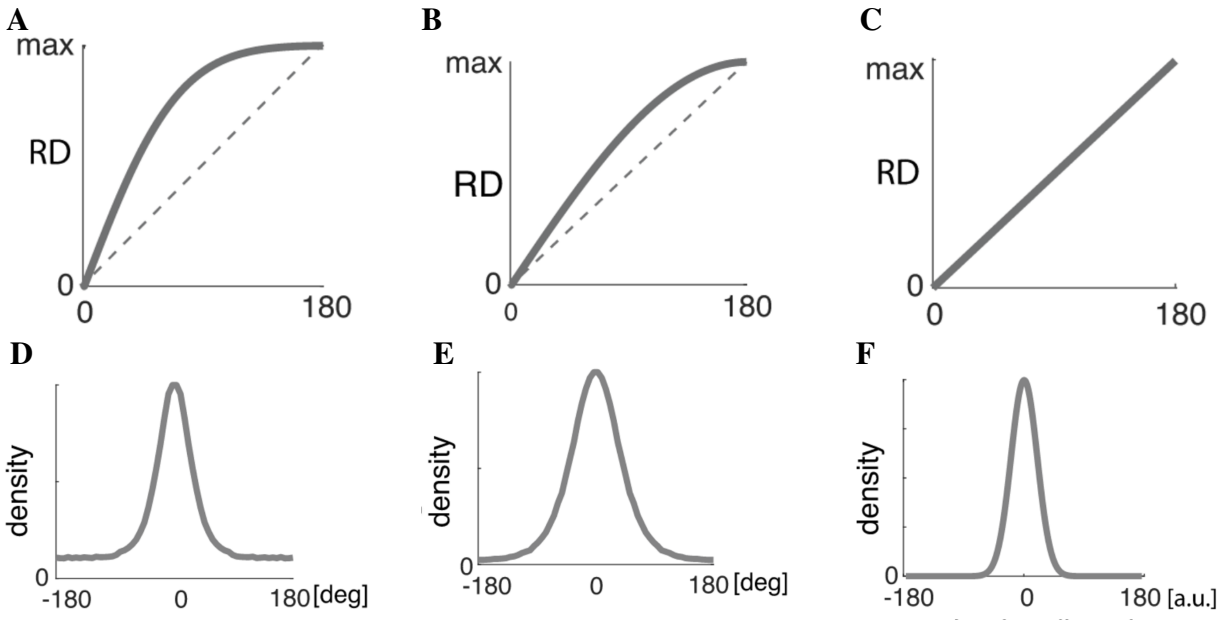


Fig. 2.3 **Geometry decides recall errors** (adapted from 32). In all panels, the horizontal axis denotes the angular distance between the true stimulus θ_0 and the decoded estimate $\hat{\theta}$. **A–C**: Representational–distance (RD) functions for three manifold geometries. The vertical axis shows RD: Euclidean distance between the neural embeddings of two angles. Early RD saturation in the high-dimensional manifold (A) contrasts with the periodic growth of a circular manifold (B) and the linear growth of a flat manifold (C). **D–F**: Simulated error-distribution densities on the same angular axis. Saturated RD in panel A stretches Gaussian encoder noise into a heavy, nearly flat tail (D); circular curvature yields a milder heavy tail (E); a linear manifold preserves Gaussianity (F). Together, these results connect the non-linear psychophysical similarity observed by Schurgin et al. [25] to a geometric mechanism whereby high-dimensional curvature transforms Gaussian neural noise into heavy-tailed recall errors.

We reproduced this model and confirmed its ability to replicate key behavioral and neural phenomena: task accuracy decreases with set size (Figure 2.4B), and neural responses exhibit divisive-normalization-like modulation (Figure 2.4C). Notably, both effects are largely driven by interference across rings mediated by the central random network.

Despite its empirical relevance, the model's architecture raises several concerns. First, the core mechanism—ring-to-ring interference—emerges from a functionally unspecified, untrained random network that contributes no explicit computation. Second, the model enforces strict modularity by assigning each item to a separate ring attractor. As a consequence, neurons within each ring exhibit *pure selectivity* for the preferred items—an observation inconsistent with empirical findings of *mixed selectivity* in prefrontal cortex [22] (see Appendix A.1.3).

2.3 Objectives

Motivated by above limitations, we aim to construct a mechanistic model that can provide explanatory power beyond behavioral fitting. A central question driving this work is whether the **capacity limitations** of VWM can arise naturally from the need to store information under metabolic constraints. To this end, we propose to:

- Develop a biologically plausible recurrent neural network (RNN) that performs VWM tasks under metabolic constraints, prioritizing task performance over behavior fitting.
- Investigate whether divisive normalization (DN) emerges intrinsically in trained networks and contributes to capacity limits.

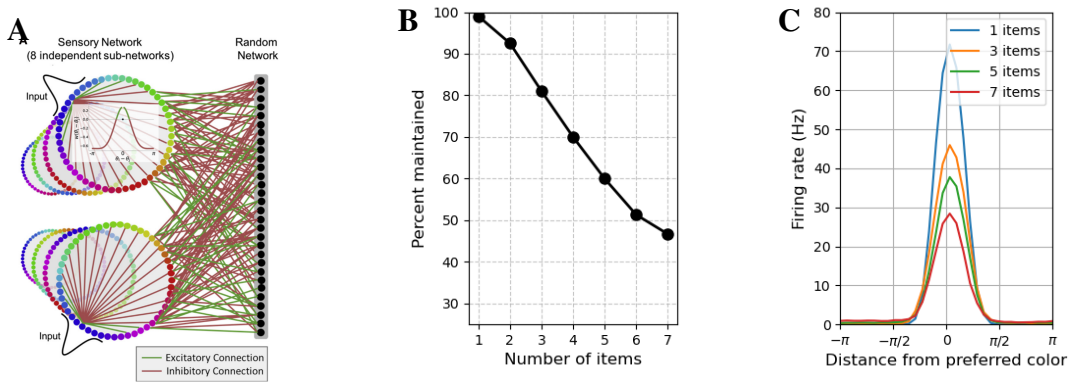


Fig. 2.4 **Architecture and behavior of the Bouchacourt model. A**: Network architecture with eight ring attractors connected via a central random network. **B**: Task accuracy declines with increasing set size. **C**: Divisive normalization–like modulation of tuning curves in the presence of multiple items.

- Leverage the interpretability of mechanistic models to study emergent neural computations and representations, especially those related to representational geometry and resource allocation.

This framework enables us to bridge normative theories and biological mechanisms, offering a unified and mechanistically grounded account of working memory limitations.

3 Constructing a Biologically Plausible RNN Substrate for VWM

3.1 Delayed Estimation Task Setup

In this project, we train RNNs to perform the *Delayed Estimation Task* in the single-report, no-cue variant (see Section 2.1.1).

For each trial with a given *set size* m, m items are randomly selected from the full set of M items to be present (i.e., $a_i = 1$), while the remaining $M - m$ items are marked as absent ($a_i = 0$). Each present item is then assigned a random orientation $\theta_i \in (-\pi, \pi]$, and the corresponding external input vector $\mathbf{h}_t$ is constructed according to the formulation in Section 3.2.3. In this project we set the maximum number of items $M = 10$, and each training batch consists of 512 trials.

Each trial consists of four distinct phases: an initial phase ($T_{\text{init}} = 20$ ms) to stabilize the network dynamics, a stimulus onset phase ($T_{\text{stim}} = 500$ ms) during which external input is delivered, a delay period ($T_{\text{delay}} = 1000$ ms) requiring memory maintenance, and a decoding phase ($T_{\text{decode}} = 500$ ms) during which the maintained memory is read out.

The RNN produces m orientation estimates—one per presented item—based on its internal state during the decoding phase. Although all m items are decoded, this is equivalent to m independent single-report tasks, rather than a whole-report task: there is no temporal order or selective cueing for which item to report.

3.2 Biologically Plausible RNN

3.2.1 Rate-Based Dynamics

We model the activity of a recurrent neural network comprising N unit using first-order rate dynamics [10]. Throughout, we refer to these units as neurons, though they represent *artificial* rather than biological neurons. Each unit corresponds to a *mean-field abstraction*—that is, it approximates the average activity of a local population of Z biological neurons, rather than an individual cell. This coarse-grained view facilitates stable training and captures population-level dynamics relevant to visual working memory.

$$\tau \circ \dot{\mathbf{r}}(t) + \mathbf{r}(t) = \Phi\big(\mathbf{W}\tilde{\mathbf{r}}(t) + \mathbf{B}\mathbf{h}(t)\big) \tag{3.1}$$

$$\hat{\mathbf{u}}(t) = \mathbf{F}\tilde{\mathbf{r}}(t)$$

For numerical simulation with a step size $\Delta t - 10$ ms, Eq. (3.1) is discretized using forward-Euler method [21]:

$$\mathbf{r}_{t+1} = \mathbf{r}_t + \frac{\Delta t}{\tau} \circ \Big(-\mathbf{r}_t + \Phi\big(\mathbf{W}\tilde{\mathbf{r}}_t + \mathbf{B}\mathbf{u}_t\big)\Big), \tag{3.2}$$

$$\hat{\mathbf{u}}_t = \mathbf{F}\tilde{\mathbf{r}}_t \tag{3.3}$$

where the division and Hadamard product ($\circ$) are element–wise. The symbols are defined as follows:

- $\mathbf{r}_t, \tilde{\mathbf{r}}_t \in \mathbb{R}^N$ the latent and observed (noise corrupted) firing rates, respectfully.
- $\mathbf{W} \in \mathbb{R}^{N\times N}$ the recurrent connections, $\mathbf{W}\tilde{\mathbf{r}}(t)$ the recurrent input.
- $\mathbf{B} \in \mathbb{R}^{N\times 3M}$ the input matrix, $\mathbf{h}_t \in \mathbb{R}^{3M}$ the external drive.
- $\mathbf{F} \in \mathbb{R}^{2M\times N}$ the readout matrix, $\hat{\mathbf{u}}_t \in \mathbb{R}^{2M}$ the readout vector.
- $\tau \in \mathbb{R}^N$ A vector of heterogeneous membrane time constants.
- $\Phi\big(\,\cdot\,\big)$ the activation function.

Each component of Eqs. (3.2)–(3.3) is realised by a concrete design choice motivated by neurophysiology and RNN optimisation considerations; the following subsections unpack these choices in detail.

3.2.2 Activation Function

In constructing a biologically plausible recurrent neural network (RNN), the activation function should map synaptic input to a non-negative firing rate in Hertz, with the following properties:

- **Non-negativity:** Output must be ≥ 0.
- **Saturation:** Smoothly approaches a biologically realistic upper bound (e.g., 60 Hz).
- **Soft thresholding:** Gradual firing onset as input increases.
- **Scale matching:** Input and output ranges should be comparable.

We adopt a scaled and shifted hyperbolic tangent:

$$\Phi(x) = 30 \cdot (1 + \tanh(gx - \theta)), \tag{3.4}$$

where g controls steepness and θ sets the threshold. We use:

$$g = 0.14, \quad \theta = 4.2. \tag{3.5}$$

As shown in Figure 3.1, firing remains near-zero below 10 Hz, imposing a soft lower bound on the effective input. Even in minimal set-size conditions ($m = 1$), the input drive $\mathbf{Bh}$ must exceed this threshold to activate the network.

3.2.3 Input Formulation

The model can take up to M items, each has independent orientation. In this section we provide two difference formulations to the external input $\mathbf{h}_t$.

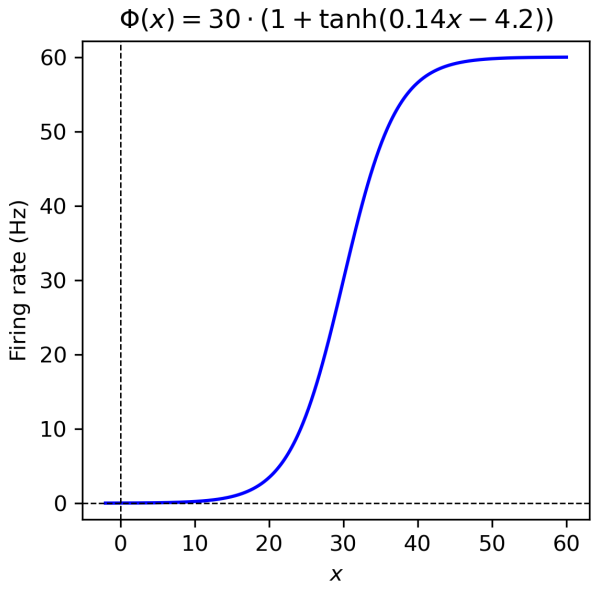


Fig. 3.1 **Activation function used in the RNN.** The function $\Phi(x) = 30 \cdot (1 + \tanh(gx - \theta))$ transforms synaptic input into firing rates (Hz), ensuring biological plausibility. It satisfies key constraints: non-negativity, soft thresholding near $x \approx 30$, and saturation around 60 Hz. For inputs below threshold ($x < 30$), the output remains near zero, creating a regime of effective silence. This behavior is essential for metabolic efficiency and ensures that even at minimal set sizes ($m = 1$), the external drive **Bh** must exceed a minimum level to elicit sustained activity.

