RESEARCH ARTICLE

Slow manifolds within network dynamics encode working memory efficiently and robustly

Elham Ghazizadeh[1]*, ShiNung Ching[1,2,3]

1 Electrical and Systems Engineering, Washington University in St. Louis, St. Louis, Missouri, United States of America, **2** Biomedical Engineering, Washington University in St. Louis, St. Louis, Missouri, United States of America, **3** Division of Biology and Biomedical Sciences, Washington University in St. Louis, St. Louis, Missouri, United States of America

* elham@wustl.edu

Abstract

Working memory is a cognitive function involving the storage and manipulation of latent information over brief intervals of time, thus making it crucial for context-dependent computation. Here, we use a top-down modeling approach to examine network-level mechanisms of working memory, an enigmatic issue and central topic of study in neuroscience. We optimize thousands of recurrent rate-based neural networks on a working memory task and then perform dynamical systems analysis on the ensuing optimized networks, wherein we find that four distinct dynamical mechanisms can emerge. In particular, we show the prevalence of a mechanism in which memories are encoded along slow stable manifolds in the network state space, leading to a phasic neuronal activation profile during memory periods. In contrast to mechanisms in which memories are directly encoded at stable attractors, these networks naturally forget stimuli over time. Despite this seeming functional disadvantage, they are more efficient in terms of how they leverage their attractor landscape and paradoxically, are considerably more robust to noise. Our results provide new hypotheses regarding how working memory function may be encoded within the dynamics of neural circuits.

Author summary

The ability to remember information for brief periods of time before using it is a key human ability. For example, retaining a phone number for a few moments prior to entering it into a keypad. Such ability, known as working memory, enables many more complex functions such as planning and reasoning. In this paper, we use theory and computational modeling approaches to try and better understand how circuits and networks in the brain might be achieving working memory. Specifically, we construct hypothetical network models to perform a task that embodies essential aspects of working memory. We then dissect our model to reveal how its components—simulated neural units—interact with each other to represent and maintain information. It turns out that

some models can achieve memory by maintaining a fixed representation over time, i.e., the units remain 'still' during memory. However, other models fluctuate their activity during memory in a particular way that is seemingly quite efficient and less susceptible to distraction. In total, our computational study provides new theory for how this form of memory might be implemented in brain networks.

This is a *PLOS Computational Biology* Methods paper.

Introduction

Working memory (WM) is a temporary store that allows for active manipulation of information in the absence of external stimuli [1]. Critical cognitive functions such as reasoning, planning and problem solving rely on working memory and thus its mechanistic basis is a key question in brain and cognitive science. Presumably, memory retention relies on an invariant latent neural representation of past stimuli [2], but the precise nature of these representations and the dynamical mechanisms by which they are created in neural circuits remain enigmatic. Experimental and theoretical characterizations of working memory typically center on a delay period that occurs after stimulus presentation and before onset of a behavioral response or action [3–6]. Characterizations of neural activity during delay periods dichotomize into two broad categories: (i) persistent, tonic activity and (ii) time varying, phasic activity. In the former, neurons are tuned to relevant features of a stimulus and produce elevated and relatively constant activity throughout the delay [7–9]. In the latter, neuronal activity fluctuates, ramping up and down during delay periods [5, 10, 11]. Tonic and phasic paradigms have been observed in working memory tasks *in vivo* [10, 11] and in computational models [12, 13]. However, the mechanisms underlying these descriptions and the reasons why one may manifest over the other in certain circumstances is far from clear.

Understanding the network mechanism of working memory often revolves around the role of self-sustaining attractors, including discrete fixed points [14], which correspond to neuronal activity patterns that are maintained indefinitely in the absence of exogenous stimuli or perturbation. Tonic delay activity is thought to coincide with such attractors [14–16], thus allowing for stable maintenance of memory representations for potentially arbitrary lengths of time.

On the other hand, in the phasic hypothesis memory representations do not coincide with self-sustaining attractors. Instead, high-dimensional neuronal activity fluctuations may project onto a lower-dimensional latent space upon which an invariant representation is held during delay intervals [17, 18]. For example, if the activity of a neuron gradually drops, the activity of another neuron increases to compensate for that drop. Thus, during delay, neural activity may traverse a low-dimensional manifold corresponding to this invariant representation [13, 19].

Disambiguating the above mechanisms requires deriving an understanding of the generative processes that give rise to time-varying, task-evoked neural activity. Ideally, we would be able to analytically characterize these mechanisms in a dynamical systems framework that could reveal the details of the attractor landscape embedded within neuronal networks.

However, ascertaining dynamical systems models of biological networks is not straightforward, especially at a level of scale commensurate with networks thought to be relevant to WM, such as prefrontal cortex [20, 21]. In this regard, artificial recurrent neural networks

(RNNs) can form an interesting and potentially useful surrogate from which to derive mechanistic hypotheses. Such networks can be optimized in a top-down fashion to engage high-level cognitive tasks that include WM requirements [22–26]. Then, the emergent dynamics of the synthesized model can be analyzed and used to make arguments for or against different mechanisms, based on the predictive validity of the model outputs relative to actual brain activity.

In this spirit, recent works have tried to reconcile the aforementioned hypotheses regarding persistent vs. transient delay activity in the context of WM [27–30]. Orhan and colleagues [30] optimized RNNs to perform several short-term memory tasks and they observed that both tonic and phasic delay activity could arise, depending on specific task details and optimization/learning parameters. Similarly, Nachstedt and colleagues [27] showed that the existence of both mechanisms simultaneously can mediate reliable task performance in the face of uncertain stimulus timing. However, it remains unclear what factors sway RNNs to manifest one mechanism over another and, related, whether they carry different functional advantages.

With regards to the last point, the ability of optimized RNNs to predict actual brain activity may depend crucially on certain restrictions regarding the optimization method that is used, e.g., by encouraging solutions that manifest connection motifs that are more biologically realistic [13, 31]. Thus, using RNNs to build potentially explanatory hypotheses regarding neural circuit mechanisms likely requires careful consideration of the numerical optimization strategy used, including hyperparameters and initialization schemes, as well as prior constraints on network architecture [25, 32]. Expanding on this idea, in this work, we pursue the top-down RNN approach to study potential mechanisms underlying WM function, training thousands of networks to perform an analytically tractable sequential, memory-dependent pattern matching task. To train our network, we modify the First-Order Reduced and Controlled Error (FORCE) [33] method by using a temporally restricted error kernel to confine error regression to occur only during brief intervals within each trial. The proposed framework blends trial-based reinforcement learning with first-order regression, thus obviating the need for a continual external supervisory error-signal.

Our premise is that this revised optimization framework, by leaving long epochs unconstrained, may allow for a wider range of possible emergent dynamics. Indeed, by optimizing RNNs across different hyperparameters and initialization schemes within this framework, we identify a diversity of network mechanisms, each achieving the desired function, but varying in their key dynamical properties. We find that networks can embed predominantly asymptotically stable fixed points, stable limit cycle attractors, or a combination thereof. Most interestingly, we show here that there are two distinct mechanisms by which stable fixed points can be used to serve memory encoding, one leading to tonic activation and the other leading to phasic activation. We show that the latter, while unable to sustain memories over arbitrary lengths of time (i.e., wherein the model 'forgets') nonetheless constitutes a more efficient and robust mechanism by which memories can be encoded.

Results

Working memory can be encoded via distinct dynamical mechanisms associated with tonic and phasic neural activation

We enacted a trial-based WM task involving sequential pattern matching (SPM) that exhibits working memory requirements (Fig 1a). In our design, high-dimensional stimuli are encoded as bivariate random processes, such that the network is required to temporally integrate each stimulus and then store a latent representation of said stimulus for later processing. We optimized RNNs to perform this task by using a modified FORCE method [33] that included a
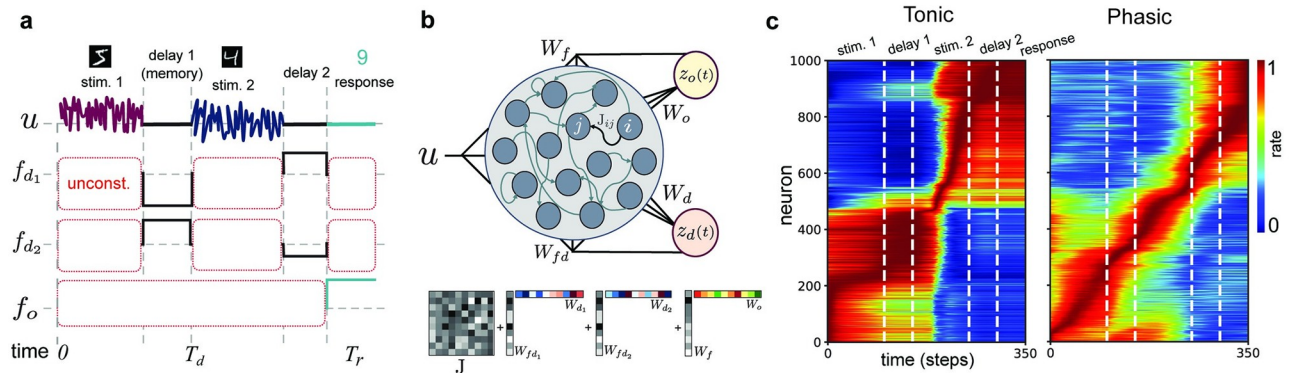
Fig 1. Optimizing RNN using modified FORCE to perform a sequential pattern matching (SPM) task. a, A single trial of a SPM task, wherein low-dimensional random process representations of handwritten digit stimuli are followed by short delay intervals. The network is optimized to generate the correct 'summation' output during a prescribed response interval. **b**, Schematic diagram of RNN architecture and low-rank structure added to initial connectivity **J**. After the network receives input trials sequentially via input weights, it encodes the memory representation $z_d(t)$ and generates the task outputs $z_o(t)$. We use a rank 2 structure for encoding memory and a rank 1 structure for generating response. **c**, Tonic and phasic activity for two different networks. Activity patterns (normalized) of neurons are sorted by the time of their peak value.

