bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Trained recurrent neural networks develop phase-locked limit cycles in a working memory task

Matthijs Pals[1,2*], Jakob H Macke[1,2,3], Omri Barak[4,5*]

1 Machine Learning in Science, Excellence Cluster Machine Learning, University of Tübingen, Tübingen, Germany
2 Tübingen AI Center, University of Tübingen, Tübingen, Germany
3 Department Empirical Inference, Max Planck Institute for Intelligent Systems, Tübingen, Germany
4 Rappaport Faculty of Medicine Technion, Israel Institute of Technology, Haifa, Israel
5 Network Biology Research Laboratory, Israel Institute of Technology, Haifa, Israel

* Corresponding authors:
matthijs.pals@uni-tuebingen.de (MP)
omri.barak@gmail.com (OB)

Abstract

Neural oscillations are ubiquitously observed in many brain areas. One proposed functional role of these oscillations is that they serve as an internal clock, or ‘frame of reference’. Information can be encoded by the timing of neural activity relative to the *phase* of such oscillations. In line with this hypothesis, there have been multiple empirical observations of such *phase codes* in the brain. Here we ask: What kind of neural dynamics support phase coding of information with neural oscillations? We tackled this question by analyzing recurrent neural networks (RNNs) that were trained on a working memory task. The networks were given access to an external reference oscillation and tasked to produce an oscillation, such that the phase difference between the reference and output oscillation maintains the identity of transient stimuli. We found that networks converged to stable oscillatory dynamics. Reverse engineering these networks revealed that each phase-coded memory corresponds to a separate limit cycle attractor. We characterized how the stability of the attractor dynamics depends on both reference oscillation amplitude and frequency, properties that can be experimentally observed. To understand the connectivity structures that underlie these dynamics, we showed that trained networks can be described as two phase-coupled oscillators. Using this insight, we condensed our trained networks to a reduced model consisting of two functional modules: One that generates an oscillation and one that implements a coupling function between the internal oscillation and external reference. In summary, by reverse engineering the dynamics and connectivity of trained RNNs, we propose a mechanism by which neural networks can harness reference oscillations for working memory. Specifically, we propose that a phase-coding network generates autonomous oscillations which it couples to an external reference oscillation in a multi-stable fashion.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Author summary

Many of our actions are rhythmic—walking, breathing, digesting and more. It is not surprising that neural activity can have a strong oscillatory component. Indeed, such brain waves are common, and can even be measured using EEG from the scalp. Perhaps less obvious is the presence of such oscillations during non-rhythmic behavior—such as memory maintenance and other cognitive functions. Reports of these cognitive oscillations have accumulated over the years, and various theories were raised regarding their origin and utilization. In particular, oscillations have been proposed to serve as a clock signal that can be used for temporal-, or phase-coding of information in working memory. Here, we studied the dynamical systems underlying this kind of coding, by using trained artificial neural networks as hypothesis generators. We trained recurrent neural networks to perform a working memory task, while giving them access to a reference oscillation. We were then able to reverse engineer the learned dynamics of the networks. Our analysis revealed that phase coded memories correspond to stable attractors in the dynamical landscape of the model. These attractors arose from the coupling of the external reference oscillation with oscillations generated internally by the network.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Introduction

Rhythmic neural activity is an abundant phenomenon in nervous systems. Neural oscillations naturally underlie behavior with an observable oscillatory component, such as walking and digesting [1, 2]. Oscillating neural activity is also widely observed in brain regions implicated in higher cognitive functions, where there is no obvious correlate to oscillatory behavior [3]. Such rhythmic neural activity has been suggested to support short-term memory maintenance (among other functions), by serving as an internal clock for the brain [4–9]. This makes *phase-coding* possible: Information can be encoded by spikes that are systematically timed with respect to the phase of ongoing oscillations. Empirical observations of a *phase code* in the brain were first described in the hippocampus of moving rats [10]. Since then, phase coding has also been associated with the representation of discrete object categories [11–13] for short-term maintenance of stimuli [7, 14–16] and goals [17]. Such observations have been reported in a wide range of brain regions, including the human medial temporal lobe as well as primate prefrontal and sensory cortices.

Oscillations are a dynamic phenomenon, and it is therefore a natural question to ask: How do ongoing oscillations in the brain interact with the neural dynamics that support cognitive functions? Specifically, we seek to characterize dynamical systems that could underlie working memory relying on phase coding with neural oscillations. We do so by assuming that cognitive functions can be described by a low-dimensional dynamical system, implemented through populations of neurons (computation through dynamics) [18–23], in line with empirical observations [24, 25].

We make use of artificial recurrent neural networks (RNNs). RNNs are universal dynamical systems, in the sense that they can approximate finite time trajectories of any dynamical system [26, 27]. We are thus able to use these networks as hypothesis generators — first training them on a cognitive task, and then reverse engineering the resulting networks [18, 22, 28–35].

Concretely, we provide RNNs with a reference oscillation, as well as transient stimuli as input. We train the networks to produce an oscillation whose relative phase maintains stimulus identity (Fig. 1). We find two different solutions, of which one corresponds to network units coding for information by their phase, and one solution in which units code with their average firing rates (Fig. 2). We show that phase-coded memories correspond to stable limit cycles and demonstrate how empirically observable quantities control the stability of these attractors (Fig. 3). Having detailed the dynamics, we study the connectivity of our trained RNNs (Fig. 4). Finally, we show that the system can be well approximated by two coupled oscillators. Based on these analysis, we propose that phase-coded memories reside in stable limit cycles, resulting from the coupling of an oscillation generated by the recurrent network to an external reference oscillation.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Results

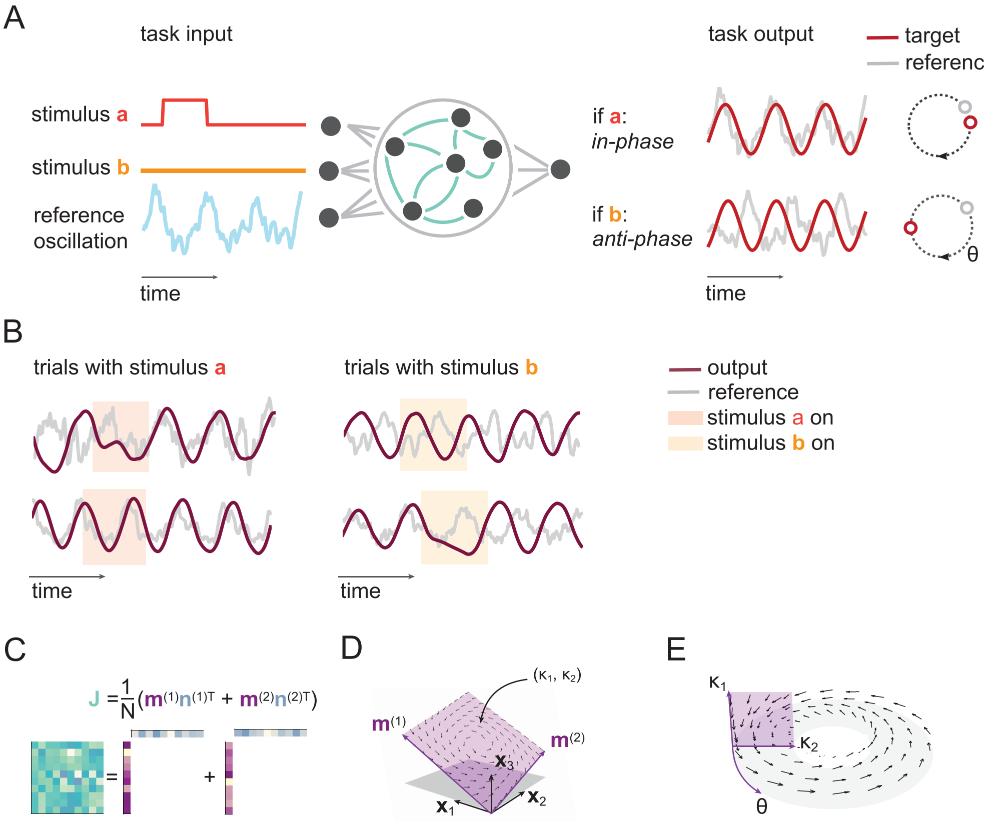

Fig 1. Trained RNNs encode stimuli in oscillation phase. A) RNNs receive transient stimuli as input, along with a reference oscillation. Networks are trained to produce an oscillation, such that the phase of the produced oscillation (relative to the reference oscillation), maintains the identity of transient stimuli. B) Example output of trained networks. Transient presentation of stimulus a, results in an in-phase output oscillation (left), regardless of the initial phase (top or bottom). Similarly, the b stimulus results in an anti-phase oscillation, again irrespective of its initial phase (right). C) To obtain a tractable model, we apply a low-rank constraint to the recurrent weight matrix of the RNN, i.e., we require that the weight matrix can be written as the outer product of two sets of vectors $\mathbf{m}^{(1)}, \mathbf{m}^{(2)}$ and $\mathbf{n}^{(1)}, \mathbf{n}^{(2)}$. D) Low-rank connectivity leads to low dimensional dynamics. In the absence of any input, the recurrent dynamics, described by coordinates κ_1, κ_2, lie in a linear subspace spanned by $\mathbf{m}^{(1)}$ and $\mathbf{m}^{(2)}$ (purple). E) When probing the model with sinusoidal oscillations, we can rewrite the system as a dynamical system in a three-dimensional phase space, where the additional axis, θ, is the phase of the input oscillation. We can visualize this phase space as a toroid, embedded in a 3 dimensional space, such that θ is the horizontal circle, and the vertical circle lies in the κ_1, κ_2 plane.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Tractable, oscillating recurrent neural networks perform a working memory task

In order to study the dynamics of phase coding during working memory, we defined a task in which an RNN receives transient stimuli, and has to encode their identity using the relative phase of oscillations (Fig. 1A). The network consists of N units, with activation $\mathbf{x}(t) \in \mathcal{R}^N$, recurrently connected via a connectivity matrix $\mathbf{J} \in \mathcal{R}^{N\times N}$, and receiving external input $\mathbf{u}(t) \in \mathcal{R}^3$,

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x}(t) + \mathbf{J}\tanh(\mathbf{x}(t)) + \mathbf{I}\mathbf{u}(t) + \boldsymbol{\xi}(t), \tag{1}$$

where τ represents the time constant of the units, tanh is an elementwise non-linearity, $\mathbf{I} \in \mathcal{R}^{N\times 3}$ represents the input weights, and $\boldsymbol{\xi}(t) \in R^N$ independent noise for each unit.