Naïve Input We define the external input vector as:

$$\mathbf{h}_t = \mathbf{h}_0 \cdot f_t, \tag{3.6}$$

where $f_t \in \{0, 1\}$ indicates the presence of stimulus input at time t. The static vector $\mathbf{h}_0 \in \mathbb{R}^{2M}$ encodes the presence and orientation of M items:

$$\mathbf{h}_0 = \left[a_1 \cos\theta_1,\ a_1 \sin\theta_1,\ \ldots,\ a_M \cos\theta_M,\ a_M \sin\theta_M\right]^\top, \tag{3.7}$$

where a_i denotes whether item i is present. Figure **??** illustrates this encoding for a single item.

The input matrix **B** maps the external input into the recurrent state space and is initialised using the standard Xavier scheme [14] for simplicity.

$$B_{ij} \sim \mathcal{N}\left(0, \frac{2}{N + 2M}\right). \tag{3.8}$$

Excitatory Input The enforcing of positive external input is motivated by two facts:

1. In the cerebral cortex, long-range connections that convey external or stimulus-driven inputs are predominantly excitatory [12].

2. If external input were allowed to take both positive and negative values, excitatory and inhibitory contributions could naturally cancel out, reducing the functional relevance of divisive normalization (DN)-like behavior.

To ensure that the external input term $\mathbf{B}\mathbf{h}_t$ remains strictly non-negative, we require both the input pattern $\mathbf{h}_t$ and the connection matrix **B** to be non-negative.

Specifically, for each item, we design the static input vector for $\mathbf{h}_0$ such that (a) it remains non-negative for all orientations θ, and (b) all orientations are treated symmetrically, preserving

rotational invariance. To achieve this, we embed the original 2D unit circle into a 3D space. The circle lies on a plane orthogonal to the vector $[1,1,1]^\top$, and its center is translated to $[1,1,1]^\top$, ensuring all vector elements are symmetric and positive. As visualized in Figure **??**B, the static vector for a single item is defined as:

$$\mathbf{h}_0 = a\left(\mathbf{1} + \mathbf{M}\begin{bmatrix}\cos\theta \\ \sin\theta\end{bmatrix}\right), \quad \text{where} \quad \mathbf{M} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{6}} \\ 0 & -\frac{2}{\sqrt{6}} \end{bmatrix}, \quad \mathbf{1} = \begin{bmatrix}1\\1\\1\end{bmatrix}.$$

For multiple items, these static vectors are concatenated to form the full input vector $\mathbf{h}_0 \in \mathbb{R}^{3M}$.

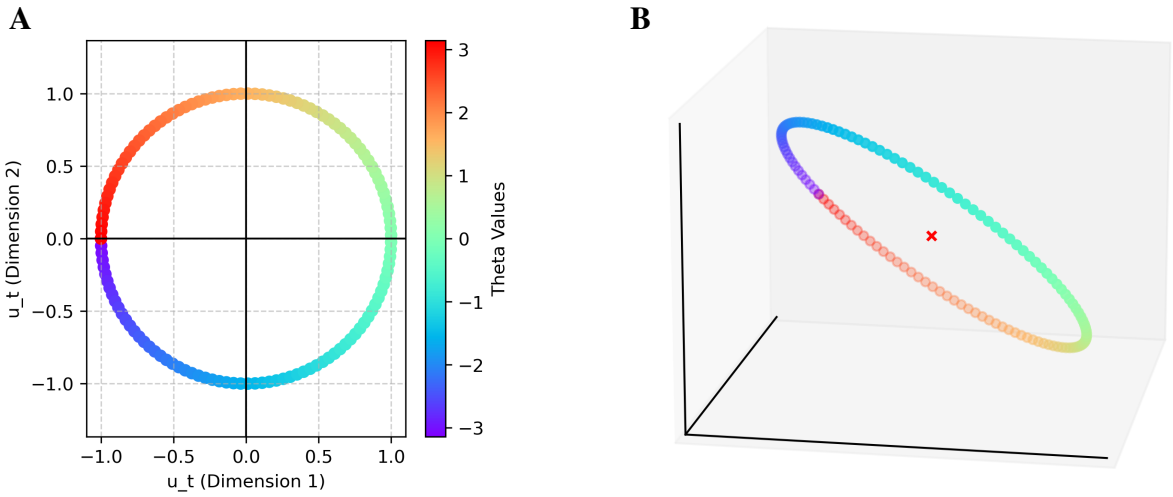


Fig. 3.2 **Orientation-encoded input vectors with and without positivity constraints. A**: Input representation using a standard 2D embedding without non-negativity. **B**: Modified 3D formulation that ensures all input vectors are non-negative and norm-conserved across angles θ. The color hue indicates stimulus orientation.

The magnitude of input $\|\mathbf{B}\mathbf{h}_t\|$ scales approximately linearly with the set size m. To ensure that even a single item ($m = 1$) produces input strong enough to activate the network, we initialize $\mathbf{B}$ from a zero-mean Gaussian and rectify it:

$$B_{ij} = |\xi_{ij}|, \quad \xi_{ij} \sim \mathcal{N}(0, \sigma^2), \tag{3.9}$$

The standard deviation σ is chosen such that the expected input from a single item equals a target input level $R = 12$ Hz. This yields the scaling (see Appendix A.2.2 for derivation):

$$\sigma = \frac{R}{3}\sqrt{\frac{\pi}{2}} \approx R \times 0.418. \tag{3.10}$$

3.2.4 Recurrent Weights and Dale's Law

Biological neurons obey *Dale's Law*, which states that each neuron releases either excitatory or inhibitory neurotransmitters, but not both [26]. In the context of recurrent neural networks, this implies that the outgoing synaptic weights of each neuron must be either all non-negative (excitatory) or all non-positive (inhibitory).

To enforce Dale's Law, we begin by initializing the recurrent weight matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$ with independent Gaussian entries:

$$\text{if neuron } j \text{ is excitatory:} \quad W_{ij} \sim \mathcal{N}\left(0, \sigma_e^2\right), \tag{3.11}$$

$$\text{if neuron } j \text{ is inhibitory:} \quad W_{ij} \sim \mathcal{N}\left(0, \sigma_i^2\right). \tag{3.12}$$

As shown in Appendix A.2.1, excitatory and inhibitory neurons can be initialized from the same zero-mean Gaussian distribution without introducing systematic imbalance, provided the variance is chosen as:

$$\sigma_e^2 = \sigma_i^2 = \frac{1}{\sqrt{N}\left(1 - \frac{1}{\pi}\right)}. \tag{3.13}$$

We then enforce Dale's Law by projecting the weights of each neuron to the appropriate sign:

$$\text{if neuron } j \text{ is excitatory:} \quad \mathbf{W}_{:,j} \leftarrow |\mathbf{W}_{:,j}|, \tag{3.14}$$

$$\text{if neuron } j \text{ is inhibitory:} \quad \mathbf{W}_{:,j} \leftarrow -|\mathbf{W}_{:,j}|. \tag{3.15}$$

This procedure ensures that each neuron emits either purely excitatory or purely inhibitory outputs. A visual comparison between unconstrained and Dale-constrained weight matrices is provided in Figure 3.3.

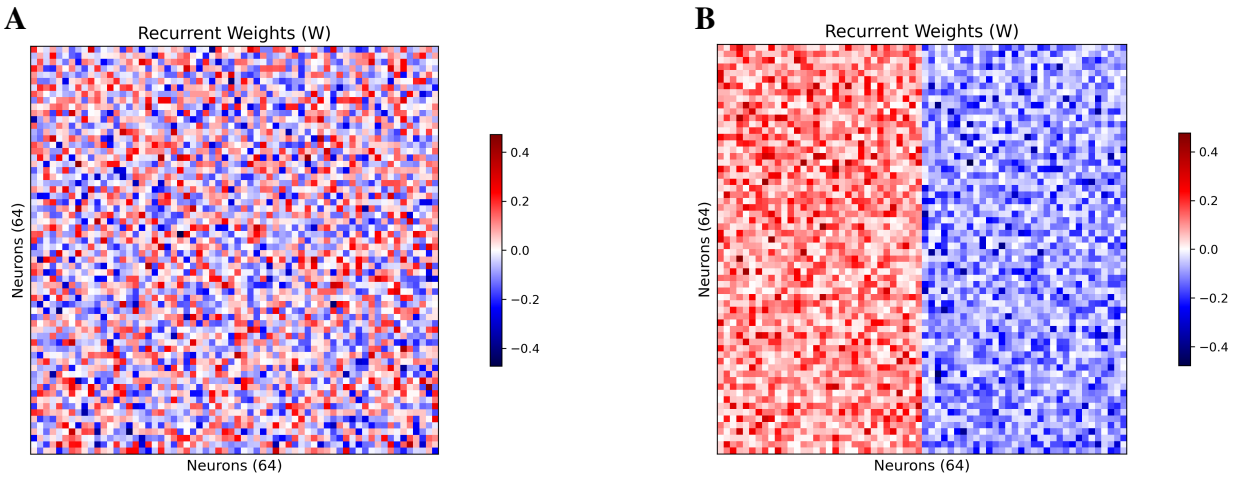


Fig. 3.3 **Effect of Dale's law on recurrent connectivity. A**: Recurrent weight matrix without sign constraints. Each neuron may emit both excitatory and inhibitory outputs. **B**: Weight matrix after imposing Dale's law. Each column is strictly positive (excitatory) or negative (inhibitory), enforcing sign-consistent output per neuron.

3.2.5 Readout and Decoding

Readout Vector. The model readout is defined as a linear projection of the noisy firing rates:

$$\hat{\mathbf{u}}_t = \mathbf{F}\tilde{\mathbf{r}}_t$$

where $\tilde{\mathbf{r}}_t \in \mathbb{R}^N$ is the noise-corrupted firing rate vector at time t, and $\mathbf{F} \in \mathbb{R}^{2M\times N}$ is the fixed readout matrix that maps neural activity back to a two-dimensional representation of orientation for each of the M items. For item i, its orientation is encoded via a 2D vector $(x_i, y_i) = (a_i \cos\theta_i, a_i \sin\theta_i)$, and the network is trained to produce an output vector $(\hat{x}_i, \hat{y}_i)$ whose angle estimates the stimulus orientation.

Decoded Orientation. To decode the estimated orientation $\hat{\theta}$, we first average the readout vector over the decoding phase:

$$\bar{\mathbf{u}} = \frac{\Delta t}{T} \sum_{\text{decode}} \hat{\mathbf{u}}_t = \mathbf{F} \frac{\Delta t}{T} \sum_{\text{decode}} \tilde{\mathbf{r}}_t \tag{3.16}$$

In the single-item case ($M = 1$), $\bar{\mathbf{u}} \in \mathbb{R}^2$ encodes the decoded orientation in its components: $(\cos\hat{\theta}, \sin\hat{\theta})$. The estimated angle is then given by:

$$\hat{\theta} = \text{Arg}(\bar{\mathbf{u}}), \tag{3.17}$$

where $\text{Arg}(\cdot)$ denotes the complex argument, i.e., the angle formed by the 2D vector $\bar{\mathbf{u}}$. This decoding rule corresponds to projecting time-averaged neural activity back into feature space and interpreting its direction as the estimated stimulus orientation.

This procedure is biologically plausible in that it mimics how downstream neurons could temporally integrate noisy synaptic input to compute an average direction of activity. It also aligns with how perceptual reports in psychophysics are derived—by taking the internal estimate that best approximates the true stimulus.

Behavioral Error. The model's behavioral output is evaluated by the angular deviation between the estimated orientation $\hat{\theta}$ and the true target orientation θ:

$$\Delta\theta = \text{wrap}(\hat{\theta} - \theta) \in (-\pi, \pi], \tag{3.18}$$

where $\text{wrap}(\cdot)$ maps the difference to the interval $(-\pi, \pi]$.