https://doi.org/10.1371/journal.pcbi.1009366.g001

temporally restricted error kernel. Here, regression occurs at two phases during each trial: (i) during memory/delay periods, wherein we promote the formation of an invariant latent linear projection from neural units nominally associated with maintenance of a memory representation; and (ii) at the conclusion of each trial, wherein we promote a linearly decoded output response signal (Fig 1a and 1b). All other temporal epochs are unconstrained, thus obviating the need to generate an error signal continuously throughout trials, which may overly constrain the dynamics [34] (see also Methods for additional details and S1 Fig).

We found that optimized networks could produce both tonic and phasic activity patterns during delay periods, as exemplified for two different networks of 1000 neurons in Fig 1c. In order to study the dynamical mechanisms underlying these overt patterns we first used a numerical criteria on neuronal activity at the end of the delay period, T_d. Specifically, we arrested trials at T_d and forward simulated the networks autonomously to ascertain whether the activity was sustained at a fixed point (see Methods). We identified four distinct dynamical mechanisms that could mediate working memory. In the case of tonic activation, network activity would indeed remain persistent, i.e., $\mathbf{x}(t)$, the state vector of neuronal activity, would remain near $\mathbf{x}(T_d)$ with $\|\dot{\mathbf{x}}\| \simeq 0$, indicative of a fixed point attractor (Fig 2a). We refer to this mechanism as direct fixed point encoding (DFP). In the case of phasic patterns, $\mathbf{x}(t)$ in the forward simulation would deviate from $\mathbf{x}(T_d)$. In some cases, the network would always settle at a *different* fixed point from the memory representation (Fig 2b, termed indirect fixed point encoding, IFP), independent of the stimulus or network initial condition. In other cases the network would always asymptotically approach a stable limit cycle attractor (Fig 2c, limit cycle encoding, LC). In a fourth case (not depicted), the network could asymptotically approach either a disparate fixed point or a limit cycle and exhibit either tonic or phasic activity, depending on the stimulus realization (termed mixed encoding, see S3 Fig). In total, we optimized 1524 network models, of which 703 were identified of the direct fixed point (DFP) mechanism, 534 were of the indirect fixed point (IFP) mechanism, 182 were of the limit cycle (LC) mechanism, and 105 were of the mixed (Mix) mechanism. Given their dominance in the emergent solutions, our subsequent attention will be on understanding the workings of the DFP and IFP mechanisms, though we will later also untangle the factors that cause each mechanism to arise over the others.
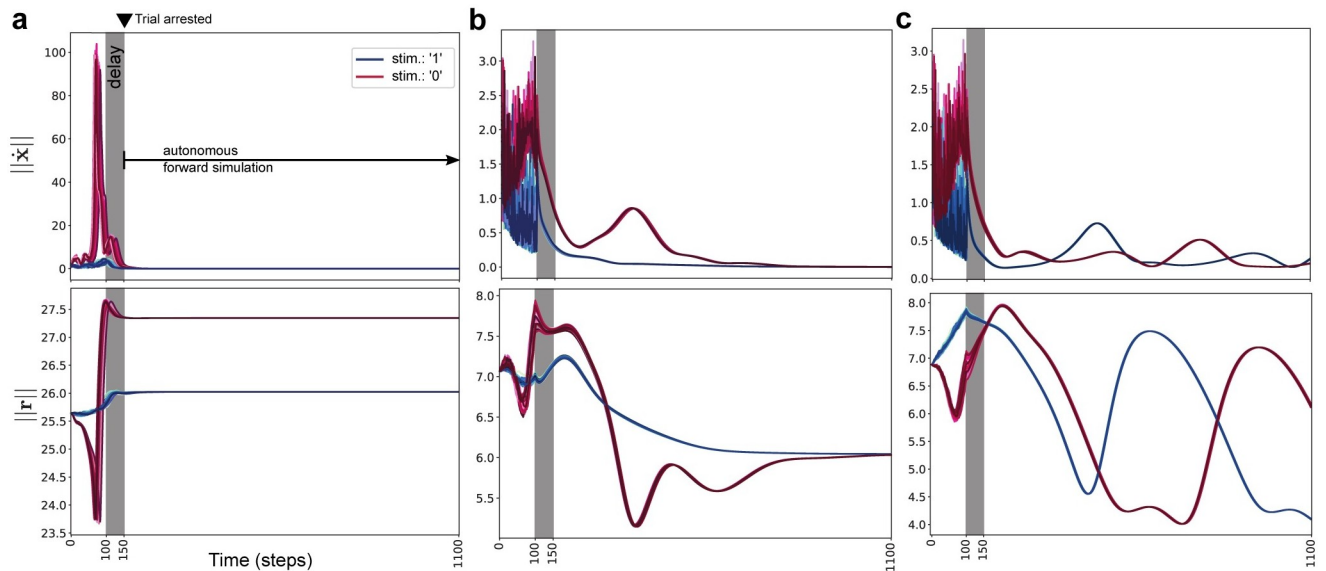
Fig 2. Forward simulation of network after delay to identify distinct dynamical mechanisms underlying WM. a, Direct Fixed Point encoding (DFP), where the network uses fixed points to encode memory representations of each stimulus. **b**, Indirect Fixed Point encoding (IFP), where the network asymptotically settles at a fixed point but this fixed point does not correspond to a memory representation. **c**, Limit Cycle (LC), where the network asymptotically approached a stable limit cycle attractor. For all plots, colors in the spectrum of blue show different realizations of first stimulus while colors in the spectrum of red show different realization of second stimulus.

https://doi.org/10.1371/journal.pcbi.1009366.g002

Indirect encoding efficiently uses the network attractor landscape

The above findings suggests that key invariant structures in the network attractor landscape—stable fixed points and attractive limit cycles—determine whether and how delay activity takes on a tonic or phasic characteristic. To delve further into these mechanisms, we attempted to analyze how networks in each of the four categories leverage their respective attractor landscapes during the task.