The RNN received an oscillatory reference input, for which we used filtered rat CA1 local field potentials (LFPs) [36, 37]. Additionally, during a given trial, one of two stimuli a or b, was transiently presented to the network at a randomized onset-time, with amplitude s_a or s_b, respectively. Networks were trained with backpropagation through time (Fig. S1). The task was to produce an oscillation, that is either in-phase with respect to the reference signal, following stimulus a, or anti-phase, following stimulus b. Networks were able to successfully learn the task (Fig. 1B).

As we were interested in reverse-engineering the networks, we made two simplifications. After training, we replaced the LFP reference signal with a pure sine wave with phase θ. During training, we chose a constraint on the connectivity that reduces the complexity of our analysis while still allowing for expressive networks. Specifically, we constrained the recurrent weight matrix to be of rank two, by decomposing it as an outer product of two pairs of vectors.(Fig. 1C; see Fig. S2 for unconstrained networks) [27, 32, 38, 39],

$$\mathbf{J} = \frac{1}{N}(\mathbf{m}^{(1)}\mathbf{n}^{(1)\mathsf{T}} + \mathbf{m}^{(2)}\mathbf{n}^{(2)\mathsf{T}}). \tag{2}$$

By constraining the weight matrix, we directly constrain the dynamics. Specifically, the projections of network activity on the two vectors $\mathbf{m}^{(1)}, \mathbf{m}^{(2)}$, which we term κ_1, κ_2 respectively, are sufficient to describe the network dynamics in the absence of inputs (Fig. 1D).

In the presence of sinusoidal input, the κs are not sufficient to describe the dynamics, and we also need to know the current phase θ of the reference oscillation. These three numbers constitute the complete phase space $\mathcal{M}$ for our dynamical system (Fig. 1E),

$$\mathcal{M} = \{(\kappa_1, \kappa_2, \theta) \in \mathcal{R}^2 \times \mathcal{S}^1\}.$$

Phase-coded memories correspond to limit cycle attractors

We reverse-engineered the dynamics of our trained networks in order to understand how they solve the task [35]. Depending on the training setup, we found that networks converge to one of two solutions,characterized by their activity following the transient stimuli: In the first solution, individual network units code for stimuli by using their phase of oscillation relative to the reference, as illustrated by rate traces of example units (Fig. 2A), as well as statistics for all units (Fig. S3). In κ space, the population activity corresponds to two cycles that roughly lie on the same area, but have a different phase relation to the reference oscillation (Fig. 2B). In the full phase space $\mathcal{M}$ introduced above, this solution corresponds to the network activity residing in one of two linked cycles (Fig. 2C).

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

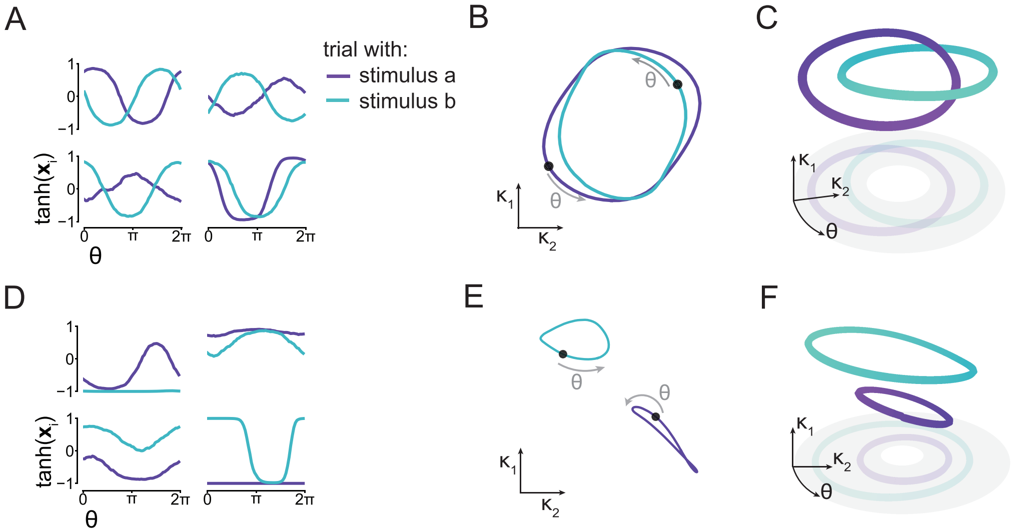

Fig 2. Both phase-coding and rate-coding can solve the working memory task. We found two qualitatively different solutions. A) In the first solution (top row), we find that single unit activity codes for stimulus information by relative phase: We plot the rates $\tanh(\mathbf{x}_i)$ of 4 units i, as a function of the reference oscillation phase θ. We find that single unit activity oscillates, with the phase relative to the reference oscillation depending on stimulus identity (colors). B) Projecting $\mathbf{x}$ in the κ_1, κ_2 plane reveals that population activity lies on overlapping cycles in this plane. Here, the black dots denote $\theta = 0$. C) In the full phase space $\mathcal{M}$, the trajectories are non-overlapping, but the cycles are linked. D) In the second solution (bottom row), we find that stimulus identity does not modulate the phase of single units, but rather their mean activity. E) This rate-code corresponds to two cycles separated in the κ_1, κ_2 plane. F) These cycles are also separated in the full phase space.

In the second solution, single units code stimulus identity by their average activation (Fig. 2D, Fig. S3). Due to saturation of the nonlinearity, this effectively means that a different set of units determines the phase of the output oscillation after either stimulus. In κ space, activity lies on two non-overlapping cycles (Fig. 2E), which are likewise completely separated in the full phase space (Fig. 2F). Given the similarities of the second solution to previous work on fixed point dynamics in RNNs [27, 28, 32, 40], we focus here on the analysis of the 'phase-coding' solution (Fig. 2A-C).

Single trials have a limited duration, and hence the cycles we observed might arise either from a transient dynamical phenomenon or from stable attractors. These would lead to different experimental observations of residual dynamics (trial-to-trial variability in neural population responses [41]), and responses to perturbations [34]. To study stability, we used discrete-time iterative maps, or Poincaré maps [42, 43]. Given a cross-section $Q = \{(\kappa_1, \kappa_2, \theta) : \mod \theta = 2\pi\}$ through the phase space, one can follow trajectories as they go through Q multiple times. We define the iterative map from Q to itself (Fig. 3A),

$$\boldsymbol{\kappa}_{c+1} = \mathbf{P}(\boldsymbol{\kappa}_c),$$

where $\boldsymbol{\kappa_c} = [\kappa_{1,c}, \kappa_{2,c}]^\mathsf{T} \in S$ corresponds to the c'th intersection. We found that trajectories starting from many initial conditions quickly converged to one of two fixed points in Q (Fig. 3A), corresponding to the cycles observed during the working memory task (Fig. 2C). To confirm their stability, we performed linear stability analysis by calculating the Floquet multipliers (λ), i.e., the eigenvalues of the linearized Poincaré map. Limit cycles are stable if these have a magnitude less than one, i.e., $|\lambda| < 1$.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

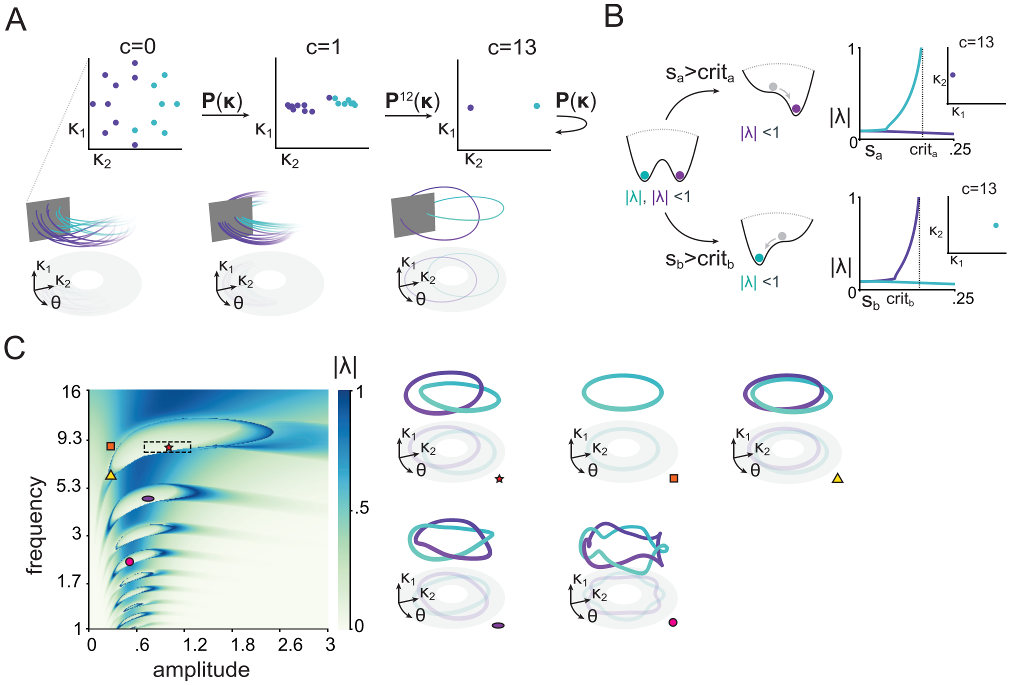

Fig 3. Input controls the stability of the attractors in which phase-coded memories reside. A) Poincaré maps illustrate that the previously observed cycles correspond to attractive limit cycles. Sixteen trajectories with different initial conditions are shown after 1 and 13 cycles. Trajectories were colored by the cycle they end up in. B) Linear stability analysis of the Poincaré maps shows that stimuli of sufficient magnitude ($s_i > crit_i$) lead to a bifurcation, such that only one of two limit cycles remains stable. Left: cartoon of the bistable dynamics (without stimuli) and bifurcation during stimulus presentation illustrated as a potential well. Right: Floquet multiplier norm (a measure of stability) as a function of stimulus amplitude. Insets show the Poincaré map after 13 iterations with the stimulus presented at amplitude $s_i = 1$. C) Stability analysis for a range of amplitudes and frequencies of the reference oscillation (quantities that naturally vary in the brain). The box with dashed lines indicates the parameters seen during training. The boundaries of the box are 5'th and 95'th percentiles of the amplitudes and frequencies used during training trials. The 'islands' correspond to regions where bistable dynamics persist. If the reference oscillation and amplitude are within such a bistable region (e.g. the red star), there are two stable cycles, and the model can maintain memory of a stimulus. For lower amplitudes (orange square and yellow triangle), the model only retains bistability if the frequency is also lower (yellow triangle). The additional 'islands' correspond to regions with bistable $m : n$ phase locking, where the reference oscillation is an integer divisor of the oscillation generated by the RNN (e.g. purple oval, bistable $1 : 2$ phase locking; pink circle, bistable $1 : 4$ phaselocking). Trajectories on the right correspond to the different markers in the parameter space on the left.