Analyzing the distribution of $\Delta\theta$ across trials and set sizes allows us to compare model behavior to experimental data from human and animal working memory tasks.

3.2.6 Heterogeneous Time Constants

Biological neurons exhibit substantial diversity in their temporal integration properties. Cortical pyramidal neurons typically display membrane time constants in the range of 10–30 ms, while fast-spiking interneurons integrate even more rapidly [10]. Our mean-field framework, however, allows the use of slower time constants to emulate the dynamics of integrative or modulatory subcircuits.

Based on the findings in Section 3.4.2, we assign each neuron a distinct time constant τ_i, independently drawn from a log-uniform distribution over the interval $[50, 300]$ ms. This range reflects a functional trade-off: excessively small τ leads to rapid memory decay and failure to maintain information over the delay period, whereas overly large τ makes the task trivial, yielding both unrealistically low error and low activation.

3.2.7 Process Noise: Poisson-like Variability

Neural activity in the brain exhibits stochastic variability, often approximated as arising from Poisson processes (see Appendix A.1.4). This variability shapes both the reliability and efficiency of information transmission. In this section, we explore several continuous approximations of Poisson-like variability suitable for differentiable rate-based RNN.

Poisson Variability Neural spiking activity exhibits intrinsic variability, often modeled as a Poisson process: the spike count within a small time window Δt is a random variable with both

mean and variance equal to $r\Delta t$, where r denotes the underlying firing rate. In our model, each unit represents the average activity of a population of Z biological neurons. Under this mean-field abstraction, the total spike count becomes $n \sim \text{Poisson}(Zr\Delta t)$, and the normalized firing rate is:

$$\tilde{r} = \frac{n}{Z\Delta t}, \tag{3.19}$$

which yields $\mathbb{E}[\tilde{r}] = r$ and $\text{Var}[\tilde{r}] = \frac{r}{Z\Delta t}$.

To control noise magnitude, we introduce a scale parameter $k \equiv 1/\sqrt{Z}$. This yields variance $\text{Var}[\tilde{r}] = \frac{rk^2}{\Delta t}$, interpolating between strong variability ($k = 1$, i.e., single-neuron Poisson noise) and deterministic rates ($k \to 0$, i.e., infinite population averaging).

The signal-to-noise ratio (SNR) of this Poisson-derived model grows with firing rate:

$$\text{SNR} = \frac{\mathbb{E}[\tilde{r}]}{\sqrt{\text{Var}[\tilde{r}]}} = \sqrt{\frac{r\Delta t}{k^2}}. \tag{3.20}$$

Thus, higher firing rates intrinsically lead to higher SNR—implying that total population activity acts as a computational resource, limited by Divisive Normalisation. This principle underlies the core idea in resource-based spiking models of working memory [1]: precision improves with greater neural activity.

While biologically grounded, the Poisson distribution is discrete and thus not differentiable–making it incompatible with gradient-based learning. We therefore approximate it using continuous, differentiable distributions (Gaussian or Gamma) that preserve the same mean and variance.

Gaussian Approximation. By the Central Limit Theorem, the normalized Poisson distribution converges to a Gaussian as $Z \to \infty$ or $k \to 0$:

$$\tilde{r} \sim \mathcal{N}\left(r,\, k^2 \frac{r}{\Delta t}\right). \tag{3.21}$$

Equivalently, we may write:

$$\tilde{r} = r + k\sqrt{\frac{r}{\Delta t}} \cdot \mathcal{N}(0,1). \tag{3.22}$$

Gamma Approximation. An alternative to Gaussian noise is the Gamma distribution, which preserves the non-negativity of firing rates. To match a target mean r and variance $k^2 r/\Delta t$, we parameterize the Gamma distribution with:

$$\text{rate } \lambda = \frac{\Delta t}{k^2}, \qquad \text{shape } \alpha = \frac{r\Delta t}{k^2}, \tag{3.23}$$

so that:

$$\tilde{r} \sim \text{Gamma}(\alpha,\, \lambda). \tag{3.24}$$

For differentiable sampling, we use the reparameterized `rsample()` method in PyTorch, which expresses the sample as a differentiable transformation of a fixed-base distribution.

Advantage of Gamma Approximation Both Gaussian and Gamma distributions can approximate Poisson-like variability, but Gamma strictly preserves non-negativity. As shown in Figure 3.4

compares the two approximations for a ground-truth rate of $r = 15$ Hz across different noise scales k. When variability is low ($k = 0.05$), both approximations behave similarly around the true rate $r = 15$ Hz. However, at higher noise levels ($k = 0.3$), the Gaussian distribution yields many negative values that must be clamped to zero, introducing bias. In contrast, the Gamma distribution remains positive and well-behaved, making it more suitable for biologically realistic modeling under high variability.

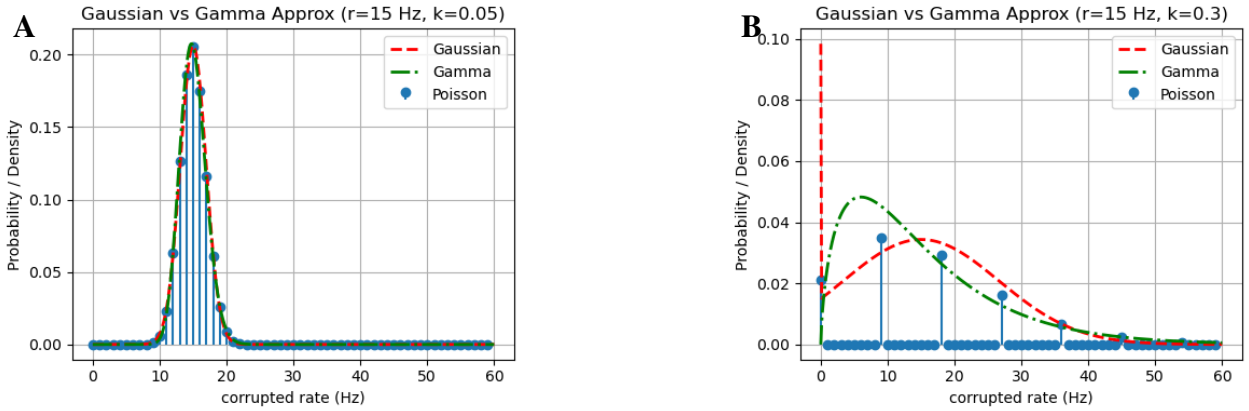


Fig. 3.4 **Gaussian vs. Gamma approximations to Poisson-like variability.** **A**: At low variability ($k = 0.05$), both distributions approximate the Poisson distribution well. **B**: At higher variability ($k = 0.3$), the Gaussian produces many negative samples (leads to the peak at zero), whereas the Gamma remains non-negative and symmetric around the mean. These results underscore the Gamma distribution's suitability for biologically realistic noise modeling.

Implementation in RNN Dynamics. The noise-corrupted firing rate $\tilde{\mathbf{r}}_t$ replaces the latent rate $\mathbf{r}_t$ wherever neural activity is observed. It is used in both the recurrent input $\mathbf{W}\tilde{\mathbf{r}}_t$ and the readout $\mathbf{F}\tilde{\mathbf{r}}_t$, allowing variability to propagate through dynamics and decoding.

3.3 Loss Functions and Training

The total loss function used to train the network is composed of two components:

$$\mathscr{L} = \mathscr{L}_{\text{activ}} + \mathscr{L}_{\text{error}}, \tag{3.25}$$

where $\mathscr{L}_{\text{activ}}$ penalizes high average firing rates, and $\mathscr{L}_{\text{error}}$ quantifies task-related performance. Several different forms of $\mathscr{L}_{\text{error}}$ were designed, motivated by geometric, angular, and psychophysical considerations.

3.3.1 Metabolic Cost

To constrain the overall activity level of the network, we include an ℓ_1-based activation regularization term:

$$\mathscr{L}_{\text{activ}} = \left\langle \frac{\Delta t}{T_{\text{sim}}} \sum_{t=0}^{T_{\text{sim}}} \frac{\lambda}{N} \|\mathbf{r}_t\|_1 \right\rangle, \tag{3.26}$$

where $\|\mathbf{r}_t\|_1/N$ denotes the average firing rate (Hz) across the population at time t, and $\lambda = 0.00001$ is used in this project. The outer average $\langle\cdot\rangle$ is taken over trials. This term serves to enforce metabolic constraints by suppressing global activity and encouraging sparsity[29, 19].

3.3.2 Variants of Error

Euclidean Error

A natural geometric loss is the squared Euclidean distance between the target vector $\mathbf{u}_0 \in \mathbb{R}^{2m}$ and the decoded output vector $\bar{\mathbf{u}} \in \mathbb{R}^{2m}$, averaged over items and trials:

$$\mathscr{L}_{\text{error}} = \left\langle \frac{1}{m} \|\mathbf{u}_0 - \bar{\mathbf{u}}\|_2^2 \right\rangle, \tag{3.27}$$

where m is the number of presented items on each trial.

Angular Error

When orientations are represented on the unit circle, it is natural to measure error in angular space. This also represents the linear similarity in [25]. In the single-item case ($m = 1$), where both the target $\mathbf{u}_0 = [\cos\theta, \sin\theta]^\top$ and the decoded estimate $\bar{\mathbf{u}} = [\cos\hat{\theta}, \sin\hat{\theta}]^\top$ are unit vectors, the unsigned angular error can be computed using the dot product:

$$\cos|\Delta\theta| = \mathbf{u}_0^\top \bar{\mathbf{u}}, \tag{3.28}$$

$$|\Delta\theta| = \arccos(\mathbf{u}_0^\top \bar{\mathbf{u}}). \tag{3.29}$$

This generalizes to multi-item trials as:

$$\mathscr{L}_{\text{error}} = \left\langle \frac{1}{m} \sum_{i=1}^{m} |\Delta\theta_i| \right\rangle = \left\langle \frac{1}{m} \sum_{i=1}^{m} \arccos\left(\mathbf{u}_{0,i}^\top \bar{\mathbf{u}}_i\right) \right\rangle, \tag{3.30}$$

where $\mathbf{u}_{0,i}$ and $\bar{\mathbf{u}}_i \in \mathbb{R}^2$ are the target and decoded vectors for item i.

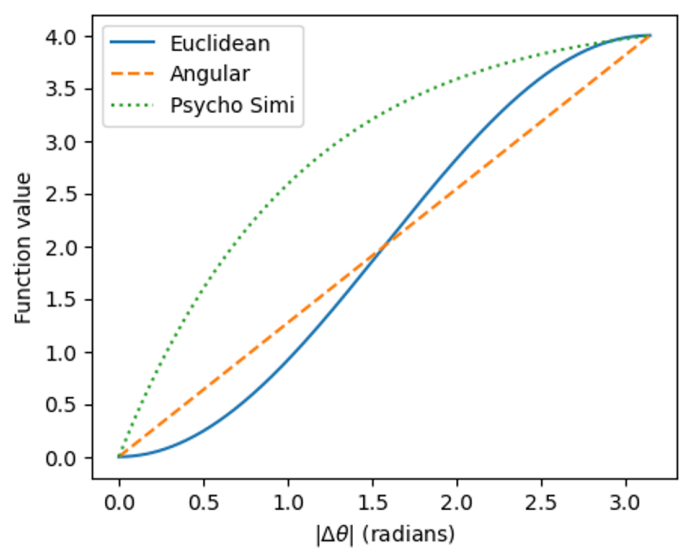


Fig. 3.5 **Comparison of error metrics used for training.** The plot shows the functional form of three loss functions as a function of unsigned angular error $|\Delta\theta| \in [0, \pi]$, all scaled to a comparable range. The Euclidean loss (blue solid) grows quadratically, penalizing small errors more softly and large errors more strongly. The angular loss (orange dashed) increases linearly with error magnitude, offering a balanced gradient across the range. The psychophysical similarity loss (green dotted), derived from perceptual similarity curves [25], exhibits exponential decay of similarity with error, saturating quickly—leading to flatter penalties for large errors. This perceptually inspired shape promotes robustness to large deviations, but can introduce vanishing gradients during training.

Psychophysical Similarity Error

Inspired by empirical findings in human perception, we also consider a loss function based on psychophysical similarity. As discussed in Section **??**, Schurgin et al. [25] demonstrated that perceived similarity between two orientations declines exponentially with their angular separation. Specifically, the similarity function is modeled as:

$$S(|\Delta\theta|) = \exp\left(-\frac{\lambda}{\pi}|\Delta\theta|\right), \tag{3.31}$$

where $\lambda = 5$ is a fitted parameter derived from behavioral data.