We began by linearizing the dynamics at the origin and using mean-field results [35] to establish lower bounds on the number of fixed point attractors manifest in the network attractor landscape. Fig 3a and 3b show how our four mechanistic categories break down along three key properties of spectra of the ensuing Jacobian matrix, where distinctions are readily observed. Most notably, the landscapes associated with direct fixed point encoding involve a greater number of fixed point attractors relative to indirect encoding. In support of this point, Fig 3c illustrates representative low-dimensional projections of network activity in each of the four mechanisms with stable fixed points overlaid (here, we restrict attention to the positive quadrant, see also Discussion). In DFP encoding (Fig 3c), the sequential stimuli move the trajectory between different fixed points (associated with memory representations), culminating in an output that is itself associated with a different fixed point (i.e., here a total of four fixed points are used in the service of the task). In contrast, the landscape for IFP encoding (Fig 3c) involves a *single* fixed point that does not encode memories, nor does it encode the nominal output (though, it is approached asymptotically if networks are forward simulated autonomously after trial cessation). Thus, IFP encoding is able to maintain invariant representations during the relevant memory periods without relying directly on the presence of multiple fixed point attractors (see also S3 Fig).
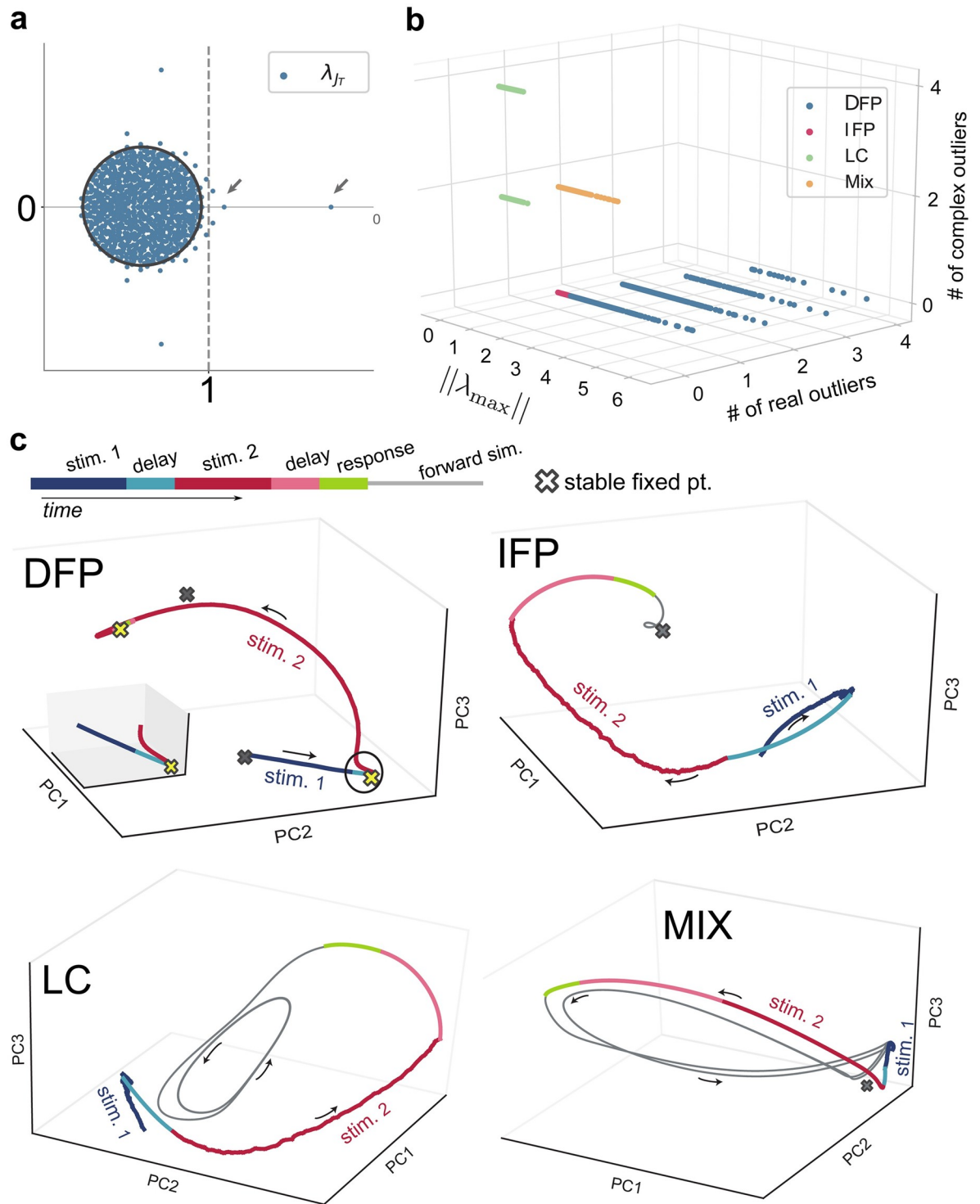
Fig 3. Attractor landscape for optimized networks. a, Eigenvalue spectrum λ_{J_T} (for an exemplar DFP network). The gray circle shows the radius of the theoretical eigenvalue spectrum of the initial connectivity matrix J. After optimization, a set of outliers emerges in the eigenvalue spectrum. Here, the initial connectivity matrix is the Jacobian at the origin (shifted by $-I$). **b**, Categorization of four distinct mechanisms along key properties of the network Jacobian evaluated at the origin. λ_{max} is the eigenvalue with the largest real part in the eigenvalue distribution of the Jacobian matrix. An eigenvalue is an outlier if it is outside the radius of the circle depicting the theoretical boundary of the eigenvalue spectrum of J. **c**, Attractor landscape and trajectory of exemplar task trials. Three-dimensional neural trajectories are obtained via applying Principle Component Analysis (PCA) to 1000-dimensional neural activity from networks of each dynamical mechanism. In DFP, the network creates 4 stable fixed points to solve the SPM task. For the displayed trajectory, the network

uses two fixed points (shown in yellow) to directly encode the memory and trial output (the inset shows the area inside the circle). In IFP, the memory representation and trial output are encoded along the slow manifold of the single fixed point in the state space. In LC, the trajectories approach a stable limit cycle. For the mixed mechanism, both a stable fixed point and limit cycle are observed.

Indirect encoding uses slow manifolds to sustain memory representations

Following from the above, IFP encoding appears to use the geometry of the stable manifolds of the single fixed point to maintain memory representations. Fig 4a illustrates the spectrum of the linearized dynamics about the fixed point in the previous IFP example encoding model, where we see many eigenvalues near the imaginary axis, indicating the presence of slow, stable manifolds along which activity flows in a relatively invariant fashion. These manifolds provide the opportunity for a low-dimensional latent representation of memory to be maintained, despite phasic activity (along the manifold) [8, 36, 37]. Indeed, because we encourage linearly mapped latent representations via our optimization method (see Methods), we know these manifolds have a planar geometry in the firing rate activity variables. In contrast, Fig 4b illustrates the spectra resulting from linearization about two memory fixed points in a DFP model.
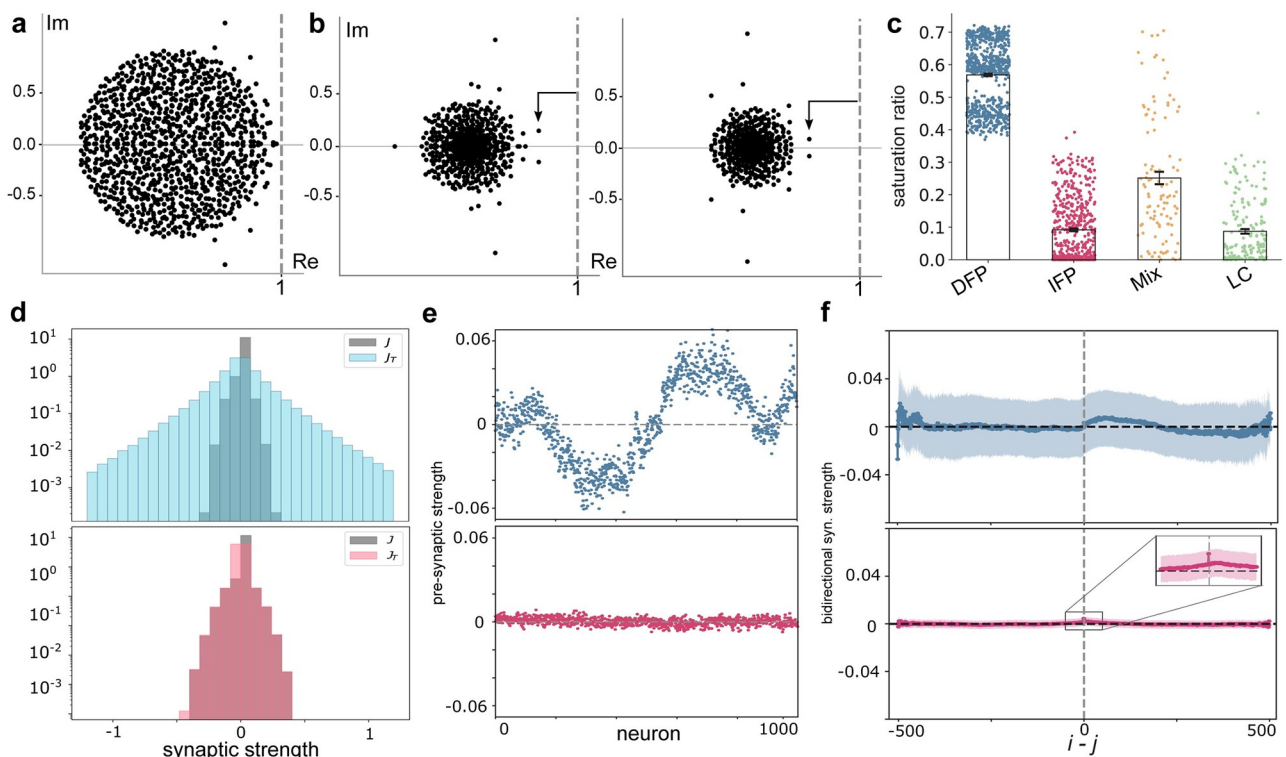

Fig 4. Eigenvalue spectrum at task fixed points and connectivity characterization. a, Eigenvalue spectrum of Jacobian matrix at the single non-zero stable fixed point of IFP (shown in Fig 2b). **b**, Eigenvalue spectra of the Jacobian matrix computed at memory fixed points of DFP, wherein the network uses two fixed points to encode memory representations associated with stimulus 1 and stimulus 2. (shown in Fig 2a). **c**, Saturation ratio (the ratio of neurons with activity in saturated range of activation function during memory interval (averaged over all trials)) for all networks simulated (across all four mechanisms). Standard error of the mean is depicted. **d**, Distribution of connectivity matrix entries (i.e., weights) before and after training for DFP (the top panel) and IFP (the bottom panel). J and J_T denote connectivity matrix before and after training, respectively. **e**, Average pre-synaptic (incoming connections) strength sorted by peak activation of neurons (as in Fig 1c) for DFP and IFP, respectively. **f**, Comparison of mean and variance of elements of task connectivity matrix based on temporal distance of neurons. For IFP (the bottom panel) temporally adjacent neurons are more tightly coupled and a peak can be observed. (The inset shows this peak and i, j denote neurons indices.).