This analysis allows us to study how the two inputs direct activity to the corresponding limit cycle. Without inputs, both cycles are stable (Fig. 3B, leftmost cartoon). We can then tonically provide input corresponding to a scaled version of one of the stimuli, and recalculate the maximal Floquet multiplier for the resulting limit cycle. We found that this procedure gradually destabilizes the other limit cycle, so that eventually only the limit cycle corresponding to the correct input remains (Fig. 3B,

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

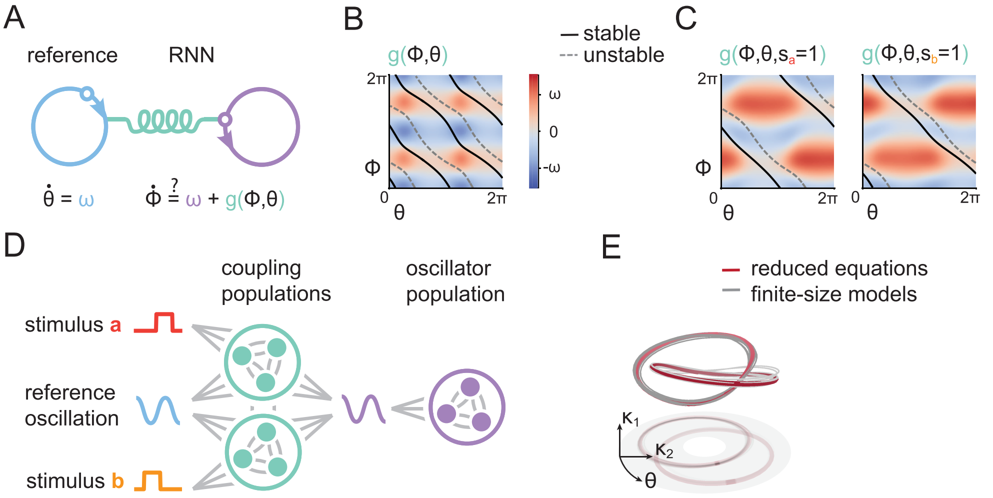

Fig 4. Phase-coding RNNs are oscillators. A) We hypothesized that the model functions as two coupled oscillators, where one represents the external reference oscillation phase and one the RNNs' internal oscillation phase. B) We extracted the coupling functions from our trained RNNs. This coupling function induces bistable dynamics when it couples two oscillators, as apparent from the superimposed stable and unstable trajectories. C) Input stimuli transiently modify the coupling function, resulting in the bifurcation, previously observed in Fig. 3B. D) To formalize the coupled oscillator description, we created a reduced model where weights are drawn from a mixture of Gaussians. This model consists of a population that generates oscillations, and two populations that together implement the coupling function between internal and external oscillations. E) Simulating a reduced set of equations that describe the idealized dynamics of RNNs with connectivity in terms of a Gaussian mixture distribution, as well as 10 finite-size models with weights sampled from this distribution, all result in trajectories similar to our original system, validating our reduced description.

right cartoons). Thus, for both stimuli, once a sufficient amplitude is crossed ($s_i > crit_i$; Fig. 3B), a bifurcation occurs and only one limit cycle remains stable. When the transient stimulus is removed, both limit cycles are stable again, but the network has already been directed to one of them.

As realistic neural oscillations exhibit drifts in both frequency and amplitude, we next assessed the robustness of our network to variations of the reference oscillation, going substantially beyond the range encountered during training. This analysis also makes testable predictions, as changes in oscillation amplitude and frequency can be experimentally observed, and potentially controlled [44].

We calculated the norm of the maximum Floquet multiplier for varying frequencies and amplitudes of the reference oscillation (Fig. 3C), in order to determine when bifurcations occur (i.e., maximum multiplier norm exceeding one). The resulting diagram allows us to draw two conclusions: First, there are multiple regions with two stable oscillations, but with an $m : n$ phase coupling, i.e., where the internal oscillation frequency is an integer multiplier of the reference frequency. This kind of cross-frequency phase-phase coupling is an integral part of previously proposed theories of phase coding [9, 45] (however, interpretation of experimental observations can be challenging [46]). Second, based on the shape and location of the bistable regions (the 'islands' in Fig. 3C), we predict that being able to phase-code stimuli is possible for reference oscillations with low frequency, only if the amplitude is also low.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Low-rank RNNs as phase-coupled oscillators

Having described the dynamics, we next aimed to understand the population structure that gives rise to these dynamics. To proceed, we observe that, following the stimulus, the internal dynamics of the RNN quickly converge to a trajectory in the (κ_1, κ_2) plane, which can be approximately described by its phase (Fig. 2B $\phi = \arctan(\frac{\kappa_1}{\kappa_2})$). Using this insight, we can rewrite our model as two phase-coupled oscillators (Fig. 4A): one that describes the external reference oscillation phase (θ) and one that characterizes the internal RNN phase (ϕ). The dynamics of these oscillators are then determined by the oscillation frequency ω, and a coupling function $g(\phi, \theta)$ [47]:

$$\begin{aligned} \dot{\theta} &= \omega, \\ \dot{\phi} &= \omega + g(\phi, \theta). \end{aligned} \tag{3}$$

To extract the coupling function g from our trained networks, we rewrote the model equations in polar coordinates, and approximated the radius in κ-space as constant (Fig. 4B, Methods). We then simulated the oscillators of equation 3 using this function, and observed two stable trajectories (superimposed lines in Fig. 4B), showing that the coupling function is sufficient to induce bistable dynamics. We found that the convergent trajectories are close to those of the RNN projected to the same phase space (θ, ϕ), verifying that our approximate description captures the dynamics of the full RNN (Fig. S4). Furthermore, the observed stimulus-induced dynamics (Fig. 3C) can now be explained by the effect of input on the coupling function. In the presence of input, the coupling function only admits a single stable trajectory, resulting from one stable and one unstable trajectory colliding and annihilating in a saddle-node bifurcation (Fig. 4C).

Finally, we wanted to know how the rank-two connectivity leads to these coupled oscillators. To this end, we approximated the weights in the connectivity vectors of the RNN using a mixture of Gaussians [27, 32, 38]. Fitting a mixture of Gaussians to the connectivity of trained networks, and sampling weights from this mixture, as in previous work [32], did not reliably lead to functioning networks (Fig. S5A). We were, however, able to manually design a reduced model, with weights drawn from a mixture of Gaussians, based on the reverse-engineered dynamics of trained networks, as well as the structure in their connectivity as revealed through the clustering analysis (Fig. S5B). The reduced model consisted of three mixture components (or subpopulations; Fig. 4D, Fig. S6). One component is not connected to the reference oscillation and autonomously generates its own oscillation. The other two components implement the required coupling function. The two coupling components differ only in their connectivity with the input; one population saturates (i.e. is inhibited) by stimulus a, and one by stimulus b.

The description in terms of a mixture of Gaussians enables us to derive a reduced set of three equations, which describe the dynamics of the RNN, in the limit of infinite units (Reduced models). We confirmed that finite-size networks are appropriately described by this reduced description, by simulating trajectories of 10 RNNs (with N=4096 units) sampled from the deduced connectivity (Fig. 4E). Thus, we show that a sufficient connectivity for working memory through phase-coding entails two modules: an oscillator and a coupling function that induces bistable dynamics.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Discussion

In this study, we raised a hypothesis about a potential dynamical mechanism underlying phase-coding with neural oscillations. Namely, that phase-coding in recurrent networks can be implemented through multi-stable coupling of two oscillations, one internal and one external to the network. We arrived at the hypothesis by training RNNs on a working memory task, while supplying them with oscillatory reference input. The networks had to encode transient stimuli by producing an oscillation with a persistent phase shift with respect to the reference input.

Through reverse engineering the dynamics of trained RNNs, we found that phase-coded memories correspond to stable periodic attractors. These materialized in our models as linked cycles in phase space. The presence of attractive oscillatory dynamics, as opposed to marginally stable or transient trajectories, can be detected by analyzing residual dynamics from data [41] or through perturbation studies [34]. We showed how LFP frequency and amplitude jointly control the stability of the attractors. As LFP oscillation frequency and amplitude vary naturally in the brain, and can potentially be steered [44], this relationship could be probed directly from neural data.

Beyond characterizing the dynamics, we also revealed the effective underlying connectivity. We showed that our trained RNNs are analogous to coupled oscillators and that a sufficient connectivity for phase coding entails two modules: one generating an oscillation and one implementing a coupling function between this oscillation and an external reference one. Two independently generated oscillations that couple during memory are found in the medial temporal lobe (i.e. theta and gamma oscillations) [3, 5, 15]. Our model would predict that coupling functions extracted from neural data [47, 48], recorded from subjects performing working memory tasks, should have a structure that induces multi-stability.