To construct a corresponding loss function, we take the dissimilarity $1 - S$:

$$\mathscr{L}_{\text{error}} = 1 - \left\langle \frac{1}{m}\sum_{i=1}^{m} S(|\Delta\theta_i|) \right\rangle. \tag{3.32}$$

This loss penalizes large errors more gently than standard geometric or angular losses, reflecting the graded perceptual indistinguishability of nearby orientations. It is expected to encourage behaviorally realistic error patterns with heavy-tailed distributions.

3.3.3 Empirical Comparison of Error Metrics

To better understand how different objective functions shape learning dynamics, we compare three error metrics—Euclidean, angular, and psychophysical similarity—in terms of how they penalize angular deviations $|\Delta\theta|$. As shown in Figure 3.5, the Euclidean loss increases quadratically, assigning stronger penalties to large deviations. The angular error grows linearly and thus yields a constant gradient across the range. In contrast, the psychophysical similarity loss, inspired by perceptual scaling laws [25], rapidly saturates and flattens out for large errors, which can lead to vanishing gradients and unstable training early on.

Among these three, the **Euclidean loss** proved to be the most stable and effective for training. It consistently produced robust results across random seeds and hyperparameter configurations. In contrast, the **angular error**—though theoretically well-justified—exhibited high variance in training outcomes: under identical settings, models either converged to low-error solutions or failed to learn the task entirely. This instability likely stems from the more complex gradients. The **psychophysical similarity loss**, while behaviorally grounded and perceptually meaningful, introduced severe optimization challenges. Its gradients vanish for large angular errors, making early training highly inefficient and often leading the model to converge to a trivial, non-informative solution that suppresses activations.

For this reason, we adopt the Euclidean error for all subsequent hyperparameter analyses.

3.3.4 RNN Training

All models were trained using PyTorch on a server equipped with two NVIDIA 2080 Ti GPUs and an Intel(R) Core(TM) i9-10900X CPU @ 3.70 GHz, hosted at the Computational and Biological Learning (CBL) Lab.

To prevent overfitting, we adopted an early stopping criterion based on validation performance, implemented using the open-source PyTorch utility provided by Sunde [27]. Specifically, training was halted if no improvement in validation loss was observed for 600 consecutive epochs.

An adaptive learning rate schedule was employed using PyTorch's `ReduceLROnPlateau`, which dynamically reduces the learning rate when the validation loss plateaus. In our experiments, the

learning rate was initialized at 0.001, and the scheduler was configured with a patience of 400 epochs.

3.4 Effects of Hyperparameters on Model Behavior

In this section, we investigate how error strength, network size, and time constant τ influence model behavior. To ensure robustness of training, we use a loss function combining the Euclidean decoding error with a L1 metabolic cost.

3.4.1 Impact of Network Size and Noise Level

We evaluated how network size and input noise jointly influence performance by comparing mean angular error across set sizes. Figure 3.6 shows results for networks with 64, 128, and 256 neurons, each tested under varying levels of Poisson-like input noise.

The error–set size curves reveal two distinct regimes of network behavior. In the first regime, observed under moderate noise and sufficient network size, the mean error increases monotonically with set size. This pattern aligns with empirical findings in human behavioral experiments, where remembering more items leads to reduced precision per item.

In the second regime, the error remains flat across all set sizes at approximately $\pi/2$ radians—the expected value for a uniform distribution on the circle. This indicates that the network is essentially guessing, failing to encode or retain any meaningful item-specific information. Such behavior typically arises when the network is too small to support robust representations or when input noise is too strong for the system to extract usable information.

Within each subplot, increasing the noise level reliably led to higher error, confirming that stronger input variability impairs memory precision. Across subplots, larger networks consistently achieved lower errors under the same noise conditions, suggesting that increased network size improves memory capacity and robustness to noise.

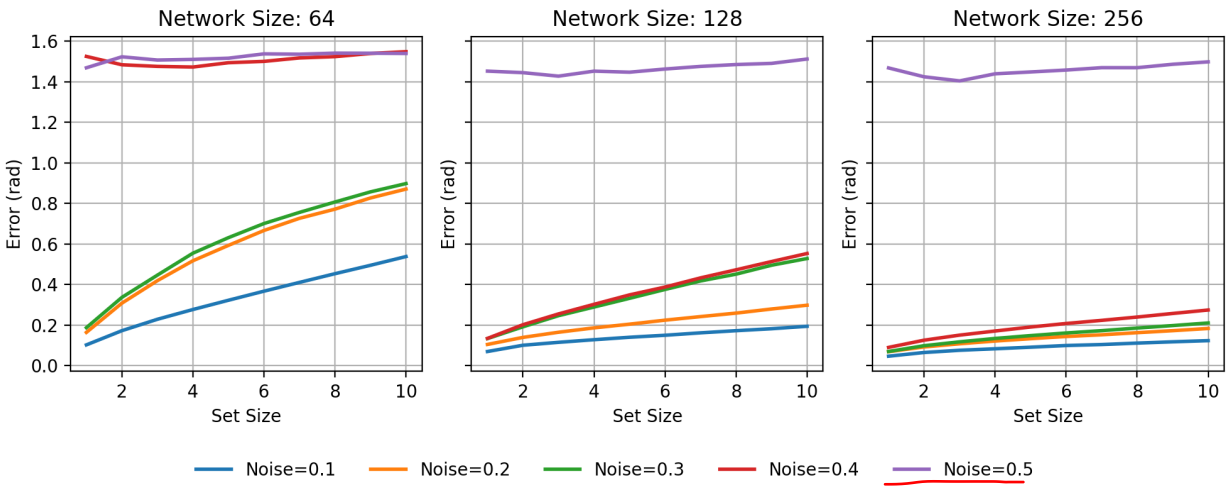


Fig. 3.6 **Mean angular error (rad) as a function of set size**, across different network sizes and input noise levels. Each subplot corresponds to a network size (64, 128, 256). Within each subplot, curves show increasing Poisson noise from bottom to top. Larger networks and lower noise produce lower error.

3.4.2 Effect of Time Constant τ

The membrane time constant τ determines the temporal integration window of individual neurons. To examine its influence on working memory performance and metabolic cost, we trained networks with heterogeneous τ drawn from three ranges. The results are shown in Figure 3.7.

- **Short** (30–100 ms): The network fails to retain information over the delay period. Both performance and mean activation collapse, indicating the network has given up to save metabolic cost.
- **Medium** (50–300 ms): This regime yields the best trade-off. Error increases with set size, resembling human-like degradation, while activation remains moderate. We adopt this range in the rest of the project.
- **Long** (100–1000 ms): Information is accurately preserved. Both error and activation remain low across set sizes, suggesting the task has become too simple to be informative.

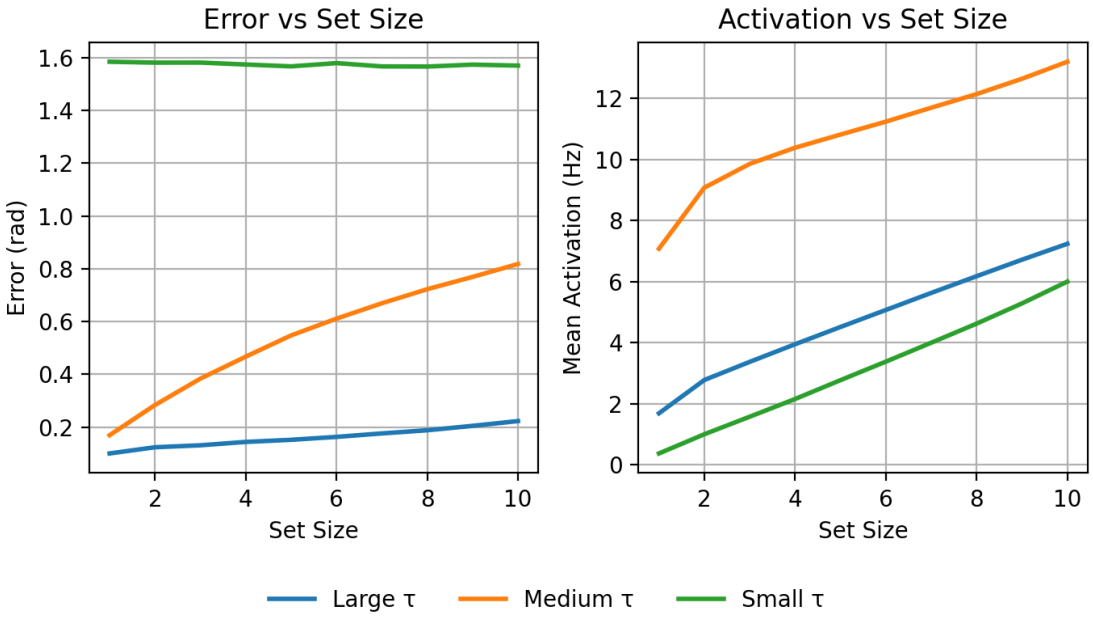


Fig. 3.7 **Effect of heterogeneous time constant τ on task performance and network activation. Left:** Mean error as a function of set size. **Right:** Mean firing rate. Short τ fails to sustain memory; long τ trivializes the task.

4 Emergent Neural Computations

In this chapter, we identify two model configurations whose behavioral outputs closely match empirical human data. We then analyze the emergent neural computations underlying these models.

4.1 Matching Model Behavior to Human Error Patterns

To compare our model outputs with human behavioral data from Bays [1], we examined the distribution of recall errors across different set sizes. Specifically, we computed two circular statistics[13] for each set size:

- **Circular variance** captures the dispersion of angular errors around the target:

$$\sigma^2 = -2\log|\bar{m}_1|, \quad \bar{m}_1 = \frac{1}{N}\sum_{j=1}^{N} e^{i\Delta\theta_j} \tag{4.1}$$

- **Circular kurtosis** quantifies the heaviness of the tails of the error distribution. Lower values of κ (closer to 0) indicate distributions that are more Gaussian-like:

$$\kappa = \frac{|\bar{m}_2|\cos(\arg\bar{m}_2 - 2\arg\bar{m}_1) - |\bar{m}_1|^4}{(1-|\bar{m}_1|)^2}, \quad \bar{m}_2 = \frac{1}{N}\sum_{j=1}^{N} e^{2i\Delta\theta_j} \tag{4.2}$$

4.1.1 Model Selection Based on Behavior Matching

Among all model configurations explored, two stood out for their close match to human recall error patterns:

1. **Model with Euclidean loss**: input strength $R = 12$, network size $N = 64$ neurons, Poisson noise factor $k = 0.2$, trained using Euclidean error.
2. **Model with angular loss**: $R = 12$, $N = 256$ neurons, $k = 0.3$, trained using angular error.

Both models reproduced the heavy-tailed structure of the empirical error distribution (see Figure 4.1). Moreover, their circular variance and kurtosis trends approximately track the empirical data across set sizes, further supporting their behavioral plausibility.

4.2 Preliminary Characterization of Neural Responses

To build intuition for the dynamical regime underlying each trained network, we conducted a series of checks on basic neural activity patterns. These descriptive statistics offer a coarse, yet informative, lens into how information is represented and maintained across task phases.