Here we note that eigenvalues are relatively offset from the imaginary axis, indicating rapid convergence to the fixed point. This conclusion is supported in Fig 4c, which shows the relative proportion of delay periods in which neurons are in the saturated (nonlinear) vs. linear range of the activation function, i.e. tanh(.), for each model we trained. The much larger proportion of saturated neurons in DFP encoding indicates that the mass of eigenvalues for these models is relatively contracted and offset from the imaginary axis (see Methods and Eq (10)) and thus associated with fast decay to the fixed points [37].

To further understand the circuit-level details mediating the DFP and IFP mechanisms, we characterized the connectivity between neurons. We first noted that DFP encoding leads to an overall much greater distribution of connectivity weights between neurons relative to IFP (Fig 4d). Next, we sorted neurons according to their peak activation (as in Fig 1c) and examined their average pre-synaptic activity throughout the course of trials. We found that neurons in DFP encoding exhibited highly structured synaptic tuning to different stimuli and memory periods, in contrast to IFP encoding (Fig 4e). Finally, we examined the bidirectional synaptic weight between 'adjacent' neurons (ones with temporally sequential maximal activation). Here, DFP exhibits no systematic connectivity structure, while IFP shows that neurons with similar peak activation times are more tightly coupled (Fig 4f). This latter point suggests that traversal along the slow manifolds is mediated by an internal sequential, 'daisy chain' type of structure embedded within the trained IFP encoding network.

Stable manifold encoding is forgetful, but robust

We sought to better understand the functional advantages of the different mechanism types. In this regard, we interrogated networks by extending delay periods beyond the nominal training requirements, a form of increased memory demand. The main question here is how increasing the memory demand in this way would affect activity and consequently degrade task performance. Fig 5a illustrates the comparison of neural activity patterns for DFP and IFP encoding categories (with extended delay equal to five times the nominal delay interval). For DFP encoding, regardless of the length of the extended delay, the neural activity is unaffected since the network uses fixed points as the invariant structure to encode memory traces. Consequently, after the extended delay ends and the network receives the second stimulus, task computations can be executed correctly. However, for IFP encoding, during the extended delay interval, neural activity gradually drops away which results in loss of function due to deviation from the 'correct' activity pattern upon receiving the second stimulus. Fig 5b summarizes the deviation from the nominally 'correct' post-delay neural activity as a function of delay extension for our two FP mechanisms, as well as LC and Mix. As expected, for DFP encoding this deviation is near zero. In contrast, for IFP the networks can tolerate extended delay up to % 100 of the nominal delay, after which point performance gradually drops, i.e., the correct representation is 'forgotten'.

To assay other functional aspects of these mechanisms, we examined how performance of our networks would tolerate the presence of a distracting noise added to the actual stimulus. Here, we found a counterintuitive functional advantage of 'forgetting,' relative to the DFP mechanism. We specifically added uncorrelated noise of differing variance to the first of the two sequential stimuli and examined deterioration from the nominal 'correct' neural representation at trial conclusion. For values of noise variance that are less than the stimulus variance (vertical dotted line Fig 5c), IFP and LC encoding are highly robust to perturbations, and indeed variances in excess of an order of magnitude greater than the stimulus can be tolerated. In stark contrast, DFP encoding is highly fragile with respect to distracting noise, with
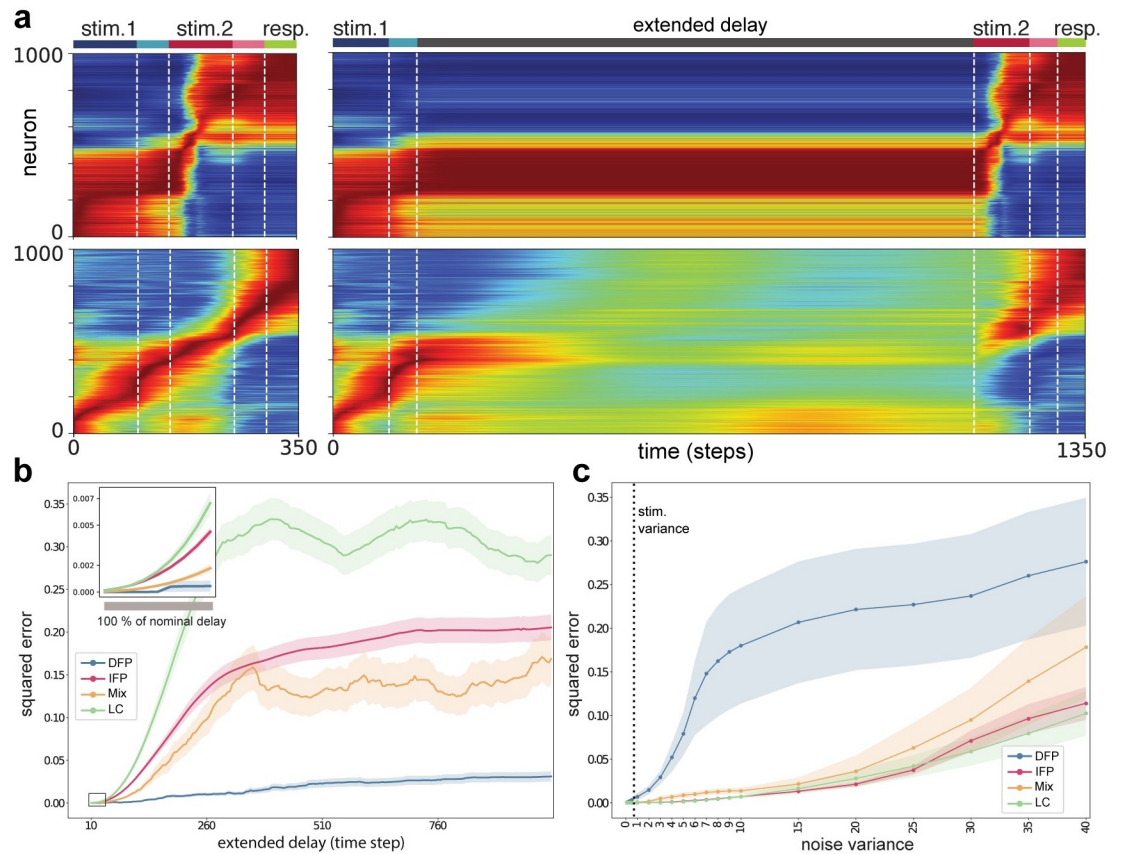
Fig 5. Functional advantages/disadvantages of each mechanism type. a, Comparison of activity patterns before and after increasing memory demand for DFP (top panel) and IFP (bottom panel). **b**, Summary of deviation from correct pattern of activity across different values of extended delay for all optimized networks. The squared error shows the difference between correct and deviated trial outputs averaged over all trials and associated networks. **c**, Summary of deviation from correct pattern of activity across different values of noise variance for all optimized networks. **b** and **c** show that IFP, LC and mixed mechanisms are forgetful, but robust to sizable perturbations.

rapid and near-complete breakdown of the correct neural representation after modest perturbation (Fig 5c). To understand this mechanism we carefully studied the trajectories in low-dimensional space in the presence of distracting noise (S4 Fig), from which we ascertained that the distracting noise was placing the trajectory in an erroneous basin of attraction, i.e., causing an incorrect memory fixed point to be asymptotically approached. This result appears to run counter to classical Hopfield-type associative memory theory [38], which presumes that basins are useful to rejecting noise and uncertainty. Our results suggest that a short-term working memory mechanism that is reliant on excessive creation and use of fixed point attractors is not robust to persistent distracting stimulus noise (see also Discussion).

Initial network properties dictate the emergence of different solution dynamics

Finally, we sought to understand the factors prior to optimization that bias the emergent dynamics towards one type of mechanism versus another. We considered three main network properties: (i) the strength of connectivity, g, (ii) the variance of feedback, σ_f, and (iii) the
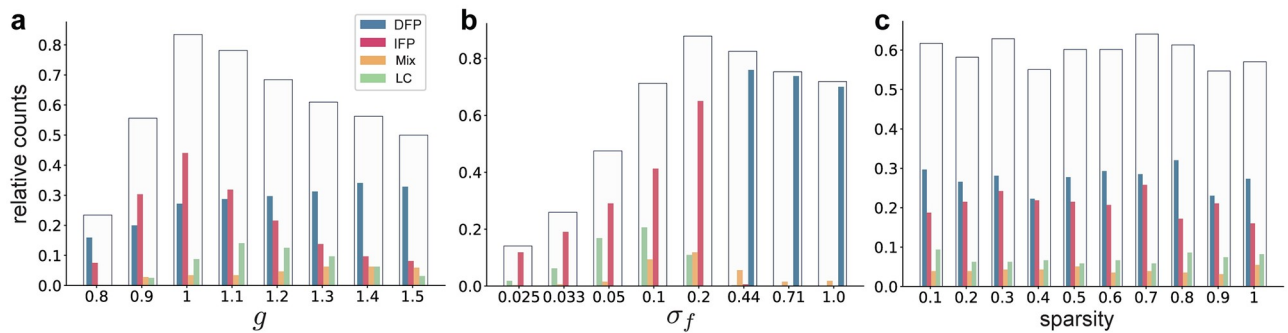
Fig 6. The effect of parameters prior to optimization on the diversity of the emergent solutions. Relative count shows the number of trainable networks divided by the total number of networks for each specified value of parameters. Transparent bars show the relative count of trainable networks and the inner bars show the corresponding emergent types for each specified value of parameters. **a**, The strength of connections within the initial connectivity matrix J. **b**, The variance of feedback weights. **c**, The sparsity of J.

sparsity of the initial connectivity matrix. We varied these parameters over their possible ranges. Fig 6 illustrates the effect of different parameterizations: for small values of g the trainability of networks is poor, but improves significantly as g increases. In other words, large random initial connectivity facilitates training, consistent with known results [33, 35]. For $g < 1$ the untrained network has one stable fixed point at the origin and the emergent trained dynamics tend to be of DFP or IFP encoding (Fig 6a). Also, note that networks with DFP are not chaotic after optimization even for large g, because the contribution of the low-rank component is much larger than the initial connectivity matrix. Interestingly, the variance of feedback weights, σ_f has a notable effect on the emergent dynamics; for large values of σ_f the networks tend to form DFP models and as σ_f decreases only IFP and LC models arise (Fig 6b). The sparsity of initial connectivity matrix has no significant effect on the trainability of networks nor the emergent dynamics (Fig 6c).