We used trained recurrent neural networks [18, 20, 22]. RNNs can be trained to reproduce both neural activity and behavior [16, 21, 28, 31, 34]. The resulting networks can be understood in terms of their dynamical objects, i.e. the set of attractors [35, 49], and in terms of the trajectories [33, 50] flowing between them. In particular, recurrent models that implement discrete memory, as in this study, often have a separate *static* attractor, or fixed point, for each memory [27, 29, 51, 52], although oscillatory models of memory have also been proposed [16, 40, 53]. Here we complement previous work by showing that stable *dynamic* attractors naturally emerge when tasking networks to store memories in the relative phase of oscillations.

Our findings support the notion that rhythmic neural activity can play a supporting role in cognitive phenomena. Brain waves during working memory are widely observed in neural systems of rats, primates, and birds [54]. Phase-coding of information relative to theta oscillations has been proposed to be a general coding scheme in the brain [5]. Here, we focused on the coding of two distinct stimuli in phase, which suffices to highlight the dynamics underlying the coding of discrete pieces of information. Both our model and findings can also be extended to coding for more than two items (Fig. S7). Phase-coding has also been observed during memory of sequences of information [16], as well as for coding of position [10]. Linking these findings to our model requires further investigation. We note that phase precession, in the sense of a continuous variable (e.g., position in a place field) changing the phase to which a group of neurons locks could be explained by our model through input translating the coupling function, as is tentatively shown in Fig. S8.

In summary, we used RNNs as trainable dynamical systems to form a hypothesis on how oscillations can be harnessed for neural computation. We proposed that phase-coded memories in recurrent networks reside in stable limit cycles resulting from the coupling of internal and external oscillations.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Models and methods

Code availability

All code for obtaining the results and generating the figures is available at: https://github.com/mackelab/phase-limit-cycle-RNNs.

Training RNNs on a phase-coding task

Model definition

As stated in Eq 1, we used a continuous time RNN with N units [30],

$$\tau \frac{d\mathbf{x}}{dt} = -\mathbf{x}(t) + \mathbf{J}\tanh(\mathbf{x}(t)) + \mathbf{I}^{(osc)}u(t) + \mathbf{I}^{(s_b)}s_b(t) + \mathbf{I}^{(s_a)}s_a(t) + \boldsymbol{\xi}(t). \tag{4}$$

The input weights $\mathbf{I}$ of Eq 1 are split into three vectors in $\mathcal{R}^N$: The oscillatory reference input $u(t)$ with input weights $\mathbf{I}^{(osc)}$ and transient stimuli $s_a(t)$ and $s_b(t)$ with weights $\mathbf{I}^{(s_a)}$ and $\mathbf{I}^{(s_b)}$, respectively. To obtain networks with tractable dynamics, we implemented a low-rank constraint according to Eq. 2. Trained networks consisted of $N = 512$ units, with $\tau = 20$ ms.

Task

We defined a task in which stimulus identity is to be encoded in the phase of an oscillation. During each trial, of duration $T = 0.8s$, a reference oscillation $u(t)$ with random initial phase θ was provided to the RNN. Trials started with an initial period, with duration drawn uniformly from $[0.125s, 0.25s]$. After the initial period, either stimulus a, $s_a(t)$, or stimulus b, $s_b(t)$, was shown, consisting of a 'pulse' with a constant amplitude of 1 (see Fig. 1A). The duration of stimuli was drawn uniformly from $[0.125s, 0.175s]$. For the remainder of the trial, the target $\hat{r}(t)$ was either $\sin(\theta(t) - 0.2\pi)$, when stimulus a was shown ('in-phase' trial), or else $\sin(\theta(t) - 1.2\pi)$ when stimulus b was shown ('anti-phase' trial).

LFP processing

We used a publicly available data set containing LFP recordings from 3 Long-Evans rats, to obtain a reference oscillation that captures the statistics of ongoing oscillation in biological neural systems [36, 37]. Rats were chasing randomly placed drops of water or food, and neural activity was recorded using multichannel silicon probes inserted in area CA1 of the right dorsal hippocampus. The data contains LFP from 31 to 64 channels, recorded over multiple sessions.

We read the data using Neo [55]. We first re-sampled the data from 1250 Hz to 500 Hz and then high-pass filtered at 7 Hz using a FIR filter with Hamming window and 511 taps [56]. We normalised the signal by dividing the signal channel-wise with $\sqrt{2}\sigma_{LFP}$, where σ_{LFP} is the channel-wise standard deviation, resulting in the signals having a root mean square equal to a sine wave with amplitude 1.

In order to obtain a unique reference signal for each training trial, we split the recordings for each rat in chunks. In particular, we first split the data into 4 second segments, and picked a random channel for each segment. To extract instantaneous phase of the LFP oscillation for creating training targets, we convolved the signal with complex Morelet wavelets, consisting of the product of a complex sinusoid with a Gaussian with standard deviation $\frac{c}{2\pi f}$. We picked frequencies f from 7 to 9 in steps of 0.2, and set c ('cycles') to 7. For each trial we took the phase (angle) corresponding to the frequency with the highest power. The first second of the signal was discarded to

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

avoid boundary effects. We also discarded a small fraction of trials with artifacts, i.e. trials where the maximum absolute value of the signal was larger than 4 (after normalisation). We ended up with 5708 trials from rat 1, 5669 from rat 2, and 5632 from rat 3.

Training

We approximated Eq. 4 using the Euler–Maruyama method with timestep h, giving us for a rank-two network,

$$\begin{aligned}\mathbf{x}_{t'+1} =&(1-\frac{h}{\tau})\mathbf{x}_{t'}\\ &+\frac{h}{\tau}(\frac{1}{N}\sum_{i=1}^{2}[\mathbf{m}^{(i)}(\mathbf{n}^{(i)\mathsf{T}}\tanh(\mathbf{x}_{t'}))]+\mathbf{I}^{(osc)}u_{t'}+\mathbf{I}^{(s_b)}s_{b,t'}+\mathbf{I}^{(s_a)}s_{a,t'})\\ &+\sqrt{\frac{h}{\tau}}\mathcal{N}(0,2\sigma_{noise}^2).\end{aligned}$$

We use $h=2$ for training, whereas for the stability analysis and Figures, we take $h=0.5$. We defined a mean-squared error loss $\mathcal{L}$ between the targets $\hat{r}(t)$ and a linear readout $r(t)$ of the model's activity at time t,

$$\begin{aligned}r(t) &= \frac{1}{N}\mathbf{w}^{\mathsf{T}}\mathbf{x}(t),\\ \mathcal{L} &= \frac{1}{T-T_s}\int_{T_s}^{T}(r(t)-\hat{r}(t))^2 dt,\end{aligned} \tag{5}$$

where T_s is the time of the stimulus offset and T is the time at the end of a trial. In principle, one could also take $r(t)=\frac{1}{N}\mathbf{w}^{\mathsf{T}}\tanh(\mathbf{x}(t))$, but this generally leads to a different solution (Fig. 2D-F, Fig. S3.

We drew initial entries in the connectivity vectors from a zero-mean normal distribution, with a covariance matrix that is the identity $\mathbf{I}$, except for an initial covariance between $\mathbf{m}^{(1)},\mathbf{n}^{(1)}$ and $\mathbf{m}^{(2)},\mathbf{n}^{(2)}$ of 0.6 and a variance for $\mathbf{w}$ of 16 [32]. We minimised Eq 5, by optimising with stochastic gradient descent all entries in $\mathbf{I}^{(osc)},\mathbf{I}^{(s_a)},\mathbf{I}^{(s_b)},\mathbf{m}^{(1)},\mathbf{m}^{(2)},\mathbf{n}^{(1)},\mathbf{n}^{(2)}$, as well as a scalar multiplying the readout weights $\mathbf{w}$. We used the Adam optimizer in Pytorch, with default decay rates (0.9, 0.999) and learning rate 0.01 [57, 58].

We trained a separate RNN for LFP data from each rat. Shown in the main text is an RNN trained on LFP from rat 2, see Fig. S9 for RNNs trained on rat 1 and 3. During training, 90% of all trials were used for calculating the gradients, whereas the other 10% served as a validation set. We trained for 50 epochs, in batch sizes of 128, where one epoch denotes all training trials created from the LFP of one rat.

Analysing dynamics of oscillating low-rank RNNs

Dimensionality and dynamics of low-rank RNNs with periodic input

Here, we we show that the dynamics of a rank-2 RNN with periodic input lie on a 3 dimensional manifold. To facilitate the analysis, we first consider the dynamics of the network in the absence of transient stimuli ($s_a(t)=s_b(t)=0$), and no recurrent noise ($\sigma_{noise}=0$). Furthermore we set $\mathbf{m}^{(1)}\perp\mathbf{m}^{(2)}$, which one can always obtain by singular value decomposition of $\mathbf{J}$, even if during training $\mathbf{m}^{(1)}$ and $\mathbf{m}^{(2)}$ became correlated. We split up $\mathbf{I}^{(osc)}$ in the parts parallel and orthogonal to the $\mathbf{m}$'s,

$$\mathbf{I}^{(osc)}=\mathbf{I}_{\perp}+\mathbf{m}^{(1)}\alpha_1+\mathbf{m}^{(2)}\alpha_2.$$

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

We can then express $\mathbf{x}(t)$ in the orthogonal basis

$$\mathbf{x}(t) = \mathbf{m}^{(1)}\kappa_1(t) + \mathbf{m}^{(2)}\kappa_2(t) + \mathbf{I}_\perp v(t),$$

with $i \in \{0, 1\}$:

$$\kappa_i(t) = \frac{\mathbf{m}^{(i)\mathsf{T}}\mathbf{x}(t)}{\mathbf{m}^{(i)\mathsf{T}}\mathbf{m}^{(i)}},$$
$$v(t) = \frac{\mathbf{I}_\perp^{\mathsf{T}}\mathbf{x}(t)}{\mathbf{I}_\perp^{\mathsf{T}}\mathbf{I}_\perp}.$$