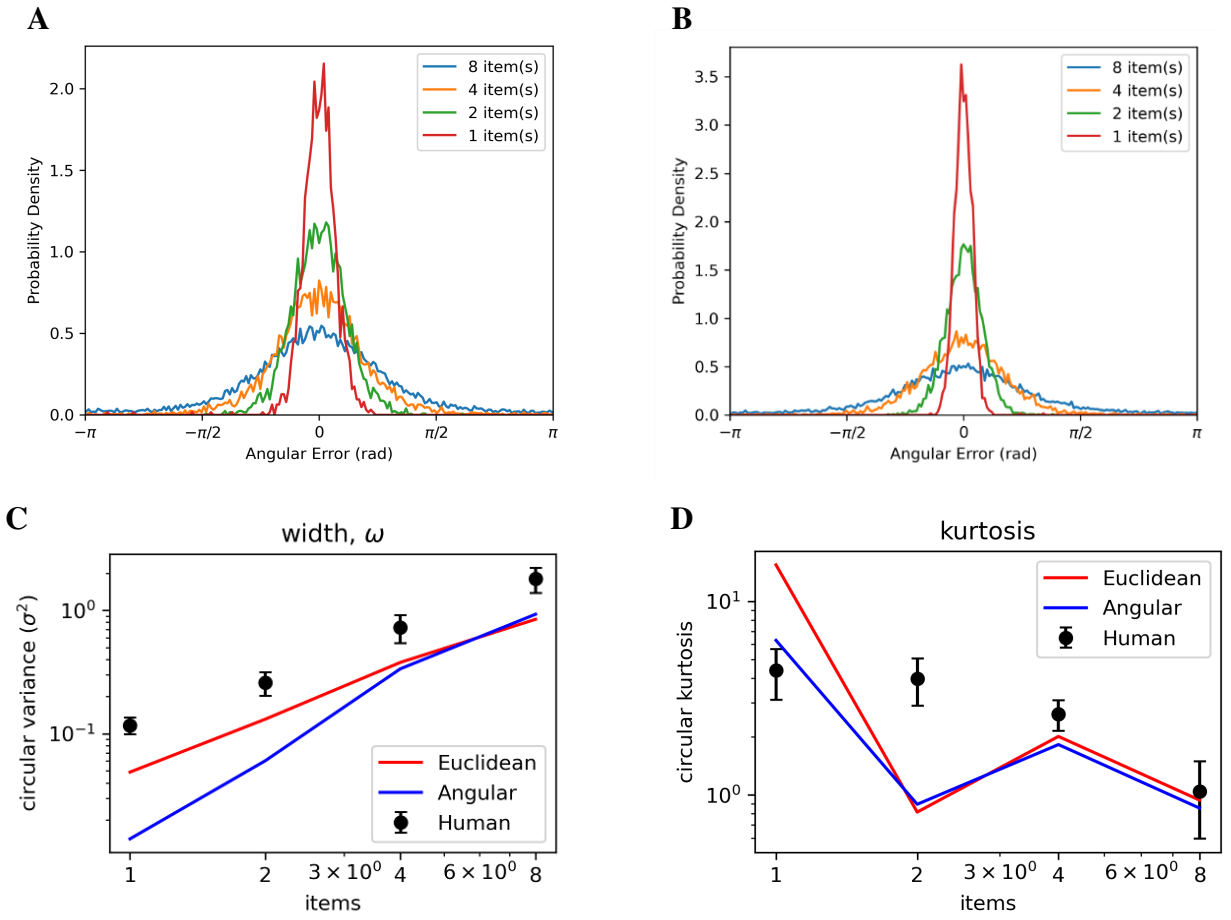


Fig. 4.1 **Comparison of model and human error statistics.** **A**: Error distribution of the model with Euclidean loss($R = 12, N = 64, k = 0.2$). **B**: Error distribution of the model with angular loss ($R = 12, N = 256, k = 0.3$). **C**: Circular variance as a function of set size, overlaid with human data from [1]. **D**: Circular kurtosis trends across set size conditions. Both models reproduce the heavy-tailed errors observed empirically.

We analyzed 1000 trials with set size 5 for both models. For each neuron, we measured:

- **External input**: Compute the trial-wise standard deviation. Its variability scales linearly with the mean (Fig. 4.2A, D), consistent with static, non-noisy inputs determined by random stimulus orientations.
- **Recurrent input**: Calculate the mean and standard deviation over the delay phase and trials. Recurrent inputs span both excitatory and inhibitory regimes (Fig. 4.2B, E), with the model with angular loss exhibiting larger magnitudes, reflecting balanced dynamics for sustained activity.
- **Firing rate**: Measure the mean and standard deviation during the delay phase. Variability scales differently across models (Fig. 4.2C, F): sublinear with the mean in the model with angular loss, but approximately linear in the model with Euclidean loss.

4.3 Dynamical Landscape of the Trained Networks

4.3.1 Model with Euclidean Loss

To characterise how the model with Euclidean loss stores information when only one item is present (set size = 1), we analyse its activity at three complementary levels:

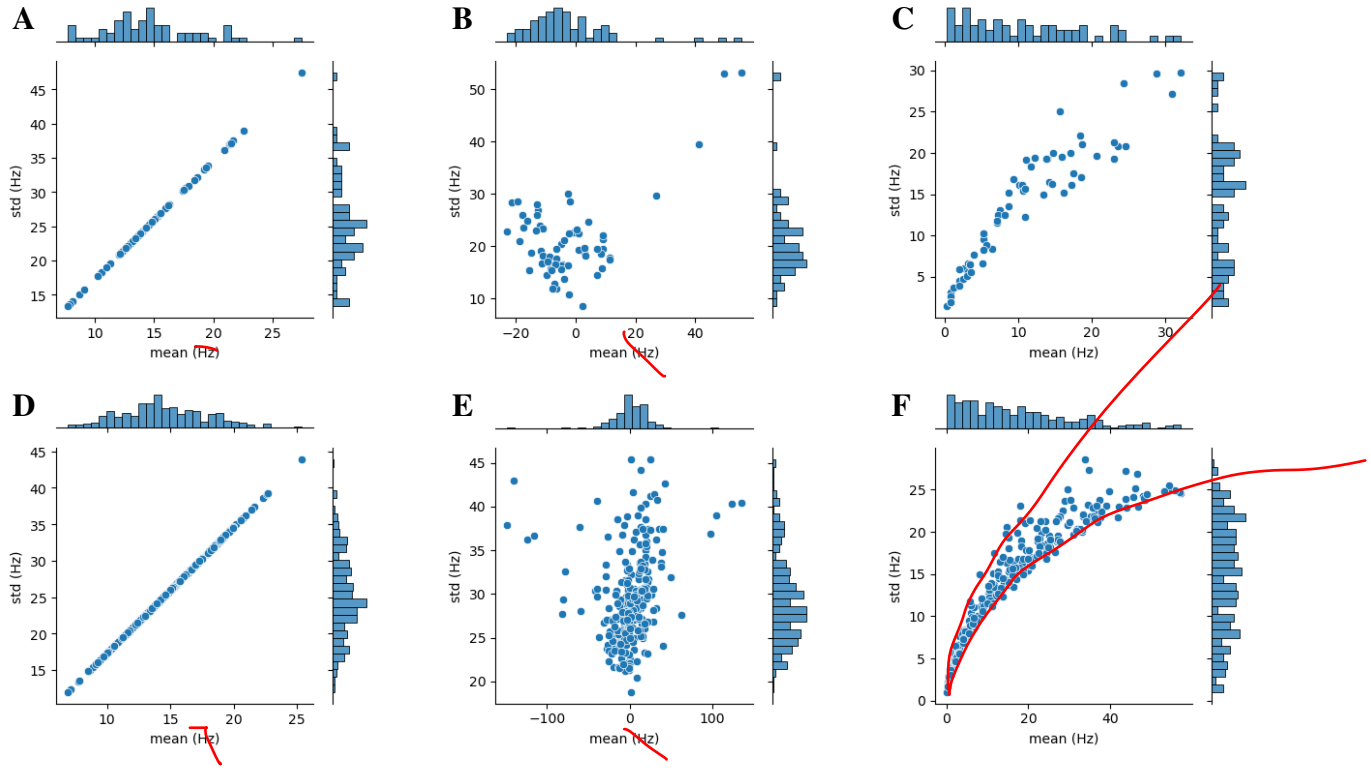


Fig. 4.2 **Neuron-wise statistics across trials (set size = 5)**. **A–C**: model with Euclidean loss. **D–F**: model with angular loss. Each scatter point corresponds to a single neuron, summarizing its mean and standard deviation across trials and time. External input is static and thus showing linear mean-variance relationship; recurrent input spans both excitation and inhibition; firing activity saturates sublinearly around 30 Hz for model with angular loss.

(i) **Single-neuron firing**: illustrates whether individual units sustain, ramp, or decay.
(ii) **Decoding-plane trajectory**: projects the network state $\mathbf{r}_t$ onto the 2-D readout space spanned by **F**.
(iii) **Full-state trajectory in PCA space**: visualises the high-dimensional dynamics, with and without fixed points identified by *FixedPointFinder* (FPF) [28].

The Euclidean network operates via a brief input-driven excursion followed by exponential decay. During the stimulus epoch, neuronal activity ramps up and successfully encodes orientation, but this signal dissipates rapidly during the delay (Fig. 4.3A). FixedPointFinder detects only a single stable fixed point (Fig. 4.3D), revealing a monostable landscape in which no stimulus-specific attractors emerge to support persistent mnemonic activity. Thus, although the network can retain information across the delay at set size $= 1$, the underlying neural mechanism remains elusive. One possibility is that multiple attractors do exist but are so tightly clustered that they fall below the resolution of our current analysis.

4.3.2 Model with Angular Loss

Using the same analysis pipeline, we now examine the model with angular loss at set size $= 1$. The three complementary views are identical to those used for the Euclidean network:

- **Persistent coding (Panel A).** Five randomly selected neurons maintain—or even amplify—their firing throughout the delay and decode periods, in stark contrast to the transient decay observed in the model with Euclidean loss.

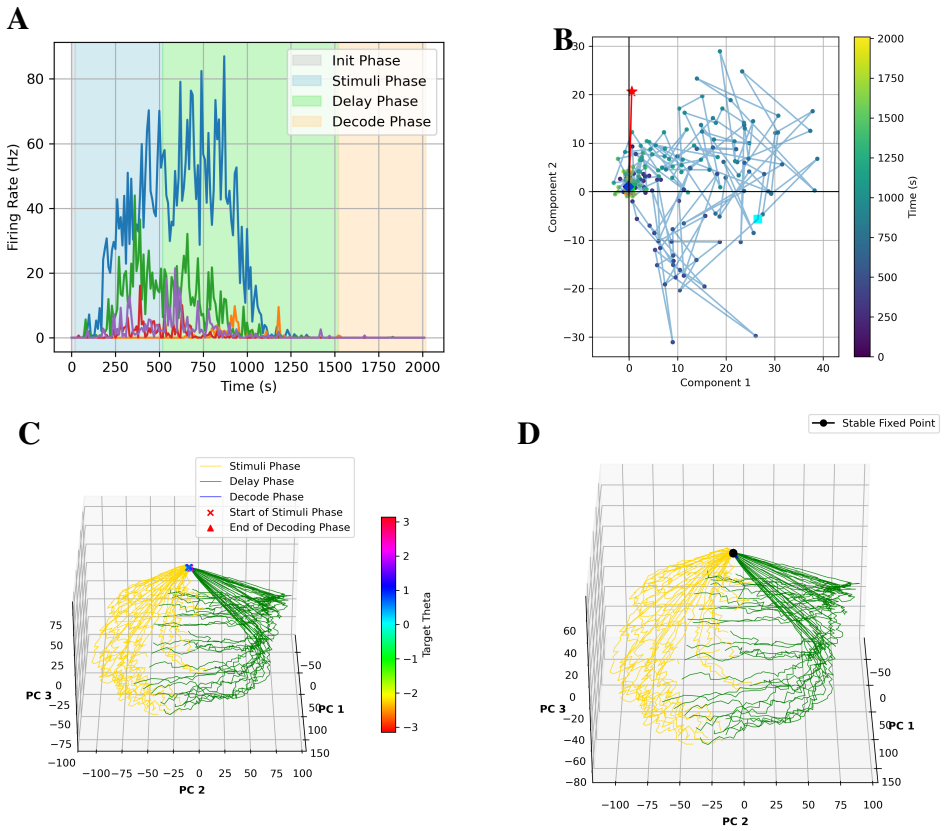


Fig. 4.3 **Dynamical signatures of the model with Euclidean loss** ($M = 1$)**. A**: the firing rates of five randomly chosen neurons over the entire task period. They ramp during stimulus, then rapidly decay. **B**: the time-coloured trajectory $\mathbf{Fr}_t$; Due to euclidean error, its magnitude converges to the unit circle at the decoding period. The state diverges during stimulus but converges toward the origin during delay and decode because euclidean error forces the magnitude to the unit circle. **C and D**: the full trajectory in the first three principal components: C omits fixed points, whereas D overlays stable (black dots) points located by FPF. 3-D PCA trajectory reveals a spiral-return pattern with no visible attractor ring. FPF detects a single global stable fixed point (black), to which all trajectories converge.

- **Continuous attractor in decoding space (Panel B).** The decoded state $\mathbf{Fr}_t$ departs the origin during stimulus presentation and then remains locked to the red guide (target orientation), demonstrating stable preservation of both amplitude and orientation.
- **Ring-manifold in high-dimensional state space (Panel C).** When projected onto the first three principal components, trajectories fan out into a broad sheet-like surface; colour-coding by stimulus orientation shows that different θ values occupy distinct sectors rather than collapsing to a single point, consistent with a continuous set of marginally stable fixed points that tile the ring attractor.