Discussion

Learning a diversity of dynamics for working memory function

In this work we used a top-down optimization-based approach to investigate potential dynamical mechanisms mediating WM function. By training/optimizing RNNs using a modification of the FORCE regression method, we found four qualitatively different types of network dynamics that can mediate function. At a mechanistic level, these solutions are differentiated on the basis of the number of asymptotically stable fixed points manifest in the network vector field and, crucially, how those fixed points are leveraged in the service of the task. We note especially two solution types, one reflecting neural memory representations that are highly persistent corresponding to direct encoding at fixed points (i.e., DFP), versus the other where neural representations are transient and correspond to traversal along slow manifolds in the network state space (i.e., IFP). At the level of neural activity, DFP produces tonic sustained activity during delay periods, while IFP produces phasic, transient activity.

Our results are related to prior work that has shown that persistent versus transient encoding of memories can manifest in neural networks trained on different WM tasks and under different optimization/learning schemes [30]. Here, we choose to focus on a single, structured task in an effort to go beyond overt activity characterizations and carefully dissect the underlying dynamical mechanisms associated, namely the attractor landscape in the neural state

space. Doing so provides not only insight into potential generative circuit processes but also allows us to perform sensitivity analyses to ascertain nuanced functional advantages associated with the different mechanisms.

Tradeoff between efficiency, memory persistence and robustness

In particular, our results suggest an interesting balance between persistence and robustness of memory representations. Specifically, the DFP mechanism resembles traditional associative memory attractor dynamics, in the sense of Hopfield networks [38]. Here, each memoranda is associated with a distinct, asymptotically stable fixed point. On the one hand, such a mechanism is able to retain memories for arbitrary lengths of time. Further, the dynamics within the attractor basins can nominally correct for small perturbations to neural trajectories at the onset of memory periods. However, our results add caveats to this latter classical interpretation. Specifically, our DFP analyses suggest that noise robustness breaks down when considering sequential, time-varying stimuli. In this case, perturbations to stimuli can accrue over time and along trajectories, causing neural representations to stray into errant basins of attraction, ultimately leading to failure of performance. Our finding, in essence, indicates that the high reliance on many fixed points for stable memory representations in DFP encoding makes this mechanism more susceptible to the temporal integration of noise.

In contrast, in the IFP encoding mechanism, the network vector field exhibits a smaller number of fixed points that do not encode memoranda directly. Rather, memory representations are formed from projection of neural activity along slow manifolds that are ostensibly shaped through optimization of the network vector field. The fixed points here are, in essence, 'shared' between memoranda. This mechanism turns out to be far more robust to time-varying stimulus perturbations. There are likely two factors related to this robustness. First, noisy perturbations may not be able to easily move trajectories off of the 'correct' slow manifold. Second, there are no competing attractors to absorb errant trajectories, as would be the case in the DFP mechanism. In total, the IFP encoding can be viewed as an overall more efficient use of neural dynamics wherein the lack of a persistent representation (i.e., 'forgetfulness') is offset by both a lighter weight coding scheme in terms of the number of attractors deployed in the state space, leading—perhaps paradoxically—to more robust performance.

Shaping a landscape with few attractors

Expanding on the above point of efficiency, it is of note that the limit cycle and mixed mechanisms are most comparable to IFP in terms of the way in which the attractor landscape is used in the service of the task. In the LC mechanism in particular, the oscillatory cycle is not itself used to encode or sustain the memory, but rather shapes the landscape to create slow manifolds for encoding, similar to IFP. Thus, while the oscillation is not directly functional, it nonetheless is critical in establishing the 'correct' landscape for task completion. From an energetic standpoint, the indirect mechanisms are potentially less expensive since most neurons are inactive at any moment in time, in contrast to DFP encoding. This latter point is evidenced in Fig 4c, where we see that in DFP a much greater proportion of neurons are at activity saturation, relative to the other encoding strategies. Interestingly, we found that for networks to implement the DFP mechanism (i.e., by directly using a large number of fixed points), stronger perturbations of the initial weights are required. Relative to the smaller perturbations of IFP encoding, the larger modification of the initial weights may enable an associated large alteration to the dynamics of the system, which creates a more structured dynamical landscape consisting of multiple fixed points and basins of attractions.

Temporally restricted optimization promotes solutions that are compatible with observed dynamics *in vivo*

Our results shed light on the different means by which recurrent networks can embed memory functions within their dynamics. These findings suggest mechanistic interpretations for actual WM circuits in the brain, given the seeming prevalence of phasic activity patterns during delay intervals observed *in vivo* [20]. Indeed, it has been observed that neurons in memory-relevant regions such as prefrontal cortex do not necessarily maintain persistent activity throughout long delay periods, but rather may 'ramp' on and off at systematic time points [20], as is compatible with our IFP mechanism. Further, in the IFP mechanism, most neurons are lightly saturated (Fig 4c), meaning that most neurons are within a linear regime, as thought to occur in actual neural circuits [34, 39].

Notably, the IFP dynamical mechanism only arises after using the proposed temporally restricted error kernel. Indeed, we found that using the native FORCE method without such a kernel leads to poor trainability; and further those networks that do manage to be trained are highly fragile to the extended delay and noise perturbations we considered. This fragility ostensibly arises due to the latent outputs being overly constrained in this situation. Indeed, the choice of how to constrain these outputs throughout the task is somewhat arbitrary in the first place. Hence, the temporally restricted error kernel may be allowing for the emergence of more naturalistic dynamics in our RNNs.

Potential for enhanced fast learning and generalization

An important technical caveat is that we have set up our RNNs to produce activity in the positive quadrant. Hence, our analysis focuses on characterization of the attractor landscape in that region of the state space. However, because we consider an odd activation function, we know analytically that the fixed points analyzed in our networks have 'mirror' negative fixed points that are not directly used in the service of the task, which means that these dynamical features are essentially 'wasted' by construction and network design. A speculative hypothesis is that these fixed points may allow the network to more quickly learn a related task with minimal synaptic modification, i.e., by leveraging the mirror dynamics that are already embedded in the network. Such a concept is related to the idea of meta-learning [40] and may be an interesting line of future study.

Limitations

It is important to note and emphasize that our framework uses a prior optimization/training paradigm. In other words, after optimization, our connectivity matrix J is static. This is distinct from short-term plasticity models of working memory wherein synaptic weights are continually updated according to an online, activity-dependent learning rule, i.e., wherein the connectivity matrix is a function of time i.e. $J(t)$ [41, 42], during training and testing. Most biological synaptic plasticity rules rely on local activity only, i.e., each synapse evolves as a function of pre- and post-synaptic activity. In our work, synaptic connections are modified via an additional low-rank synaptic connectivity matrix (optimized via a modified FORCE method) that uses the full network activity. Hence, our work is different from synaptic plasticity mechanisms that adjusts the network connectivity dynamically to induce working memory encoding. Assessing the relationship between our optimization paradigm and an online learning rule is left for future study.

We also note that our study has focused on general network dynamical mechanisms, without implicating specific brain regions. Working memory in broader cognitive settings is

thought to involve the interaction of many regions, as has been described through resting state network analyses [43]. Our focus here has not been on these macro-scale functional interactions, but on more granular questions regarding how particular networks might generate activity that reliably and robustly represent memories. We posit that such mechanisms would be embedded in the regions relevant to working memory, e.g., DLPFC, and thus might manifest different functional interactions between these and other brain regions [44], though a full examination of this issue is not performed here.

From a technical standpoint, a key detail of our approach is the imposition of latent representations (i.e., z_d) during the training phase. We found that the inclusion of these enforced latent representations greatly aided training convergence and stability. Eliminating these latent representations led to training instability, similar to those encountered in reservoir computing strategies, e.g., [23]. Notably, alternative optimization frameworks such as backpropagation through time [30] have been used to produce networks without overt latent representation. However, these approaches rely on regressing the error gradient over long temporal epochs, as opposed to the step-wise, reinforcement-based modification used herein. We found that this training paradigm was essential to giving rise to the diversity of dynamical mechanisms we studied.