Here $v(t)$ is the reference oscillation filtered by the model's time constant τ. Using $*$ to denote convolution,

$$v(t) = \frac{1}{\tau}u(t) * e^{-\frac{t}{\tau}} + v(0)e^{-\frac{t}{\tau}}.$$

As both $u(t)$ and $v(t)$ explicitly depend on time, we first obtain the non-autonomous dynamical system $\mathbf{F}$,

$$\frac{d\kappa_1}{dt}, \frac{d\kappa_2}{dt} = \mathbf{F}(t, \kappa_1, \kappa_2), \tag{6}$$

with

$$\tau\frac{d\kappa_i}{dt} = -\kappa_i(t) + \frac{1}{N}\mathbf{n}^{(i)\mathsf{T}}\tanh(\mathbf{x}(t)) + \alpha_i u(t). \tag{7}$$

Next, we show that for periodic input this system can be considered as a three dimensional autonomous dynamical system. We introduce the variable $\theta = wt \mod 2\pi$ such that

$$\frac{d\theta}{dt} = \omega.$$

We take $u(t) = \sin(\theta)$ and we can now find the closed form solution for v,

$$v(t) = \frac{1}{\sqrt{(w\tau)^2 + 1}}\sin(\theta - \arctan(w\tau)) + ce^{-\frac{t}{\tau}}.$$

Here, the last term on the right hand side will decay to 0. Practically, we get rid of this dependence on t, by by taking appropriate $v(0)$ such that $c = 0$ (or by assuming the simulation has run for a little amount of time). Following this definition, both u and v are functions of θ, and we can consider the autonomous dynamical system with coordinates $\kappa_1, \kappa_2, \theta$,

$$\frac{d\kappa_1}{dt}, \frac{d\kappa_2}{dt}, \frac{d\theta}{dt} = \mathbf{G}(\kappa_1, \kappa_2, \theta).$$

Coordinate system for phase space figures

To create the phase space figures, we applied the following coordinate transform, where x', y', z' denote the coordinates in the figure,

$$x' = \cos(\theta)(r' - \kappa_1),$$
$$y' = \sin(\theta)(r' - \kappa_1),$$
$$z' = \kappa_2.$$

Given an initial condition for κ_1, i.e. $\kappa_1(0)$, one can always pick an r' such that this is an injective map for all t. This requires $r' < \kappa_1(t)$ for all t (otherwise trajectories would cross at the center). From Eq 7, we can see that $\kappa_1(t) > \frac{1}{N}|\mathbf{n}^{(1)}|_1 + |\alpha_1| \implies \frac{d\kappa_1}{dt} < 0$, thus if we pick $r' > \kappa_1(0) \geq \frac{1}{N}|\mathbf{n}^{(1)}|_1 + |\alpha_1|$, the requirement is satisfied.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Poincaré maps and linear stability analysis 273

We used Poincaré maps to study the stability of limit cycles [42, 43]. We took a cross section $Q = \{(\kappa_1, \kappa_2, \theta) : \mod \theta = 2\pi\}$, and created the iterative map from Q to itself,

$$\boldsymbol{\kappa}_{c+1} = \mathbf{P}(\boldsymbol{\kappa}_c),$$

where $\boldsymbol{\kappa_c} = \kappa_{1,c}, \kappa_{2,c} \in Q$ corresponds to the c'th intersection. Limit cycles correspond to fixed points $\boldsymbol{\kappa}^*$ for which

$$\boldsymbol{\kappa}^* = \mathbf{P}(\boldsymbol{\kappa}^*).$$

To study the stability of limit cycles we see what happens to a small perturbation η_0 in Q. Applying a Taylor expansion around $\boldsymbol{\kappa}^*$,

$$\boldsymbol{\kappa}^* + \eta_1 = P(\boldsymbol{\kappa}^*) + D_{\boldsymbol{\kappa}}\mathbf{P}(\boldsymbol{\kappa}^*)\eta_0 + h.o.t,$$

where $D_{\boldsymbol{\kappa}}$ denotes the gradient operator, the partial derivatives with respect to $\boldsymbol{\kappa}$. $D_{\boldsymbol{\kappa}}\mathbf{P}(\boldsymbol{\kappa}^*)$ is called the linearised Poincaré map at $\boldsymbol{k}^*$ To first order, perturbations scale proportional to the norm of its eigenvalues λ, called the Floquet (or characteristic) multipliers. To see this we can express η_0 as a linear combination of eigenvectors $\mathbf{e}$ of $D_{\boldsymbol{\kappa}}\mathbf{P}(\boldsymbol{\kappa}^*)$ (for some scalars v) and get the following expression for the perturbation after c cycles,

$$\eta_c = \sum_{i=1}^{2} v_i \mathbf{e}_i \lambda_i^c.$$

We now show how to obtain $D_{\boldsymbol{\kappa}}\mathbf{P}(\boldsymbol{\kappa}^*)$. Let $\psi(t, \boldsymbol{\kappa_0})$ denote the flow, a mapping from $(t, \boldsymbol{\kappa})$ to $\boldsymbol{\kappa}$, obtained by integrating $\mathbf{F}(t, \boldsymbol{\kappa})$ (Eq 6) for duration t, with initial conditions $\boldsymbol{\kappa_0}$. The Poincaré map then corresponds to [59]

$$\begin{aligned}\mathbf{P}(\boldsymbol{\kappa}_c) &= \psi(\mathcal{T}, \boldsymbol{\kappa_c}) \\ &= \int_0^{\mathcal{T}} \mathbf{F}(0, \psi(t, \boldsymbol{\kappa_c}))dt + \psi(0, \boldsymbol{\kappa_c}),\end{aligned}$$

where the second line is obtained by applying the fundamental theorem of calculus. $\mathcal{T}$ denotes one period of the reference oscillation. Defining $\mathbf{M}(t)$ as $D_{\boldsymbol{\kappa}}\psi(t, \boldsymbol{\kappa}_c)$, one can obtain a variational equation for the linearized Poincaré map,

$$D_{\boldsymbol{\kappa}}\mathbf{P}(\boldsymbol{\kappa}_c) = \int_0^{\mathcal{T}} D_{\psi}\mathbf{F}(0, \psi(t, \boldsymbol{\kappa}_c))D_{\boldsymbol{\kappa}_c}\psi(t, \boldsymbol{\kappa}_c)dt + \mathbf{I}$$

$$= \mathbf{M}(\mathcal{T})$$

with

$$\frac{d\mathbf{M}(t)}{dt} = D_{\psi}\mathbf{F}(t, \psi(t, \boldsymbol{\kappa}_c))\mathbf{M}(t).$$

Here, $\mathbf{M}(0) = \mathbf{I}$. $\mathbf{M}(t)$ is called the circuit or monodromy matrix. Since we can not obtain the circuit matrix analytically we approximated $\mathbf{M}(\mathcal{T})$ using the euler method with timestep h.

Coupled oscillators and population structure 277

Extracting the coupling function 278

We can rewrite the internal dynamics of the RNN in polar coordinates with

$$\kappa_1 = r\cos(\phi), \kappa_2 = r\sin(\phi).$$

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

We extracted the coupling function g,

$$\begin{aligned}\tau\frac{d\phi}{dt} &= \frac{1}{r^2N}((\kappa_1\mathbf{n}^{(2)} - \kappa_2\mathbf{n}^{(1)})^\mathsf{T}\tanh(\mathbf{x}) + (\kappa_1\alpha_2 - \kappa_2\alpha_1)u(\theta)),\\ g(\theta,\phi) &= \tau\frac{d\phi}{dt} - \omega.\end{aligned}$$

Note that the radius of the two stable limit cycles might not be exactly constant, or equal to each other, so for Fig. 3B and Fig. 3C we took for r the mean radius over the two cycles.

Reduced models

In order to find a simplified description of the dynamics of our models we assumed entries in the weight vectors of our network are N samples , indexed by j, from a probability distribution $p(\mathbf{y})$ [27, 32, 38, 39]. To keep the equations brief, we assume one input v here, which can be straightforwardly extended to multiple inputs. Then we have

$$\begin{aligned}&\mathbf{I}_{\perp j}, \mathbf{n}_j^{(1)}, \mathbf{n}_j^{(2)}, \mathbf{m}_j^{(1)}, \mathbf{m}_j^{(2)} \sim p(\mathbf{y}),\\ &p(\mathbf{y}) = p(I, n^{(1)}, n^{(2)}, m^{(1)}, m^{(2)}).\end{aligned}$$

We can then see Eq 7 as a sampling estimator for the recurrent input, which as $N \to \infty$ approaches the expectation,

$$\frac{d\kappa_i}{dt} \overset{N\to\infty}{=} -\kappa_i + \mathbb{E}_{p(\mathbf{y})}[n^{(i)}\tanh(\kappa_1 m^{(1)} + \kappa_2 m^{(2)} + v_1 I)].$$

We assume $p(\mathbf{y})$ is a mixture of L zero-mean Gaussians ([32]),

$$p(\mathbf{y}) = \sum_l^L w_l \mathcal{N}(0, \Sigma_l).$$

We can then obtain a mean-field description for the dynamics (see previous studies [27, 32, 38, 39] for a derivation). Here the effective connectivity consists of the covariances of the mixture components, modulated by a nonlinear 'gain' function (which approaches 0 when the non-linearity saturates),

$$\frac{d\kappa_i}{dt} = -\kappa_i + \sum_l^L w_l(\kappa_1\sigma^{(l)}_{m^{(1)}n^{(i)}} + \kappa_2\sigma^{(l)}_{m^{(2)}n^{(i)}} + v\sigma^{(l)}_{In^{(i)}})\mathbb{E}_{p(z)}[\tanh'(\sqrt{\Delta^{(l)}}z)], \tag{8}$$

with $\Delta^{(l)} = (\sigma^{(l)}_{m^{(1)}}\kappa_1)^2 + (\sigma^{(l)}_{m^{(2)}}\kappa_2)^2 + (\sigma^{(l)}_I v)^2$ and $z = \mathcal{N}(0,1)$.