Together, these observations indicate that the model with angular loss realises a ring-attractor–like mechanism, enabling robust maintenance of the stimulus despite metabolic constraints—behaviour absent from the Euclidean architecture.

4.4 Divisive Normalisation

Divisive Normalisation (DN) is a canonical computation observed across sensory cortical circuits, wherein neuronal responses saturate with increasing input intensity [7]. In our network, DN-like

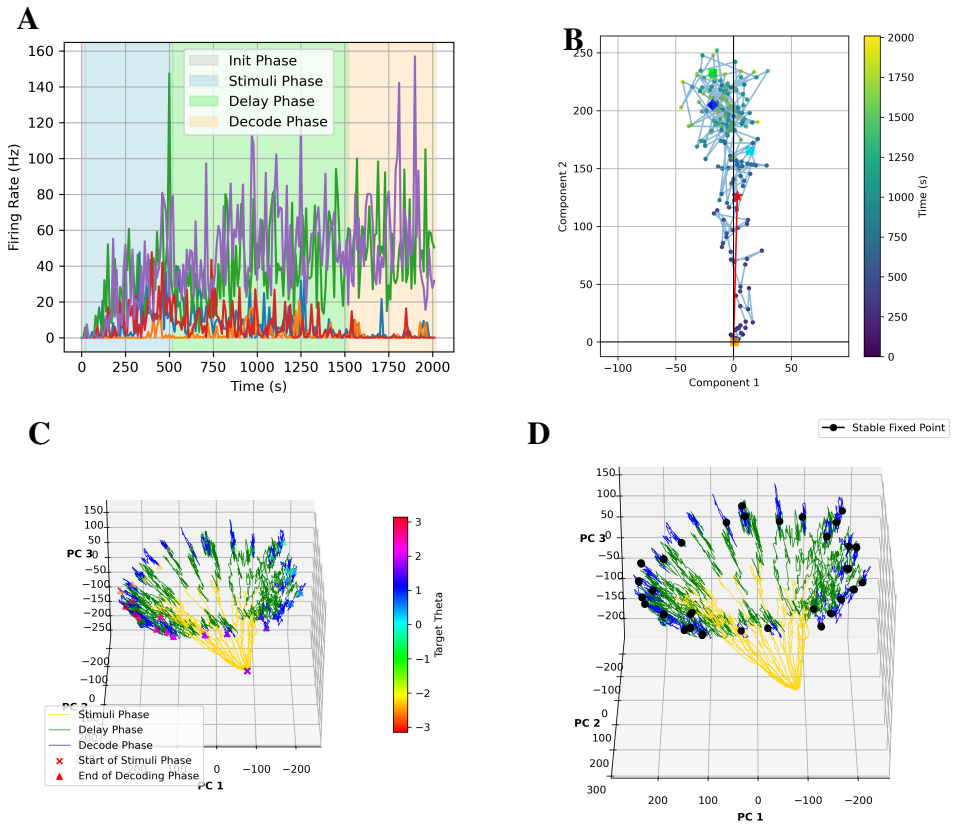


Fig. 4.4 **Dynamical signatures of the model with angular loss ($M = 1$). A**: Example neurons maintain elevated firing throughout delay and decode, providing a persistent mnemonic signal. **B**: In the decoding space the state travels outward during stimulus, then stabilises on a near-circular orbit, preserving both angle and magnitude. **C and D**: The 3-D PCA trajectory fans out into a continuous manifold whose sector encodes stimulus orientation; no global collapse to a single point is observed, consistent with a ring-like continuous attractor.

behavior emerges spontaneously, despite the absence of any architectural constraint explicitly enforcing it.

To quantify this, we measured (1) the mean input received by the network during the stimulus period, and (2) the average population activity across the entire simulation. As shown in Fig. 4.5, external input increases proportionally with the number of stimuli (left panel), while the overall activation increases much more slowly (right panel). Together, these observations are consistent with a divisive normalization–like mechanism that dynamically regulates network excitability.

We further tested whether individual neurons exhibit divisive normalization in their responses to paired items. Specifically, we compare the response to a pair of items with the linear sum of single-item responses. Let:

- $\mathbf{r}_1(\theta_1, t) \in \mathbb{R}^N$: firing-rate vector at time t when item 1 (orientation θ_1) is shown alone.
- $\mathbf{r}_2(\theta_2, t) \in \mathbb{R}^N$: firing-rate vector at time t when item 2 is shown alone.
- $\mathbf{r}_{12}(\theta_1, \theta_2, t) \in \mathbb{R}^N$: response when both items are presented together (set size 2).

We compute the mean firing rate over time $\bar{\mathbf{r}}_1(\theta_1)$, $\bar{\mathbf{r}}_2(\theta_2)$, $\bar{\mathbf{r}}_1 2(\theta_1, \theta_2) \in \mathbb{R}^N$, then fit a linear gain model:

$$\bar{\mathbf{r}}_{12} \approx W_1 \bar{\mathbf{r}}_1 + W_2 \bar{\mathbf{r}}_2, \tag{4.3}$$

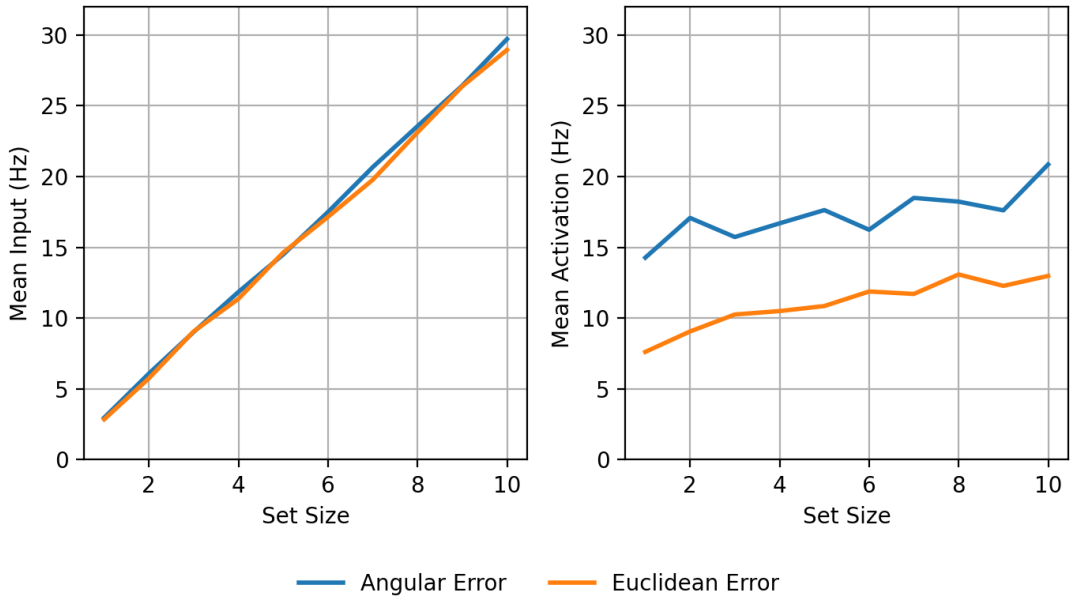


Fig. 4.5 **Mean input and population activation across set sizes.** The left panel shows that total external input increases linearly with set size for both models. The right panel reveals a sublinear relationship between set size and average activation, especially pronounced in the model with angular loss—indicating divisive normalization–like behavior.

with least-squares estimate:

$$\mathbf{W} = \begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \bar{\mathbf{r}}_{12}, \quad \mathbf{X} = [\bar{\mathbf{r}}_1 \;\; \bar{\mathbf{r}}_2]\,. \tag{4.4}$$

Here, $W_i < 1$ indicates gain suppression for input i—the hallmark of DN. We repeated this fitting across multiple (θ_1, θ_2) pairs. Fig. 4.6 shows the resulting scatter plots of (W_1, W_2).

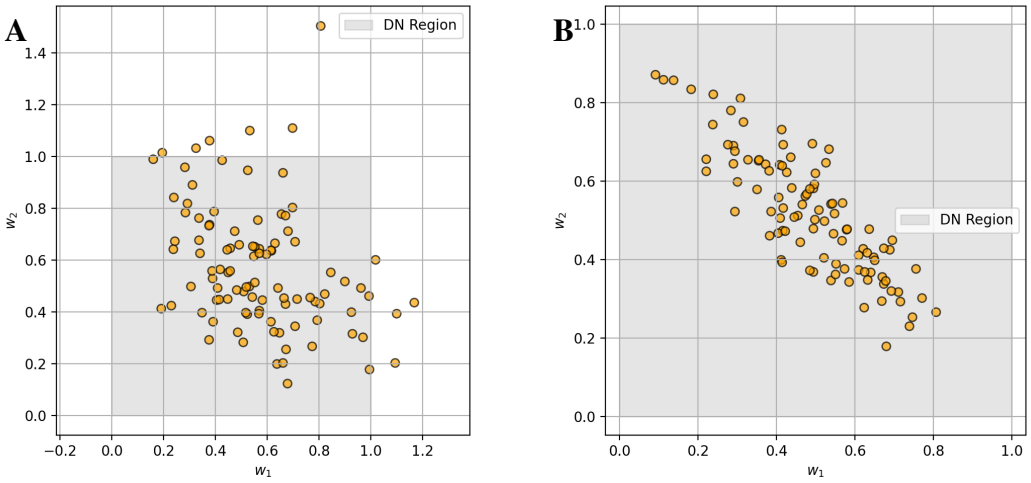


Fig. 4.6 **Sublinear population gain during paired-stimulus conditions.** Each point shows estimated weights (W_1, W_2) from the linear-gain model (4.3) for a random pair of orientations. **A**: model with Euclidean loss. Points distribute more broadly, with more instances above the DN boundary. **B**: model with angular loss. Points are strictly concentrated within the sublinear region $W_1 < 1, W_2 < 1$, consistent with divisive normalization.

In the model with Euclidean loss (panel A), weights are widely scattered, with a few points falling outside the shaded "DN region" $W_1 < 1$, $W_2 < 1$. This suggests that the network often exhibits sublinear but sometimes exhibits suprelinear integration of inputs for small set sizes. In

contrast, the model with angular loss (panel B) shows a markedly different pattern. Weights fall strictly within the sublinear region, indicating robust gain suppression across both inputs and is consistent with divisive normalization.

These results show that divisive normalization not only regulates global activation levels (as in Fig. 4.5), but also manifests at the level of individual neurons' response integration, particularly in the model with angular loss.

4.5 Mixed Selectivity

Mixed selectivity refers to the phenomenon where a single neuron exhibits conjunctive tuning to multiple task-relevant variables—for example, responding to both item identity and stimulus orientation. In contrast to pure selectivity, where a neuron is narrowly tuned to one feature of one item, mixed selectivity enables richer, more flexible population codes and is a hallmark of high-dimensional neural representations in prefrontal cortex [22] (see Appendix A.1.3.

To assess mixed selectivity in our network, we ask whether neurons encode the orientation of multiple items simultaneously, or instead specialize in responding to one item/certain orientation only. This analysis is restricted to set size 1 in this project.

4.5.1 Stimulus Selectivity via Vector Strength

For each neuron i and each item k, we computed the orientation tuning using the circular *vector strength*:

$$R_{i,k} = \frac{\left|\sum_{\theta_k} \bar{r}_i(\theta_k) e^{j\theta_k}\right|}{\sum_{\theta_k} \bar{r}_i(\theta_k)}, \tag{4.5}$$

where $\bar{r}_i(\theta_k)$ is the mean firing rate of neuron i to orientation θ_k of item k, averaged over both stimulus and decoding periods. $R_{i,k} \in [0,1]$ reflects how sharply neuron i is tuned to item k's orientation.

4.5.2 Quantifying Mixed Selectivity with IPR

To evaluate how this tuning varies across items for each neuron, we define the vector $\mathbf{R}_i = [R_{i,1}, \ldots, R_{i,M}]$ and compute its inverse participation ratio (IPR), which captures how evenly orientation selectivity is distributed across items:

$$\mathrm{IPR}(\mathbf{R}_i) = \frac{(\sum_{k=1}^{M} R_{i,k})^2}{M \sum_{k=1}^{M} R_{i,k}^2}, \tag{4.6}$$

where M is the number of items.

Interpretation.