Methods

Working memory task details

In this study, we considered a sequential pattern-matching task that takes into account key aspects of working memory tasks: stimulus processing, memory encoding and response execution [20]. Our goal was to use a task of sufficiently low dimension as to be able to perform tractable and potentially illuminating post-hoc analysis on the emergent dynamics of RNNs. In the proposed task, each trial consists of two random process stimuli that are sequentially presented and interleaved with delay intervals, followed by a brief response interval (Fig 1). Each stimulus is a two-dimensional Gaussian process obtained in the latent space of a Variational Auto Encoder (VAE) trained on the MNIST dataset of hand-written digits (see S5 Fig). We designed the task pattern association rule to emulate summation, which differs from simple match or non-match tasks [45]. Specifically, to keep the dimensionality of the task low, we use two different stimuli resulting in 3 potential task outcomes (for summation).

Recurrent network model

We considered recurrent networks composed of N nonlinear firing-rate units specified by:

$$\tau \, \dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \boldsymbol{J}\mathbf{r}(t) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^N$ is the state vector and $\mathbf{r}(t) = \tanh(\mathbf{x}(t))$ denotes the activity obtained via applying hyperbolic nonlinearity to the neuronal state variables. We set the network time constant, $\tau = 1$ for simplicity. Here, $\boldsymbol{J}$ is the (untrained) synaptic connectivity matrix with elements drawn randomly from a Gaussian distribution, i.e. $\boldsymbol{J}_{ij} \sim \mathcal{N}(0, \sigma_J^2)$. Specifically, we parameterize $\sigma_J^2 = \frac{g^2}{N}$, so that g controls the strength of synaptic interactions.

Optimization method

Recurrent networks with fully random connectivity as in Eq (1) have a rich dynamical repertoire and thus are capable of generating complex temporal patterns that are commensurate with spontaneous cortical activities [12, 46]. To make these networks learn the function of interest and thus perform the task, we first define two variables decoded from the network

activity:

$$z_o(t) = \mathbf{W}_o^T \, \mathbf{r}(t)$$
$$z_d(t) = \mathbf{W}_d^T \, \mathbf{r}(t) \tag{2}$$

where $z_o(t)$ is a network output for generating responses, while $z_d(t)$ is a low-dimensional latent variable that is linearly decoded from neural firing rate activity, i.e. $z_d(t) = (z_{d_1}(t), z_{d_2}(t))$. In our network, invariant memory representations will be formed in this latent space. Optimization/learning proceeds by modifying the projection vectors $\mathbf{W}_o \in \mathbb{R}^{N \times 1}$ and $\mathbf{W}_d \in \mathbb{R}^{N \times 2}$. As such, $\mathbf{W}_d$ linearly maps neural activity to invariant memory representations during delay intervals. The network output $z_o(t)$ and dummy output $z_d(t)$ are fed back to the network via feedback weights i.e. $\mathbf{W}_f \in \mathbb{R}^{N \times 1}$ and $\mathbf{W}_{fd} \in \mathbb{R}^{N \times 2}$, respectively. This results in modified synaptic connectivity:

$$\dot{\mathbf{x}}(t) = -\mathbf{x}(t) + (\boldsymbol{J} + \mathbf{W}_f \mathbf{W}_o^T + \mathbf{W}_{fd} \mathbf{W}_d^T)\mathbf{r}(t) + \mathbf{W}_i \mathbf{u}(t) \tag{3}$$

The elements of $\mathbf{W}_f$ and $\mathbf{W}_{fd}$ are drawn independently from Gaussian distributions with zero mean and variance σ_f^2. The network receives the exogenous input (i.e., stimulus) $\mathbf{u}(t) \in \mathbb{R}^{2 \times 1}$ via input weights $\mathbf{W}_i \in \mathbb{R}^{N \times 2}$ (see Fig 1b). This strategy effectively modifies the initial connectivity by addition of a low-rank component, allowing for more interpretable relations between the overall network connectivity and function [35, 37]. Note that a minimal rank, i.e. rank 1, perturbation could be used, but it is known to induce high correlations between emergent fixed points, thus restricting the potential range of emergent dynamics [35, 47]. Hence, to allow for a potentially wide range of solutions, we used a random connectivity plus rank 3 structure for the SPM task.

In our framework, optimization occurs only during the relevant temporal intervals in which these target signals are defined (Fig 1a), which we term a temporally restricted error kernel. When applying this kernel, the total error derived for a given trial is:

$$E(t) = \frac{1}{2} \int_{\mathcal{T}_d} e_d(t)^2 dt + \frac{1}{2} \int_{\mathcal{T}_r} e_o(t)^2 dt \tag{4}$$

where $\mathcal{T}_d$ and $\mathcal{T}_r$ are the temporal epochs associated with the two delay periods and response period, respectively (Fig 1a). Here,

$$e_d(t) = \|z_d(t) - f_d\|, \tag{5}$$

where $z_d = (z_{d_1}, z_{d_2})$ and $f_d = (f_{d_1}, f_{d_2})$ and

$$e_o(t) = \|z_o(t) - f_o\|. \tag{6}$$

Here, f_{d_1}, f_{d_2} and f_o are scalar real numbers chosen prior to optimization to represent the 2-dimensional stimulus and the trial outcome. During the delay intervals in particular, a low error thus implies that the neural activity linearly maps to a constant, invariant representation (i.e., $z_d \in \mathbb{R}^{2 \times 1}$). Activity during temporal epochs outside of the these periods do not impact the error. Optimization proceeds by modifying readout weights $\mathbf{W}_o$ and $\mathbf{W}_d$ to minimize these errors. Within the temporal error kernel, we deploy the FORCE method for parametric regression in RNNs. Here, $\mathbf{W}_o$ and $\mathbf{W}_d$ are updated using recursive least squares [33]. Briefly, to

reduce $e_o(t)$, we obtain

$$\mathbf{W}_o(t) = \mathbf{W}_o(t - \Delta t) - e_o(t)\mathbf{P}(t)\mathbf{r}(t)$$

$$\mathbf{P}(t) = \mathbf{P}(t - \Delta t) - \frac{\mathbf{P}(t - \Delta t)\mathbf{r}(t)\mathbf{r}^T(t)\mathbf{P}(t - \Delta t)}{1 + \mathbf{r}^T(t)\mathbf{P}(t - \Delta t)\mathbf{r}(t)}$$

(7)

where $\mathbf{P}(t)$ denotes the approximate estimate for the inverse of the correlation matrix of network activities with a regularization term

$$\mathbf{P}(t) = \int_{\mathcal{T}_r} \mathbf{r}(t)\mathbf{r}^T(t)dt + \alpha \mathbf{I}_N$$

(8)

where α is the regularization parameter and $\mathbf{I}_N$ the identity matrix. In the same manner, to reduce $e_d(t)$ we have

$$\mathbf{W}_d(t) = \mathbf{W}_d(t - \Delta t) - e_d(t)\mathbf{P}(t)\mathbf{r}(t).$$

(9)

Note that we update the associated inverse correlation matrices during training intervals $\mathcal{T}_d$ and $\mathcal{T}_r$ (shown in Fig 1a). In total, our training paradigm is a temporally regularized FORCE method that mitigates overfitting and in turn provides a potentially broader range of dynamical solutions to manifest. Indeed, it is known that optimizing RNNs using FORCE for a *sequential* trial-based task (here, a pattern association task with memory requirement) prevents the emergence of multiple fixed points in optimized networks, and thus can overly constrain the range of possible solution dynamics [47].

Dynamical systems analysis

The central theoretical question in our study pertains to analyzing the dynamics of our optimized networks. A first order question in this regard is to elucidate the landscape of attractors manifest in the network's vector field. In Eq (1), the origin is always a fixed point associated with the Jacobian matrix $\mathbf{J}$ (shifted by $-\mathbf{I}$). To study the stability of the origin, we can thus consider the eigenvalues (i.e. λ_J) of the connectivity matrix. It is well known that the eigenvalues of a random connectivity matrix $\mathbf{J}$ are distributed over a disk with radius g for $N \to \infty$ [46, 48]. Thus, the stability of origin varies with these parameters. For $g < 1$ the origin is asymptotically stable, while for $g > 1$ the origin is unstable, suggestive of potentially chaotic dynamics in the overall network. For the optimized networks with the rank 3 structure, the Jacobian matrix at the origin is $\mathbf{J}_T = \mathbf{J} + \mathbf{W}_f\mathbf{W}_o^T + \mathbf{W}_{fd}\mathbf{W}_d^T$, where we denote associated eigenvalues as λ_{J_T}.