Instead of numerically approximating the expectation as in previous studies [27, 32], we substitute $\mathrm{erf}(\frac{\sqrt{\pi}}{2}x)$ for $\tanh(x)$ and analytically obtain a simpler approximation.

$$\begin{aligned}\mathbb{E}_{p(z)}[\tanh'(\sqrt{\Delta}z)] &\approx \mathbb{E}_{p(z)}[\mathrm{erf}'(\frac{\sqrt{\pi\Delta}}{2}z)]\\ &\approx \mathbb{E}_{p(z)}[e^{-\frac{\pi\Delta}{4}z^2}]\\ &\approx \frac{1}{\sqrt{1+\frac{\pi}{2}\Delta}}.\end{aligned}$$

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Connectivity for mean-field model

By writing Eq 8 in polar coordinates we can see that the total change in phase of the model is the sum of the change in phase of its populations,

$$\frac{d\phi(t)}{dt} = \sum_{l}^{L} w_l \frac{d\phi^{(l)}(t)}{dt}. \tag{9}$$

Based on reverse engineering the dynamics and connectivity of our trained networks (2,3B,C,S5), we can now choose covariances in order to design a model that approximates the coupled oscillator equations (Eqs 3).

We first initialize a population that generates oscillations with angular velocity ω. We set this population unconnected to the input ($\sigma_I^{(i)} = 0$ for all i). When taking $\sigma_{m^{(2)}n^{(1)}} = -\sigma_{m^{(1)}n^{(2)}}$ and $\sigma_{m^{(1)}n^{(1)}} = \sigma_{m^{(2)}n^{(2)}}$ [27, 38, 40] this populations produces oscillations, with constant frequency,

$$\frac{d\phi^{(p1)}(t)}{dt} = \frac{w_{p_1}\sigma^{(p_1)}_{n^{(1)}m^{(2)}}}{\sqrt{1+\frac{\pi}{2}r^2)}}.$$

.

Next, we create two populations that together implement the coupling function,

$$g(\phi,\theta) = \frac{\sigma_{n^{(1)}m^{(2)}}\cos(2\phi)}{\sqrt{1+\frac{\pi}{2}(r^2+\sigma^2_{I^{(osc)}}v(\theta)^2))}}. \tag{10}$$

Based on Fig. 3B, it seems reasonable to assume that the bistable dynamics stem from $\sin(2\phi)$ and $\sin(2\theta)$ terms, which we obtained by setting $\sigma_{n^{(1)}m^{(2)}} = \sigma_{n^{(2)}m^{(1)}}$ (or $\sigma_{n^{(1)}m^{(1)}} = -\sigma_{n^{(2)}m^{(2)}}$), and $\sigma_{I^{(osc)}} \neq 0$.

In order to get the stimulus-induced bifurcation observed in Fig. 2C,Fig. 3C, we, additionally set the following connectivity for the input to the coupling populations, $\sigma^{(p_2)}_{I^{(s_a)}} \neq 0, \sigma^{(p_3)}_{I^{(s_b)}} \neq 0, \sigma^{(p_2)}_{I^{(osc)}n^{(1)}} = -\sigma^{(p_2)}_{I^{(osc)}n^{(2)}} = -\sigma^{(p_3)}_{I^{(osc)}n^{(1)}} = \sigma^{(p_3)}_{I^{(osc)}n^{(2)}}$. Then the equations for the coupling populations read,

$$\frac{d\phi^{(p2)}(t)}{dt} = w_{p_2}\frac{\frac{\sqrt{2}}{r}\sigma^{(p_2)}_{n^{(1)}I^{(osc)}}\sin(\phi+\frac{1}{4})v(\theta) + \sigma^{(p_2)}_{n^{(1)}m^{(2)}}\cos(2\phi)}{\sqrt{1+\frac{\pi}{2}(r^2+\sigma^{2(p_2)}_{I^{(osc)}}v(\theta)^2+\sigma^{2(p_2)}_{I^{(s_a)}}s_a(t)^2)}},$$

$$\frac{d\phi^{(p3)}(t)}{dt} = w_{p_3}\frac{\frac{\sqrt{2}}{r}\sigma^{(p_3)}_{n^{(2)}I^{(osc)}}\sin(\phi-\frac{3}{4})v(\theta) + \sigma^{(p_3)}_{n^{(1)}m^{(2)}}\cos(2\phi)}{\sqrt{1+\frac{\pi}{2}(r^2+\sigma^{2(p_3)}_{I^{(osc)}}v(\theta)^2+\sigma^{2(p_3)}_{I^{(s_b)}}s_b(t)^2)}}.$$

When both s_a and s_b are zero, the $\sin(\phi)$ terms cancel out and we retrieve the desired coupling function (Eq 10; where the $\sin(2\phi)$ terms dominate). When either stimulus a or b is on, population 2 or 3 is inhibited, respectively (the non-linearities of the units in this population saturate). Then the $\sin(\phi)$ term of the population that is unaffected by the stimulus takes over the coupling function, mimicking what we saw in Fig. 3B and Fig. 3C. For exact values of the parameters, see Fig. S6.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Acknowledgments

We thank Richard Gao, Kabir Dabholkar and Stefanie Liebe for insightful discussions, and Pedro J. Gonçalves, Elia Turner and Zinovia Stefanidi for feedback on the manuscript. This work was supported by the German Research Foundation (DFG): SFB 1089 'Synaptic Micronetworks in Health and Disease', the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A and the ISRAEL SCIENCE FOUNDATION (grant No. 1442/21) and HFSP research grant (RGP0017/2021). MP and JHM are members of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645, and MP is a member of the International Max Planck Research School for Intelligent Systems (IMPRS-IS).

Author contributions

MP: Conceptualization, Formal Analysis, Software, Writing – Original Draft Preparation JHM: Conceptualization, Funding Acquisition, Supervision, Writing – Review & Editing OB: Conceptualization, Funding Acquisition, Supervision, Writing – Original Draft Preparation

References

1. Delcomyn F. Neural Basis of Rhythmic Behavior in Animals. Science. 1980;210(4469):492–498.
2. Marder E, Bucher D. Understanding Circuit Dynamics Using the Stomatogastric Nervous System of Lobsters and Crabs. Annual Review of Physiology. 2007;69(1):291–316.
3. Buzsáki G. Rhythms of the Brain. 1st ed. Oxford University Press; 2006.
4. Buzsáki G, Tingley D. Space and Time: The Hippocampus as a Sequence Generator. Trends in Cognitive Sciences. 2018;22:853–869.
5. Lisman JE, Jensen O. The Theta-Gamma Neural Code. Neuron. 2013;77(6):1002–1016.
6. Lisman J. The theta/gamma discrete phase code occuring during the hippocampal phase precession may be a more general brain coding scheme. Hippocampus. 2005;15(7):913–922.
7. Kayser C, Ince RAA, Panzeri S. Analysis of Slow (Theta) Oscillations as a Potential Temporal Reference Frame for Information Coding in Sensory Cortices. PLOS Computational Biology. 2012;8(10):1–13.
8. Hopfield JJ. Pattern recognition computation using action potential timing for stimulus representation. Nature. 1995;376(6535):33–36.
9. Fell J, Axmacher N. The role of phase synchronization in memory processes. Nature Reviews Neuroscience. 2011;12(2):105–118.
10. O'Keefe J, Recce ML. Phase relationship between hippocampal place units and the EEG theta rhythm. Hippocampus. 1993;3:317–330.
11. Kraskov A, Quiroga RQ, Reddy L, Fried I, Koch C. Local Field Potentials and Spikes in the Human Medial Temporal Lobe are Selective to Image Category. Journal of Cognitive Neuroscience. 2007;19(3):479–492.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

12. Turesson H, Logothetis N, Hoffman K. Category-selective phase coding in the superior temporal sulcus. Proceedings of the National Academy of Sciences. 2012;109(47):19438–19443.

13. Watrous AJ, Deuker L, Fell J, Axmacher N, Eichenbaum H. Phase-amplitude coupling supports phase coding in human ECoG. eLife. 2015;4:e07886.

14. Siegel M, Warden MR, Miller EK. Phase-dependent neuronal coding of objects in short-term memory. PNAS. 2009;15:21341–21346.

15. Liebe S, Hoerzer GM, Logothetis NK, Rainer G. Theta coupling between V4 and prefrontal cortex predicts visual short-term memory performance. Nature neuroscience. 2012;15(3):456–462.

16. Liebe S, Niediek J, Pals M, Reber TP, Faber J, Bostroem J, et al. Phase of firing does not reflect temporal order in sequence memory of humans and recurrent neural networks. bioRxiv. 2022;.

17. Watrous AJ, Miller J, Qasim SE, Fried I, Jacobs J. Phase-tuned neuronal firing encodes human contextual representations for navigational goals. eLife. 2018;7:e32554.

18. Durstewitz D, Koppe G, Thurm MI. Reconstructing Computational Dynamics from Neural Measurements with Recurrent Neural Networks. bioRxiv. 2022;.

19. Langdon C, Engel TA. Latent circuit inference from heterogeneous neural responses during cognitive tasks. bioRxiv. 2022;.

20. Vyas S, Golub MD, Sussillo D, Shenoy KV. Computation Through Neural Population Dynamics. Annual Review of Neuroscience. 2020;43(1):249–275.

21. Pandarinath C, O'Shea DJ, Collins J, Jozefowicz R, Stavisky SD, Kao JC, et al. Inferring single-trial neural population dynamics using sequential auto-encoders. Nature Methods. 2018;15(10):805–815.

22. Barak O. Recurrent neural networks as versatile tools of neuroscience research. Current Opinion in Neurobiology. 2017;46:1–6.

23. Shenoy KV, Sahani M, Churchland MM. Cortical Control of Arm Movements: A Dynamical Systems Perspective. Annual Review of Neuroscience. 2013;36(1):337–359.

24. Chaudhuri R, Gerçek B, Pandey B, Peyrache A, Fiete I. The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. Nature Neuroscience. 2019;22(9):1512–1520.