- A neuron with low IPR (near $1/M$) is selective for only one item's orientation—indicating strong item-specificity and thus mixed selectivity across item × feature dimensions.
- A neuron with high IPR (near 1) has uniform orientation selectivity across items—exhibiting generalization rather than item-specific tuning.

4.5.3 Results and Comparison with Prior Work

Both our trained model shows that nearly all neurons exhibit strong mixed selectivity: their orientation tuning varies across items in a conjunctive, high-dimensional manner (see Fig. **??**). This stands in contrast to Bouchacourt's model, where neurons within ring attractors display pure selectivity to fixed items. Although the random network in their architecture exhibits mixed responses, it plays no functional role in storing or decoding memory content.

This result supports the notion that mixed selectivity can emerge naturally from a unified, task-optimized network, without requiring hard-coded modularity or separate subcircuits for each item.

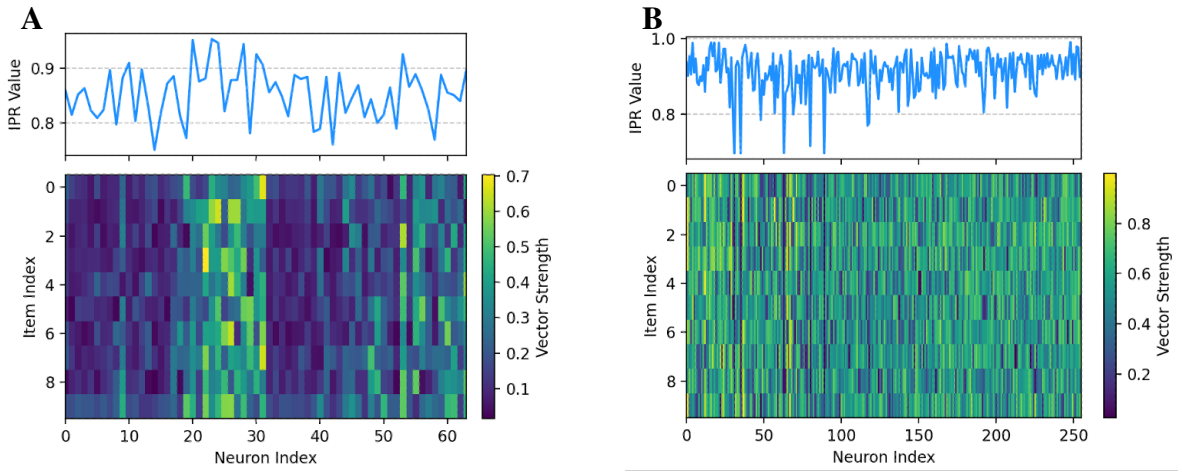


Fig. 4.7 **Mixed selectivity across neurons in two models.** Each panel contains two parts: the **bottom** heatmap shows the vector strength matrix $R_{i,k} \in \mathbb{R}^{\text{neurons} \times \text{items}}$, where each entry represents how strongly neuron i is tuned to the orientation of item k; the **top** plot shows the corresponding Inverse Participation Ratio (IPR) for each neuron, summarizing how broadly that neuron responds across items. Both model with Euclidean (**A**) and angular (**B**) losses with angular loss show high IPR values, indicating that most neurons exhibit orientation tuning for multiple items rather than a single one. This reflects a high degree of mixed selectivity across the network population in both architectures.

5 Conclusions and Outlooks

5.1 Conclusions

This report shows that a biologically constrained recurrent neural network (RNN) can reproduce core behavioural signatures of visual working memory (VWM). Specifically,

- **End-to-end construction of a biologically grounded RNN.** We built the network from scratch, revisiting virtually every design choice—activation non-linearity, excitatory-only external input, Dale's law, a Poisson–Gaussian noise approximation, and multiple loss variants—and succeeded in stable task optimisation.
- **Heavy-tailed errors without ad-hoc guessing terms.** The combination of Poisson-like variability and emergent divisive normalisation (DN) was sufficient to generate the empirically observed heavy tails in recall-error distributions, matching both variance and kurtosis across set sizes.
- **Emergent computations.** DN, mixed selectivity, and—under the angular-loss regime—a ring-attractor manifold all arose intrinsically from task optimisation; none were hard-wired into the architecture.

Together, these results bridge descriptive resource theories and mechanistic circuit models, demonstrating that realistic neural constraints can give rise to human-like memory behaviour.

5.2 Outlooks

Non-linear similarity and representational geometry. Preliminary attempts to train the network with a similarity-based loss proved unstable. Future work will (i) analyse the learned representational geometry of the current model to test for implicit non-linear distance metrics and (ii) develop training schemes that stabilise similarity-aware objectives, aiming to unify variable-precision and psychophysical-similarity accounts within a single mechanistic framework.

Multi-item interaction. Mechanistic analyses so far have centred on single-item or pairwise conditions. A full characterisation of how multiple items compete or cooperate—via recurrent inhibition, shared resource limits, or attractor interference—remains open. Resolving this issue is essential for explaining the sharp precision decline beyond approximately four items.

Origin of divisive normalisation. An open question is what circuit mechanism drives the observed DN: saturation of the activation function, reciprocal E/I balance, or something else. Ablation studies that remove the activation penalty or unbound the activation function will clarify whether DN persists without explicit activation regularisation.

Long-term vision. A unified model that (i) captures heavy tails through DN + Poisson noise, (ii) expresses non-linear representational distances consistent with similarity data, and (iii) mechanistically explains multi-item interference would form a comprehensive neuro-computational account of VWM.

References

[1] Bays, P. M. (2014). Noise in Neural Populations Accounts for Errors in Working Memory. Journal of Neuroscience, 34(10):3632–3645. 193 citations (Semantic Scholar/DOI) [2024-09-03] Publisher: Society for Neuroscience Section: Articles.

[2] Bays, P. M. (2015). Spikes not slots: noise in neural populations limits working memory. Trends in Cognitive Sciences, 19(8):431–438. 141 citations (Semantic Scholar/DOI) [2024-09-03].

[3] Bays, P. M., Catalao, R. F. G., and Husain, M. (2009). The precision of visual working memory is set by allocation of a shared resource. Journal of Vision, 9(10):7–7. 913 citations (Semantic Scholar/DOI) [2025-03-28].

[4] Bays, P. M. and Husain, M. (2008). Dynamic Shifts of Limited Working Memory Resources in Human Vision. Science, 321(5890):851–854. 465 citations (Semantic Scholar/DOI) [2025-05-28] Publisher: American Association for the Advancement of Science.

[5] Blei, D. M. (2003). Latent Dirichlet Allocation.

[6] Bouchacourt, F. and Buschman, T. J. (2019). A Flexible Model of Working Memory. Neuron, 103(1):147–160.e8. Publisher: Elsevier.

[7] Carandini, M. and Heeger, D. J. (2012). Normalization as a canonical neural computation. Nature Reviews Neuroscience, 13(1):51–62. 1598 citations (Semantic Scholar/DOI) [2024-12-07] Publisher: Nature Publishing Group.

[8] Compte, A., Brunel, N., Goldman-Rakic, P. S., and Wang, X.-J. (2000). Synaptic Mechanisms and Network Dynamics Underlying Spatial Working Memory in a Cortical Network Model. Cerebral Cortex, 10(9):910–923. 1087 citations (Semantic Scholar/DOI) [2025-06-02].

[9] Cowan, N. (2001). The magical number 4 in short-term memory: a reconsideration of mental storage capacity. The Behavioral and Brain Sciences, 24(1):87–114; discussion 114–185. 5971 citations (Semantic Scholar/DOI) [2024-11-28].

[10] Dayan, P. and Abbott, L. F. (2001). Theoretical neuroscience: computational and mathematical modeling of neural systems. Computational neuroscience. Massachusetts Institute of Technology Press, Cambridge, Mass.

[11] Engle, R. W. and Kane, M. J. (2004). Executive Attention, Working Memory Capacity, and a Two-Factor Theory of Cognitive Control. In The psychology of learning and motivation: Advances in research and theory, Vol. 44, pages 145–199. Elsevier Science, New York, NY, US.

[12] Feldmeyer, D. (2012). Excitatory neuronal connectivity in the barrel cortex. Frontiers in Neuroanatomy, 6. 267 citations (Semantic Scholar/DOI) [2024-12-17] Publisher: Frontiers.

[13] Fisher, N. (1993). Statistical Analysis of Circular Data. Cambridge University Press.

[14] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, pages 249–256. JMLR Workshop and Conference Proceedings. ISSN: 1938-7228.

[15] Luck, S. J. and Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. Nature, 390(6657):279–281. 3993 citations (Semantic Scholar/DOI) [2025-05-28] Publisher: Nature Publishing Group.

[16] Luck, S. J. and Vogel, E. K. (2013). Visual working memory capacity: from psychophysics and neurobiology to individual differences. Trends in Cognitive Sciences, 17(8):391–400. 911 citations (Semantic Scholar/DOI) [2025-05-28].

[17] Marr, D. (2010). Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.

[18] Miller, G. A. (1956). The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information[1].

[19] Mnih, V., Heess, N., and Graves, A. (2014). Recurrent Models of Visual Attention.

[20] Peters, B., Rahm, B., Czoschke, S., Barnes, C., Kaiser, J., and Bledowski, C. (2018). Sequential whole report accesses different states in visual working memory. Journal of Experimental Psychology: Learning, Memory, and Cognition, 44(4):588–603. 9 citations (Semantic Scholar/DOI) [2025-05-28] Place: US Publisher: American Psychological Association.

[21] Press, W. H., editor (2007). Numerical recipes: the art of scientific computing. Cambridge University Press, Cambridge, 3. ed edition.

[22] Rigotti, M., Barak, O., Warden, M. R., Wang, X.-J., Daw, N. D., Miller, E. K., and Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. Nature, 497(7451):585–590. 1413 citations (Semantic Scholar/DOI) [2025-05-22] Publisher: Nature Publishing Group.

[23] Sahakian, A., Gayet, S., Paffen, C. L. E., and Van der Stigchel, S. (2025). The rise and fall of memories: Temporal dynamics of visual working memory. Memory & Cognition, pages 1–18. 0 citations (Semantic Scholar/DOI) [2025-05-28] Publisher: Springer.

[24] Schneegans, S., Taylor, R., and Bays, P. M. (2020). Stochastic sampling provides a unifying account of visual working memory limits. Proceedings of the National Academy of Sciences, 117(34):20959–20968. 52 citations (Semantic Scholar/DOI) [2025-05-07] Publisher: Proceedings of the National Academy of Sciences.

[25] Schurgin, M. W., Wixted, J. T., and Brady, T. F. (2020). Psychophysical scaling reveals a unified theory of visual memory strength. Nature Human Behaviour, 4(11):1156–1172. 169 citations (Semantic Scholar/DOI) [2025-05-14] Publisher: Nature Publishing Group.

[26] Strata, P. and Harvey, R. (1999). Dale's principle. Brain Research Bulletin, 50(5-6):349–350. 109 citations (Semantic Scholar/DOI) [2025-05-29].

[27] Sunde, B. M. (2024). early-stopping-pytorch: A PyTorch utility package for Early Stopping. original-date: 2018-12-29T20:15:51Z.

[28] Sussillo, D. and Barak, O. (2013). Opening the Black Box: Low-Dimensional Dynamics in High-Dimensional Recurrent Neural Networks. Neural Computation, 25(3):626–649.

[29] Tibshirani, R. (1996). Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society Series B: Statistical Methodology, 58(1):267–288. 9999 citations (Semantic Scholar/DOI) [2024-12-31].

[30] Tye, K. M., Miller, E. K., Taschbach, F. H., Benna, M. K., Rigotti, M., and Fusi, S. (2024). Mixed selectivity: Cellular computations for complexity. Neuron, 112(14):2289–2303. 29 citations (Semantic Scholar/DOI) [2025-05-22] Publisher: Elsevier.

[31] van den Berg, R., Shin, H., Chou, W.-C., George, R., and Ma, W. J. (2012). Variability in encoding precision accounts for visual short-term memory limitations. Proceedings of the National Academy of Sciences, 109(22):8780–8785. 503 citations (Semantic Scholar/DOI) [2025-05-07] Publisher: Proceedings of the National Academy of Sciences.

[32] Wei, X.-X. and Woodford, M. (2025). Representational geometry explains puzzling error distributions in behavioral tasks. Proceedings of the National Academy of Sciences, 122(4):e2407540122. Publisher: Proceedings of the National Academy of Sciences.