Understanding the location and stability of fixed points away from the origin is harder to ascertain analytically. Hence, we rely on a number of numerical procedures to identify these points. To locate stable fixed points used for task computations, we arrest trials at relevant time moments, then forward simulate to ascertain the asymptotic behavior of the network. In one set of simulations, this forward simulation is carried out for trials arrested at the end of the first delay period. In a second set of simulations, it is carried out after trial conclusion. The forward simulation is carried out for ten times the nominal trial length, at which time we assume the network state is in a stationary regime, i.e., within an ϵ distance of either a stable fixed point or limit cycle. We can perform additional linearization about stable fixed points that are discovered numerically in this way. Here, the eigenvalue spectrum of Jacobian matrix, $\mathbf{Q}$, at these non-zero fixed points, denoted $\mathbf{x}^*$, is as follows

$$\mathbf{Q} = (\mathbf{J} + \mathbf{W}_f\mathbf{W}_o^T + \mathbf{W}_{fd}\mathbf{W}_d^T)\mathbf{R}'$$

(10)

where $\boldsymbol{R}'$ is a diagonal matrix with elements $\boldsymbol{R}'_{ij} = \delta_{ij}\mathbf{r}'_i$ with

$$\mathbf{r}' = 1 - \tanh^2(\mathbf{x}^*). \tag{11}$$

Note that if the states are largely saturated at a fixed point (as in the case of DFP encoding, Fig 4c), then the entires of $\mathbf{r}'$ are very small, which contracts the spectrum of $\boldsymbol{Q}$.

Network connectivity analysis

To link network connectivity properties with the identified mechanisms, we first sorted neurons based on the time they reach their peak activity for each task trial. Then, we sorted elements of the optimized connectivity matrix, i.e. $\boldsymbol{J}_T$, using the same ordered sequence of

neurons. We calculated the average pre-synaptic (or incoming) connections, i.e. $\bar{\boldsymbol{J}}_i = \frac{\sum_j^N J_{Tij}}{N}$, see Fig 4e. Moreover, we performed an analysis suggested by [36]; we computed the average of diagonal and successive off-diagonal elements of sorted connectivity matrix, i.e. $|i - j| = c, c \in \{0, \ldots, N\}$, see Fig 4f, which indicates the strength of reciprocal coupling between neurons as a function of the temporal distance between their peak activation.

Simulation parameters and specification

To perform dynamical system analysis for trained recurrent networks, we fixed the trained network parameters during subsequent testing and analysis. Throughout the paper we have referred to this setup as 'forward simulation' of the network dynamics. In the task, we set the stimulus interval to 100 time steps, delay intervals to 50 time steps (except for the extended delay experiments) and response interval to 50 time steps. From the sequential bivariate random process stimuli, we trained a variational auto encoder on the MNIST digits 1 and 0. We encoded the summation rule outcomes (i.e. 0, 1, 2) as 0.5, 1 and 1.5 (i.e., different values of f_o in Eq (6)), respectively for training the networks. We encoded the latent outputs f_{d_1} and f_{d_2} as the two dimensional mean vector for each digit representation, whenever that digit appeared prior to the delay period being optimized.

For all simulations the value of α is initialized to 1 and $\boldsymbol{P}$ was initialized to the identity matrix. The number of neurons was set as $N = 1000$ and elements of $\mathbf{W}_o$ and $\mathbf{W}_d$ are initialized to zero. Input weights were drawn randomly from zero mean Gaussian distribution with variance 0.02. We set the time step, dt, for Euler integration to 0.1. During training intervals, shown in Fig 1a, we updated weights every 2 time steps. For four different initialization seeds we considered all possible combination of feasible values for g, σ_f and sparsity (in Fig 6). The value of σ_f was chosen proportional to the size of network, i.e. $\sigma_f^2 \in \left\{ \frac{1}{1.5N}, \frac{1}{N}, \frac{1}{0.5N}, \frac{1}{0.1N}, \frac{1}{0.02N}, \frac{1}{0.005N}, \frac{1}{0.002N}, \frac{1}{0.001N} \right\}$. Training was terminated if the average root mean squared error between target and output was less that 0.01 for all trials.

For exemplar networks used in Figs 3a, 3c, 4a, 4b and 4d–4f, for type DFP: $g = 0.9$, $\sigma_f^2 = 1$ and sparsity is 0.2. For IFP: $g = 0.9$, $\sigma_f^2 = 0.05$ and sparsity is 0.1. For LC: $g = 0.9$, $\sigma_f^2 = 0.2$ and sparsity is 0.1. For Mix: $g = 0.9$, $\sigma_f^2 = 0.1$ and sparsity is 0.1. The initialization seed is the same for these 4 exemplar networks.

Supporting information

S1 Fig. RNN outputs after optimization for performing SPM task. Several concatenated trials of the task with outputs $z_d = (z_{d_1}, z_{d_2})$ and z_o are shown. Using the modified FORCE

method the network generates memory encodings during $\mathcal{T}_d$ and trial output during response intervals, $\mathcal{T}_r$ (shaded area).
(TIF)

S2 Fig. Flow chart. Categorization of dynamical mechanisms based on (i) the type of asymptotic attractor (either fixed point or limit cycle) and (ii) whether delay periods correspond to a fixed point.
(TIF)

S3 Fig. Neural activities and associated dynamical landscape for a single trial. For the same exemplar networks as in Fig 3c, but a different set of stimuli (i.e., here, two realizations of the same digit are presented), neural activity and associated low-dimensional (PCA) trajectories are plotted. Note that PCA components are obtained for each exemplar network individually. The trajectories are color coded using the same scheme as the color bar on the top. In DFP, the network creates four stable fixed points to solve the SPM task (the inset shows the area inside the circle). For the displayed trajectory, the network uses two fixed points (shown in yellow) to represent memory and trial output. In IFP, memory representation and trial output are encoded along the slow manifold of the single fixed point in the state space. In LC, the trajectories approach a stable limit cycle. For the mixed mechanism, both a stable fixed point and limit cycle can be seen.
(TIF)

S4 Fig. Neural trajectories of perturbed and salient trials. Plot shows how noise corrupts the salient trajectory for all four mechanisms (same trial and initial condition). PCs are exclusive to each network. In DFP, distracting noise places the trajectory in an erroneous basin of attraction and thus the network generates an incorrect response; in IFP noise pushes the trajectory away from the 'correct' slow manifold.
(TIF)

S5 Fig. VAE architecture. The latent space of VAE (i.e. z) is used to construct the input to the RNN model, where we have chosen a 2 dimensional latent space to represent x and y coordinates of MNIST digits dataset. For example, for x coordinate we obtain μ_x and σ_x corresponding to each specific digit and we can generate samples of that digit representation as temporal inputs (i.e. Gaussian process) to the network.
(TIF)

Acknowledgments

Author Contributions

Conceptualization: Elham Ghazizadeh, ShiNung Ching.

Formal analysis: Elham Ghazizadeh, ShiNung Ching.

Funding acquisition: ShiNung Ching.

Methodology: Elham Ghazizadeh, ShiNung Ching.

Project administration: ShiNung Ching.

Software: Elham Ghazizadeh.

References

1. Cowan N. What are the differences between long-term, short-term, and working memory?. *Progress In Brain Research*. 169 pp. 323–338 (2008) https://doi.org/10.1016/S0079-6123(07)00020-9 PMID: 18394484

2. Chaudhuri R. & Fiete I. Computational principles of memory. *Nature Neuroscience*. 19, 394 (2016) https://doi.org/10.1038/nn.4237 PMID: 26906506

3. Romo R., Brody C., Hernández A. & Lemus L. Neuronal correlates of parametric working memory in the prefrontal cortex. *Nature*. 399, 470–473 (1999) https://doi.org/10.1038/20939 PMID: 10365959

4. Funahashi S., Bruce C. & Goldman-Rakic P. Mnemonic coding of visual space in the monkey's dorso-lateral prefrontal cortex. *Journal Of Neurophysiology*. 61, 331–349 (1989) https://doi.org/10.1152/jn.1989.61.2.331 PMID: 2918358

5. Harvey C., Coen P. & Tank D. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*. 484, 62–68 (2012) https://doi.org/10.1038/nature10918 PMID: 22419153

6. Miller E., Lundqvist M. & Bastos A. Working Memory 2.0. *Neuron*. 100, 463–475 (2018) https://doi.org/10.1016/j.neuron.2018.09.023 PMID: 30359609

7. Riley M. & Constantinidis C. Role of prefrontal persistent activity in working memory. *Frontiers In Systems Neuroscience*. 9 pp. 181 (2016) https://doi.org/10.3389/fnsys.2015.00181 PMID: 26778980

8. Wang X. Synaptic reverberation underlying mnemonic persistent activity. *Trends In Neurosciences*. 24, 455–463 (2001) https://doi.org/10.1016/S0166-2236(00)01868-3 PMID: 11476885

9. Compte A., Brunel N., Goldman-Rakic P. & Wang X. Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. *Cerebral Cortex*. 10, 910–923 (2000) https://doi.org/10.1093/cercor/10.9.910 PMID: 10982751

10. Zhang X., Yi H., Bai W. & Tian X. Dynamic trajectory of multiple single-unit activity during working memory task in rats. *Frontiers In Computational Neuroscience*. 9 pp. 117 (2015) https://doi.org/10.3389/fncom.2015.00117 PMID: 26441626

11. Murray J., Bernacchia A., Roy N., Constantinidis C., Romo R. & Wang X. Stable population coding for working memory coexists with heterogeneous neural dynamics in prefrontal cortex. *Proceedings Of The National Academy Of Sciences*. 114, 394–399 (2017) https://doi.org/10.1073/pnas.1619449114 PMID: 28028221