25. Gallego JA, Perich MG, Miller LE, Solla SA. Neural Manifolds for the Control of Movement. Neuron. 2017;94(5):978–984.

26. Funahashi K, Nakamura Y. Approximation of dynamical systems by continuous time recurrent neural networks. Neural Networks. 1993;6(6):801–806.

27. Beiran M, Dubreuil A, Valente A, Mastrogiuseppe F, Ostojic S. Shaping Dynamics With Multiple Populations in Low-Rank Recurrent Networks. Neural Computation. 2021;33(6):1572–1615.

28. Barbosa J, Proville R, Rodgers CC, DeWeese MR, Ostojic S, Boubenec Y. Flexible selection of task-relevant features through population gating. bioRxiv. 2022;.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

29. Driscoll L, Shenoy K, Sussillo D. Flexible multitask computation in recurrent networks utilizes shared dynamical motifs. bioRxiv. 2022;.

30. Song HF, Robert Yang G, Wang XJ. Training Excitatory-Inhibitory Recurrent Neural Networks for Cognitive Tasks: A Simple and Flexible Framework. PLOS Computational Biology. 2016;12(2):1–30.

31. Mante V, Sussillo D, Shenoy KV, Newsome WT. Context-dependent computation by recurrent dynamics in prefrontal cortex. Nature. 2013;503(7474):78–84.

32. Dubreuil A, Valente A, Beiran M, Mastrogiuseppe F, Ostojic S. The role of population structure in computations through neural dynamics. Nature Neuroscience. 2022;25(6):783–794.

33. Turner E, Dabholkar KV, Barak O. Charting and Navigating the Space of Solutions for Recurrent Neural Networks. In: Advances in Neural Information Processing Systems. vol. 34; 2021.

34. Finkelstein A, Fontolan L, Economo MN, Li N, Romani S, Svoboda K. Attractor dynamics gate cortical information flow during decision-making. Nature Neuroscience. 2021;24(6):843–850.

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

36. Mizuseki K, Sirota A, Pastalkova E, Buzsáki G. Theta Oscillations Provide Temporal Windows for Local Circuit Computation in the Entorhinal-Hippocampal Loop. Neuron. 2009;64(2):267–280.

37. Mizuseki K, Sirota A, Pastalkova E, Buzsáki G. Multi-unit recordings from the rat hippocampus made during open field foraging; 2009.

38. Mastrogiuseppe F, Ostojic S. Linking Connectivity, Dynamics, and Computations in Low-Rank Recurrent Neural Networks. Neuron. 2018;99(3):609–623.e29.

39. Schuessler F, Dubreuil A, Mastrogiuseppe F, Ostojic S, Barak O. Dynamics of random recurrent networks with correlated low-rank structure. Phys Rev Res. 2020;2:013111.

40. Susman L, Brenner N, Barak O. Stable memory with unstable synapses. Nature Communications. 2019;10(1):4441.

41. Galgali AR, Sahani M, Mante V. Residual dynamics resolves recurrent contributions to neural computation. Nature Neuroscience. 2023;.

42. Strogatz S. Nonlinear dynamics and chaos : with applications to physics, biology, chemistry, and engineering. 2nd ed. Westview Press; 2015.

43. Sato S, Gohara K. Poincaré Mapping of continuous Recurrent Neural Networks excited by Temporal External Input. Int J Bifurc Chaos. 2000;10:1677–1696.

44. Pahor A, Jaušovec N. The Effects of Theta and Gamma tACS on Working Memory and Electrophysiology. Frontiers in Human Neuroscience. 2018;11.

45. Lisman JE, Idiart MAP. Storage of 7 ± 2 Short-Term Memories in Oscillatory Subcycles. Science. 1995;267(5203):1512–1515.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

46. Scheffer-Teixeira R, Tort AB. On cross-frequency phase-phase coupling between theta and gamma oscillations in the hippocampus. eLife. 2016;5:e20515.

47. Stankovski T, Pereira T, McClintock PVE, Stefanovska A. Coupling functions: Universal insights into dynamical interaction mechanisms. Reviews of Modern Physics. 2017;89(4):045001–.

48. Stankovski T, Duggento A, McClintock PVE, Stefanovska A. Inference of Time-Evolving Coupled Dynamical Systems in the Presence of Noise. Phys Rev Lett. 2012;109:024101.

49. Khona M, Fiete IR. Attractor and integrator networks in the brain. Nature Reviews Neuroscience. 2022;23(12):744–766.

50. Brennan C, Aggarwal A, Pei R, Sussillo D, Proekt A. One dimensional approximations of neuronal dynamics reveal computational strategy. PLOS Computational Biology. 2023;19(1):1–27.

51. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences. 1982;79(8):2554–2558.

52. 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. 2019;2019:15629–15641.

53. Rajan K, Harvey CD, Tank DW. Recurrent Network Models of Sequence Generation and Memory. Neuron. 2016;90(1):128–142.

54. Hahn LA, Balakhonov D, Lundqvist M, Nieder A, Rose J. Oscillations without cortex: Working memory modulates brainwaves in the endbrain of crows. Progress in Neurobiology. 2022;219:102372.

55. Garcia S, Guarino D, Jaillet F, Jennings TR, Pröpper R, Rautenberg PL, et al. Neo: an object model for handling electrophysiology data in multiple formats. Frontiers in Neuroinformatics. 2014;8:10.

56. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods. 2020;17:261–272.

57. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Advances in Neural Information Processing Systems. vol. 32; 2019. p. 0.

58. Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. In: 3rd International Conference on Learning Representations; 2015. p. 0.

59. Hirsch MW, Smale S, Devaney RL. Differential Equations, Dynamical Systems, and an Introduction to Chaos. 3rd ed. Academic Press; 2013.

60. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825–2830.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

Supporting information

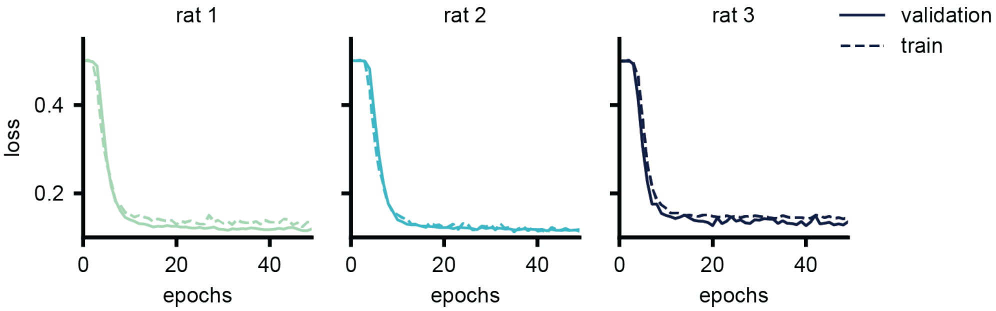

Fig S1. Loss curves. Loss over epochs for three models, each trained with LFP data from a seperate rat. An epoch denotes one pass through all trials in the training or validation set. The validation trials are defined before training and are not used for calculating gradients. See Training for training details.

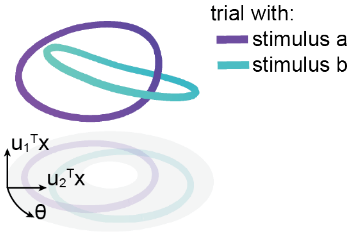

Fig S2. Dynamics of unconstrained networks are similar to low-rank RNNs. We also trained RNNs without rank constraint. For these networks initial entries in the recurrent weight matrix J were drawn from a zero mean Gaussian with variance $\frac{g^2}{N}$. We also add a regularisation term R to the loss, which keeps the average firing rates close to 0, to avoid a rate-coding solution: $R = \frac{1}{N}\sum_i^N(\frac{1}{T}\int_0^T \mathbf{x}_i(t)dt)^2$. We set g to 0.6 and took a learning rate of 0.001, with the training setup otherwise as for the low-rank networks (Training). In order to find a basis similar to the one used for plotting the dynamics of the low-rank RNN we took the following approach. First we calculated basis vectors for the activity due to recurrent dynamics: $\mathbf{J}\tanh(\mathbf{x}(t))$, by performing a Principal Component Analysis (singular vector decomposition): $\mathbf{U\Sigma V}^\mathsf{T} = \mathbf{J}\tanh(\mathbf{X})$, where $\mathbf{X}$ is an $N \times 2\mathcal{T}$ matrix containing the activity of all units for one period of oscillation for both a trial on which stimulus a and b has been shown. We took the first two columns (principal components), $\mathbf{u}_1$ and $\mathbf{u}_2$, of $\mathbf{U}$, as well as the input vector $\mathbf{I}^{(osc)}$ orthogonalised with respect to these two principal components as basis for $\mathbf{X}$. This basis retains 77% of the variance of $\mathbf{X}$ (measured by r^2). Since now, similar to the low-rank case, we can write the projection of $\mathbf{x}(t)$ on $\mathbf{I}_\perp^{(osc)}$ as a function of θ, we can plot trajectories with coordinates $(\theta, \mathbf{u}_1^\mathsf{T}\mathbf{x}, \mathbf{u}_2^\mathsf{T}\mathbf{x})$, and we obtain two stable cycles, linked in phase-space, as in Fig. 2.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

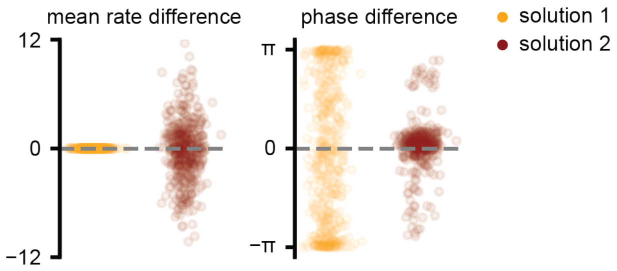

Fig S3. Statistics of the phase and mean rate of trained RNN units distinguishes solutions. There are two qualitatively different solutions to the working memory task used in this study, that can be distinguished on the single unit level. Each dot represents for a single unit in the RNN, the mean difference in firing rate (left) or phase (right) between trials with stimulus a and trials with stimulus b. Shown are two different models (colors). When trained with a linear readout on the rates $\tanh(\mathbf{x}(t))$ instead of on $\mathbf{x}(t)$ (Training), RNNs converged to the second solution (red). For these models, the mean firing rate is systematically shifted depending on the stimulus shown, whereas for the model shown in the main text (yellow) the mean firing rate is close to 0 for all units. When the firing rate of a unit shifts away from zero, its contribution to the recurrent dynamics (and the readout when reading out $\tanh(\mathbf{x}(t))$) decreases due to its nonlinearity saturating (Reduced models; [27, 32, 38]). The difference in single unit phase however is more pronounced for the first solution.