[33] Wimmer, K., Nykamp, D. Q., Constantinidis, C., and Compte, A. (2014). Bump attractor dynamics in prefrontal cortex explains behavioral precision in spatial working memory. Nature Neuroscience, 17(3):431–439. 364 citations (Semantic Scholar/DOI) [2024-09-19] Publisher: Nature Publishing Group.

[34] Zhang, W. and Luck, S. J. (2008). Discrete Fixed-Resolution Representations in Visual Working Memory. Nature, 453(7192):233–235. 1494 citations (Semantic Scholar/DOI) [2025-05-28].

A

A.1 Supplementary Concepts

A.1.1 Divisive Normalisation

Divisive normalization (DN) models how a neuron's response is normalized by the activity of a surrounding population of neurons[7]. Empirical signatures of DN can often be found by examining how a neuron's tuning curve changes depending on the number and arrangement of presented stimuli. Furthermore, if a model exhibits a strict limit on total firing rate, that too can indicate DN is shaping the network's activity.

A.1.2 Ring Attractors

Ring attractors are a class of continuous attractor networks that encode periodic variables such as orientation, head direction, or spatial phase [**?**]. Neurons in a ring attractor are arranged by their preferred feature values (e.g., angles), with local excitatory and distal inhibitory connectivity shaping the recurrent dynamics. This architecture enables the network to sustain localized "bumps" of persistent activity, which can represent a remembered feature value even in the absence of external input.

The attractor manifold forms a topological ring, meaning the bump can smoothly translate along the ring without decay. This makes ring attractors well-suited for modeling memory of circular variables, such as orientation in visual working memory tasks.

A.1.3 Mixed Selectivity

Mixed selectivity refers to the property of a neuron responding not just to individual task variables, but to nonlinear combinations of them [22, 30]. This contrasts with pure selectivity, where a neuron's response depends on a single variable independently.

Formally, let a neuron's firing rate be influenced by two task-relevant variables x and y. We distinguish between:

- **Pure selectivity**: $r = f(x)$ or $r = f(y)$
- **Linear mixed selectivity**: $r = f_1(x) + f_2(y)$
- **Nonlinear mixed selectivity**: $r = f(x,y) \neq f_1(x) + f_2(y)$

The third case—nonlinear mixing—defines true mixed selectivity. It allows the network to encode complex, context-dependent rules in a high-dimensional space, where linearly inseparable problems can become linearly separable.

This property has been shown to be essential for flexible cognition. For example, a neuron that responds only when stimulus A is presented and the current rule is to remember the color—but not shape—demonstrates mixed selectivity. Such units enable circuits to implement conditional logic and context gating.

Importantly, recurrent neural networks trained on context-dependent tasks often spontaneously develop mixed-selective units, mirroring observations in prefrontal cortex.

A.1.4 Poisson Distribution of Spikes

The number of spikes k occurring within a fixed observation window $T(s)$ follows a Poisson distribution:

$$P(k) = \frac{(\lambda T)^k e^{-\lambda T}}{k!}.$$

The mean and variance of the number of spikes is given by:

$$\mathbb{E}[k] = \mathrm{Var}(k) = \lambda T$$

Thus, λ is the firing rate and have unit Hz.

1. **Independent Events:** The occurrence of a spike at one time does not influence the probability of spikes occurring at other times (otherwise, the process would follow a Gamma distribution instead of Poisson).

2. **Homogeneous Rate:** Spikes occur with a constant firing rate λ, meaning the probability of observing a spike in a small time interval Δt is approximately $\lambda \Delta t$.

3. **Exponential Inter-Spike Intervals:** The time between consecutive spikes follows an exponential distribution:
$$P(\Delta t) = \lambda e^{-\lambda \Delta t},$$
which implies that short intervals are more likely than long ones. This property emerges because the Poisson process assumes memorylessness, meaning the probability of a spike occurring does not depend on the time since the last spike.

A.2 Derivation of Model Initialisation

A.2.1 Deriving Recurrent Weights under Dale's Law

In this appendix, we derive a principled initialization scheme for excitatory and inhibitory weights that satisfies Dale's law while preserving the mean and variance of recurrent input currents across neurons. The derivation leads to a closed-form expression for the appropriate initialization standard deviation. We aim to preserve the statistical properties of the recurrent input:

$$(\mathbf{W}\mathbf{r})_i = \sum_{j=1}^{N} W_{ij} r_j,$$

such that:

$$\mathbb{E}[(\mathbf{W}\mathbf{r})_i] = \bar{r}, \quad \mathrm{Var}[(\mathbf{W}\mathbf{r})_i] = \sigma_r^2,$$

assuming each presynaptic rate $r_j \sim (\bar{r}, \sigma_r^2)$. If weights are initialized as $W_{ij} \sim \mathcal{N}(0, \sigma^2)$, then:

$$\mathbb{E}[(W\mathbf{r})_i] = 0, \tag{A.1}$$

$$\mathrm{Var}[(W\mathbf{r})_i] = N\sigma^2 (\bar{r}^2 + \sigma_r^2). \tag{A.2}$$

This implies that naive initialization yields zero mean and requires scaling to match desired variance. However, Dale's Law introduces asymmetry: excitatory and inhibitory neurons project weights of opposite signs. Hence, a more careful initialization is required.

Half-Normal Sampling under Dale's Law

We define excitatory and inhibitory weights as:

$$W_{ij}^e = \sigma_e |\xi_{ij}|, \quad W_{ij}^i = -\sigma_i |\xi_{ij}|, \qquad \xi_{ij} \sim \mathcal{N}(0,1). \tag{A.3}$$

Using properties of the half-normal distribution:

$$\mathbb{E}[|\xi|] = \sqrt{\frac{2}{\pi}}, \qquad \mathrm{Var}(|\xi|) = 1 - \frac{2}{\pi}, \tag{A.4}$$

we obtain:

$$\mathbb{E}[W_{ij}^e] = \sigma_e \sqrt{\frac{2}{\pi}}, \qquad \mathrm{Var}(W_{ij}^e) = \sigma_e^2 \left(1 - \frac{2}{\pi}\right), \tag{A.5}$$

$$\mathbb{E}[W_{ij}^i] = -\sigma_i \sqrt{\frac{2}{\pi}}, \qquad \mathrm{Var}(W_{ij}^i) = \sigma_i^2 \left(1 - \frac{2}{\pi}\right). \tag{A.6}$$

Mean and Variance of Recurrent Input

Suppose each neuron receives N_e excitatory and N_i inhibitory inputs. Then:

$$\mathbb{E}[(W\mathbf{r})_i] = (N_e\sigma_e - N_i\sigma_i)\sqrt{\frac{2}{\pi}}\,\bar{r}, \tag{A.7}$$

$$\mathrm{Var}[(W\mathbf{r})_i] = \left(N_e\sigma_e^2 + N_i\sigma_i^2\right)\left[(\bar{r}^2 + \sigma_r^2)\left(1 - \frac{2}{\pi}\right) + \frac{2}{\pi}\sigma_r^2\right]. \tag{A.8}$$

Assuming a balanced network where $N_e = N_i = N/2$, we solve:

$$\mathbb{E}[(W\mathbf{r})_i] = \bar{r}, \tag{A.9}$$

$$\mathrm{Var}[(W\mathbf{r})_i] = \sigma_r^2, \tag{A.10}$$

which yields the constraint equations:

$$(\sigma_e - \sigma_i)\frac{N}{2}\sqrt{\frac{2}{\pi}} = 1, \tag{A.11}$$

$$(\sigma_e^2 + \sigma_i^2)\frac{N}{2} = \left[(\bar{r}^2 + \sigma_r^2)\left(1 - \frac{2}{\pi}\right) + \frac{2}{\pi}\sigma_r^2\right]^{-1}. \tag{A.12}$$

Letting

$$A = (1 - \tfrac{2}{\pi})\left(1 + \frac{\bar{r}^2}{\sigma_r^2}\right) + \tfrac{2}{\pi},$$

we obtain the closed-form solutions:

$$\sigma_e = \frac{1}{N}\sqrt{\frac{\pi}{2}} + \sqrt{\frac{2}{NA} - \frac{\pi}{2N^2}}, \tag{A.13}$$

$$\sigma_i = -\frac{1}{N}\sqrt{\frac{\pi}{2}} + \sqrt{\frac{2}{NA} - \frac{\pi}{2N^2}}. \tag{A.14}$$

Practical Implementation

For large N and assuming $\bar{r}^2/\sigma_r^2 = 1$, the expression simplifies to:

$$\boxed{\sigma_e = \sigma_i = \sqrt{\frac{2}{N\left(2 - \frac{2}{\pi}\right)}} = \frac{1}{\sqrt{N\left(1 - \frac{1}{\pi}\right)}}} \tag{A.15}$$

This initialization ensures that the recurrent input current has unit variance and non-zero mean, consistent with firing rate statistics, while obeying Dale's law.

A.2.2 Deriving Input Weight Matrix for Excitatory Input

The input matrix $\mathbf{B}$ projects the stimulus vector $\mathbf{h}_t \in \mathbb{R}^{3M}$ into the recurrent neural population. We want to ensure that the resulting input drive $\mathbf{Bu}$ lies within the dynamic range of the activation function (between 10-40 Hz) even when set size $m = 1$.

Input $\mathbf{Bh}_t$ scales *linearly* with set size (i.e. we does not impose divisive normalisation by hand!). So we investigate the scale of input into a single neuron from a single item, denoted by S. We assume B_{ij} is initialised from normal distribution with variance σ^2, then absoluted.

$$S = b_1 h_1 + b_2 h_2 + b_3 h_3 = \sum_{i=1}^{3} |\xi_i|\, h_i(\theta), \tag{A.16}$$

$$\xi_i \sim \mathcal{N}(0, \sigma^2) \quad \text{i.i.d.}, \qquad \theta \sim \mathcal{U}(-\pi, \pi), \tag{A.17}$$

with

$$h_1 = 1 + \frac{\cos\theta}{\sqrt{2}} + \frac{\sin\theta}{\sqrt{6}}, \quad h_2 = 1 - \frac{\cos\theta}{\sqrt{2}} + \frac{\sin\theta}{\sqrt{6}}, \quad h_3 = 1 - \frac{2\sin\theta}{\sqrt{6}}. \tag{A.18}$$

Note that

$$\mu = \mathbb{E}[|\xi_i|] = \sigma\sqrt{\frac{2}{\pi}}, \qquad v = \mathrm{Var}(|\xi_i|) = \sigma^2\left(1 - \tfrac{2}{\pi}\right). \tag{A.19}$$

By linearity of expectation and noting $\sum_{i=1}^{3} h_i = 3$ for all θ,

$$\mathbb{E}[S] = \sum_{i=1}^{3} \mathbb{E}[|\xi_i|]\,\mathbb{E}[h_i] = \mu \sum_{i=1}^{3} h_i = 3\mu = 3\sigma\sqrt{\frac{2}{\pi}}. \tag{A.20}$$

By the law of total variance,

$$\mathrm{Var}(S) = \underbrace{\mathrm{Var}\big(\mathbb{E}[S \mid \theta]\big)}_{=0} + \mathbb{E}\big[\mathrm{Var}(S \mid \theta)\big] \tag{A.21}$$

since $\mathbb{E}[S \mid \theta] = \mu \sum_i h_i = 3\mu$ is constant. By linearity of expectation and noting $\sum_{i=1}^{3} h_i^2 = 4$ for all θ,

$$\mathrm{Var}(S \mid \theta) = \sum_{i=1}^{3} h_i^2 \,\mathrm{Var}(|\xi_i|) = v \sum_{i=1}^{3} h_i^2 = 4v, \tag{A.22}$$

Hence

$$\mathrm{Var}(S) = 4v = 4\sigma^2\left(1 - \tfrac{2}{\pi}\right). \tag{A.23}$$

If we want j-th neuron to get $(\mathbf{Bu})_j = R = 10\text{Hz}$ input when set size $m = 1$,

$$\mathbb{E}[S] = R \to \sigma = \frac{R}{3}\sqrt{\frac{\pi}{2}} = R \times 0.418 \tag{A.24}$$

In practice, we multiply R to input vector $\mathbf{u}$. $R \times M$ is called "input_strength".

A.3 Code Availability

The code used to generate the simulations and figures in this thesis is publicly available on GitHub. The repository includes all relevant scripts and documentation to reproduce the core results. It can be accessed at: https://github.com/derek1909/vwm_rnn.

A.4 Risk Assessment

At the start of the project, the following risk(s) were identified and submitted to the risk assessment office: *Computer Use/DSE (Display Screen Equipment*). Such risks were readily mitigated with regular screen breaks.