12. Barak O., Sussillo D., Romo R., Tsodyks M. & Abbott L. From fixed points to chaos: three models of delayed discrimination. *Progress In Neurobiology*. 103 pp. 214–222 (2013) https://doi.org/10.1016/j.pneurobio.2013.02.002 PMID: 23438479

13. Chaisangmongkon W., Swaminathan S., Freedman D. & Wang X. Computing by robust transience: how the fronto-parietal network performs sequential, category-based decisions. *Neuron*. 93, 1504–1517 (2017) https://doi.org/10.1016/j.neuron.2017.03.002 PMID: 28334612

14. Zylberberg J. & Strowbridge B. Mechanisms of persistent activity in cortical circuits: possible neural substrates for working memory. *Annual Review Of Neuroscience*. 40 (2017) https://doi.org/10.1146/annurev-neuro-070815-014006 PMID: 28772102

15. Barak O. & Tsodyks M. Working models of working memory. *Current Opinion In Neurobiology*. 25 pp. 20–24 (2014) https://doi.org/10.1016/j.conb.2013.10.008 PMID: 24709596

16. Deco G. & Rolls E. Attention and working memory: a dynamical model of neuronal activity in the prefrontal cortex. *European Journal Of Neuroscience*. 18, 2374–2390 (2003) https://doi.org/10.1046/j.1460-9568.2003.02956.x PMID: 14622200

17. Druckmann S. & Chklovskii D. Neuronal circuits underlying persistent representations despite time varying activity. *Current Biology*. 22, 2095–2103 (2012) https://doi.org/10.1016/j.cub.2012.08.058 PMID: 23084992

18. Spaak E., Watanabe K., Funahashi S. & Stokes M. Stable and dynamic coding for working memory in primate prefrontal cortex. *Journal Of Neuroscience*. 37, 6503–6516 (2017) https://doi.org/10.1523/JNEUROSCI.3364-16.2017 PMID: 28559375

19. Bondanelli G. & Ostojic S. Coding with transient trajectories in recurrent neural networks. *PLoS Computational Biology*. 16, e1007655 (2020) https://doi.org/10.1371/journal.pcbi.1007655 PMID: 32053594

20. Park J., Bae J., Kim J. & Jung M. Dynamically changing neuronal activity supporting working memory for predictable and unpredictable durations. *Scientific Reports*. 9, 1–10 (2019) https://doi.org/10.1038/s41598-019-52017-8 PMID: 31664169

21. Bauer R. & Fuster J. Delayed-matching and delayed-response deficit from cooling dorsolateral prefrontal cortex in monkeys. *Journal Of Comparative And Physiological Psychology*. 90, 293 (1976) https://doi.org/10.1037/h0087996 PMID: 819472

22. Hoerzer G., Legenstein R. & Maass W. Emergence of complex computational structures from chaotic neural networks through reward-modulated Hebbian learning. *Cerebral Cortex*. 24, 677–690 (2014) https://doi.org/10.1093/cercor/bhs348 PMID: 23146969

23. Enel P., Procyk E., Quilodran R. & Dominey P. Reservoir computing properties of neural dynamics in prefrontal cortex. *PLoS Computational Biology*. 12, e1004967 (2016) https://doi.org/10.1371/journal.pcbi.1004967 PMID: 27286251

24. Pascanu R. & Jaeger H. A neurodynamical model for working memory. *Neural Networks*. 24, 199–207 (2011) https://doi.org/10.1016/j.neunet.2010.10.003 PMID: 21036537

25. Maheswaranathan, N., Williams, A., Golub, M., Ganguli, S. & Sussillo, D. Universality and individuality in neural dynamics across large populations of recurrent networks. *Advances In Neural Information Processing Systems*. pp. 15629–15641 (2019)

26. Song H., Yang G. & Wang X. Training excitatory-inhibitory recurrent neural networks for cognitive tasks: a simple and flexible framework. *PLoS Computational Biology*. 12, e1004792 (2016) https://doi.org/10.1371/journal.pcbi.1004792 PMID: 26928718

27. Nachstedt T. & Tetzlaff C. Working memory requires a combination of transient and attractor-dominated dynamics to process unreliably timed inputs. *Scientific Reports*. 7, 1–14 (2017) https://doi.org/10.1038/s41598-017-02471-z PMID: 28559576

28. Enel P., Wallis J. & Rich E. Stable and dynamic representations of value in the prefrontal cortex. *Elife*. 9 pp. e54313 (2020) https://doi.org/10.7554/eLife.54313 PMID: 32628108

29. Cavanagh S., Towers J., Wallis J., Hunt L. & Kennerley S. Reconciling persistent and dynamic hypotheses of working memory coding in prefrontal cortex. *Nature Communications*. 9, 1–16 (2018) https://doi.org/10.1038/s41467-018-05873-3 PMID: 30158519

30. Orhan A. & Ma W. A diverse range of factors affect the nature of neural representations underlying short-term memory. *Nature Neuroscience*. 22, 275–283 (2019) https://doi.org/10.1038/s41593-019-0348-9 PMID: 30664767

31. Sussillo D., Churchland M., Kaufman M. & Shenoy K. A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*. 18, 1025–1033 (2015) https://doi.org/10.1038/nn.4042 PMID: 26075643

32. Yang G., Joglekar M., Song H., Newsome W. & Wang X. Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*. 22, 297–306 (2019) https://doi.org/10.1038/s41593-018-0310-2 PMID: 30643294

33. Sussillo D. & Abbott L. Generating coherent patterns of activity from chaotic neural networks. *Neuron*. 63, 544–557 (2009) https://doi.org/10.1016/j.neuron.2009.07.018 PMID: 19709635

34. Yang, G. & Wang, X. Artificial neural networks for neuroscientists: A primer. *ArXiv Preprint ArXiv:2006.01001*. (2020)

35. Schuessler F., Dubreuil A., Mastrogiuseppe F., Ostojic S. & Barak O. Dynamics of random recurrent networks with correlated low-rank structure. *Physical Review Research*. 2, 013111 (2020) https://doi.org/10.1103/PhysRevResearch.2.013111

36. Rajan K., Harvey C. & Tank D. Recurrent network models of sequence generation and memory. *Neuron*. 90, 128–142 (2016) https://doi.org/10.1016/j.neuron.2016.02.009 PMID: 26971945

37. Mastrogiuseppe F. & Ostojic S. Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*. 99, 609–623 (2018) https://doi.org/10.1016/j.neuron.2018.07.003 PMID: 30057201

38. Hopfield J. Neural networks and physical systems with emergent collective computational abilities. *Proceedings Of The National Academy Of Sciences*. 79, 2554–2558 (1982) https://doi.org/10.1073/pnas.79.8.2554 PMID: 6953413

39. Rubin D., Van Hooser S. & Miller K. The stabilized supralinear network: a unifying circuit motif underlying multi-input integration in sensory cortex. *Neuron*. 85, 402–417 (2015) https://doi.org/10.1016/j.neuron.2014.12.026 PMID: 25611511

40. Wang J., Kurth-Nelson Z., Kumaran D., Tirumala D., Soyer H., Leibo J., et al. Prefrontal cortex as a meta-reinforcement learning system. *Nature Neuroscience*. 21, 860–868 (2018) https://doi.org/10.1038/s41593-018-0147-8 PMID: 29760527

41. Seeholzer A., Deger M. & Gerstner W. Stability of working memory in continuous attractor networks under the control of short-term plasticity. *PLoS Computational Biology*. 15, e1006928 (2019) https://doi.org/10.1371/journal.pcbi.1006928 PMID: 31002672

42. Manohar S., Zokaei N., Fallon S., Vogels T. & Husain M. Neural mechanisms of attending to items in working memory. *Neuroscience & Biobehavioral Reviews*. 101 pp. 1–12 (2019) https://doi.org/10.1016/j.neubiorev.2019.03.017 PMID: 30922977

43. Murphy A., Bertolero M., Papadopoulos L., Lydon-Staley D. & Bassett D. Multimodal network dynamics underpinning working memory. *Nature Communications*. 11, 1–13 (2020) https://doi.org/10.1038/s41467-020-15541-0 PMID: 32541774

44. Curtis C. & D'Esposito M. Persistent activity in the prefrontal cortex during working memory. *Trends In Cognitive Sciences*. 7, 415–423 (2003) https://doi.org/10.1016/S1364-6613(03)00197-9 PMID: 12963473

45. Sakai K. Task set and prefrontal cortex. *Annu. Rev. Neurosci*. 31 pp. 219–245 (2008) https://doi.org/10.1146/annurev.neuro.31.060407.125642 PMID: 18558854

46. Sompolinsky H., Crisanti A. & Sommers H. Chaos in random neural networks. *Physical Review Letters*. 61, 259 (1988) https://doi.org/10.1103/PhysRevLett.61.259 PMID: 10039285

47. Beer C. & Barak O. One step back, two steps forward: interference and learning in recurrent neural networks. *Neural Computation*. 31, 1985–2003 (2019) https://doi.org/10.1162/neco_a_01222 PMID: 31393826

48. Rajan K. & Abbott L. Eigenvalue spectra of random matrices for neural networks. *Physical Review Letters*. 97, 188104 (2006) https://doi.org/10.1103/PhysRevLett.97.188104 PMID: 17155583