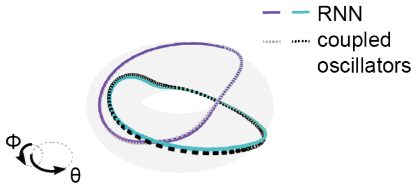

Fig S4. Converged RNN dynamics match those of coupled oscillators. We simulated two coupled oscillators with the coupling function extracted from a trained network (Fig. 3B). The two oscillators represent the RNN's phase (ϕ) and the reference oscillation phase (θ). Starting simulations from various initial conditions demonstrates that the coupling function induces bistability, as all simulations converge to one of two stable cycles on the torus. Furthermore, the convergent trajectories of the coupled oscillators are a close match to those of the full RNN projected into the same space (ϕ, θ), indicating that the coupled oscillator description is appropriate for the RNN.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

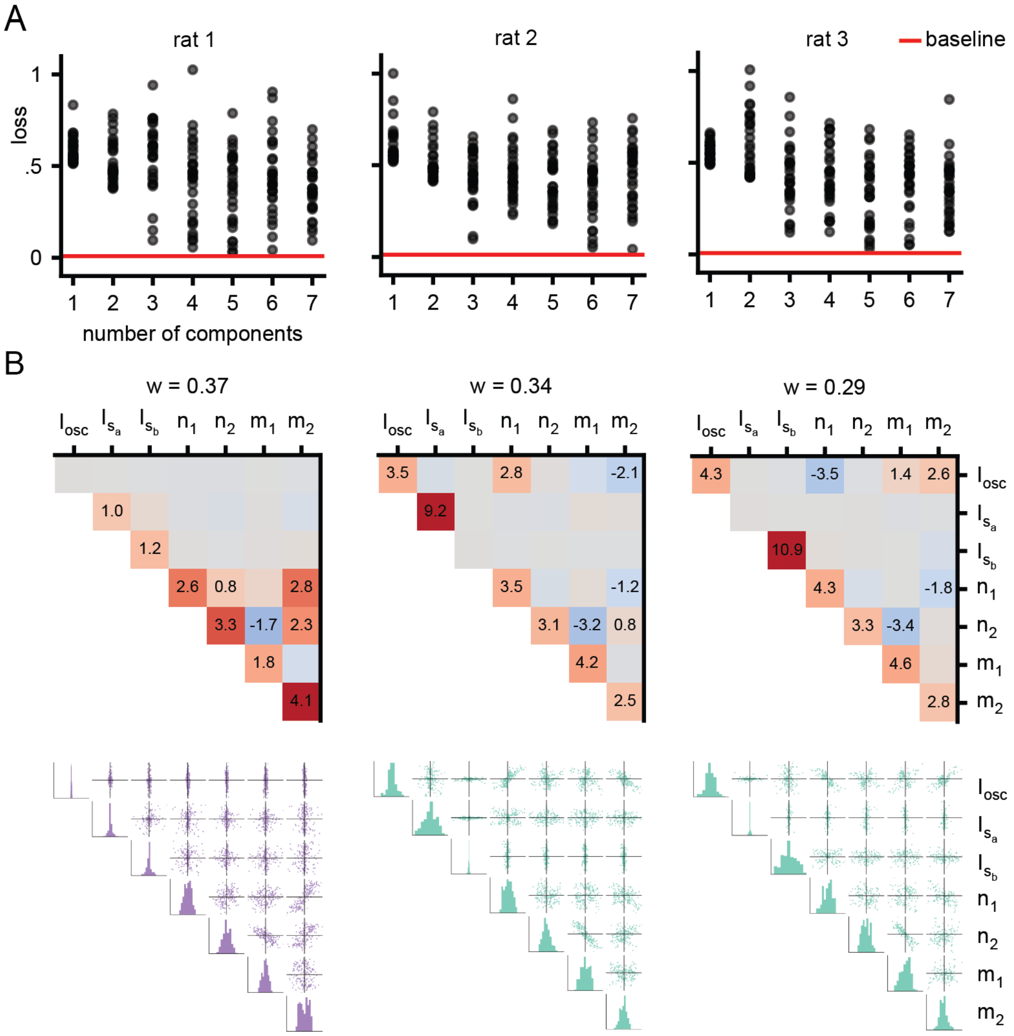

Fig S5. Connectivity of trained models. A: In order to study the connectivity of trained models, we fitted a mixture of Gaussians with 1 to 7 mixture components to the connectivity vectors of three different models [32], each trained with LFP data from a seperate rat. For this, we used variational inference with a Gaussian prior on the mean with precision 1e5 and mean 0 [60]. After each fit, we resampled the weights 30 times and computed the loss over a batch of 128 trials with a pure sine wave as reference oscillation. Altough for no amount of components we reliable were able to resample functioning models, from 3 components onwards a small fraction of sampled model have a comparable loss to the original trained model (red line). B: The covariance structure when fitting three components to weights of a trained network give us some hints as to what is needed to get a functioning model (top: covariances, bottom: pair plots). One component is unconnected (zero covariance) to the reference oscillation and has a skew-symmetric structure between the singular vectors. This structure generates oscillations [27, 38, 40]. The other two components, each connected to one stimulus and with opposite covariance between the input and singular vectors, together implement the coupling function.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

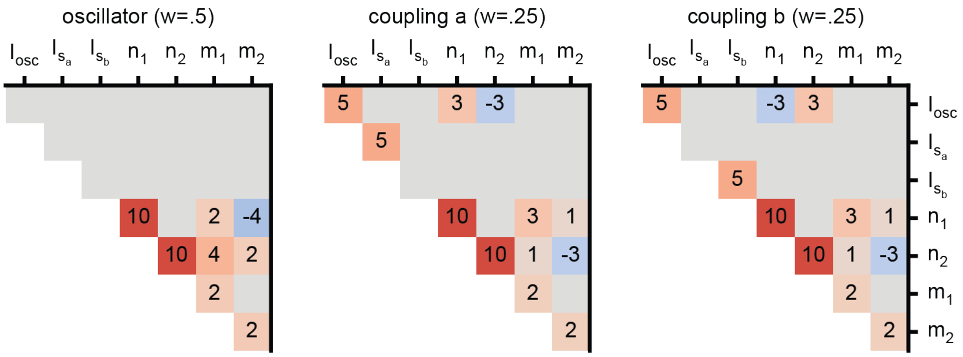

Fig S6. Connectivity of mean-field model. A: The designed covariance structure for the reduced model shown in Fig. 4D,E, which leads to dynamics similar to the trained models. It also shares connectivity structure with trained models (Fig. S5B), namely having one component unconnected to the reference oscillation that autonomously generates oscillations, and having the other two components, each connected to one stimulus, implement the coupling function.

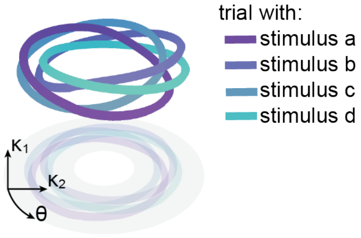

Fig S7. Geometry of dynamics is preserved for 4 stimuli. We trained a network to maintain 1 of 4 stimuli at a given trial, by producing an output oscillation at $-0.2\pi, -0.7\pi, -1.4\pi$ or -1.7π radians offset with respect to the reference LFP, for stimulus a, b, c or d, respectively. Again we find a stable limit cycle for each stimuli, which form linked cycles in phase-space.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

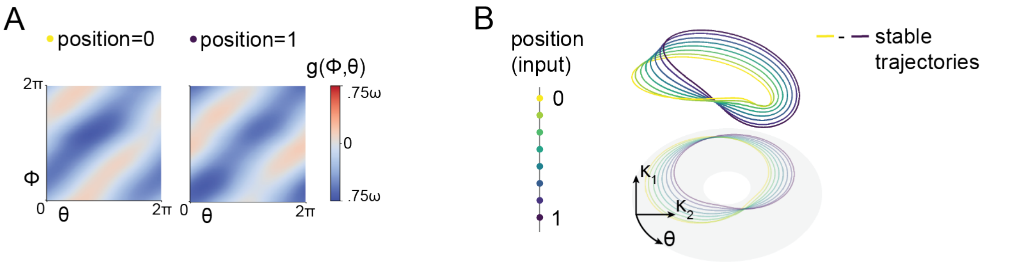

Fig S8. Phase precession through translation of the coupling function. Phase precession in rat hippocampus entails a place cell changing its relative phase of firing as a result of rat moving through a place field [10]. We here show how to create a network that changes its relative phase of oscillation depending on a continuous valued stimulus input. We setup a network with connectivity drawn from a mixture of three Gaussians, again with two components implementing a coupling function and one component implementing an oscillator. The network receives sinusoidal reference input with phase θ as $\sin(\theta), \cos(\theta)$ through input vectors $\mathbf{I}^{(osc_a)}, \mathbf{I}^{(osc_b)}$ respectively, as well as continuous valued stimulus input, representing place field position, $s(t) \in [0, 1)$ as $\sin(s(t)\frac{1}{2}\pi), \cos(s(t)\frac{1}{2}\pi)$ through input vectors $\mathbf{I}^{(s_a)}, \mathbf{I}^{(s_b)}$ respectively. The oscillator component has connectivity equal to Fig. S6, whereas for the coupling components (p_2, p_3) we define the covariance matrices as follows: For the off-diagonal elements, $\sigma^{(p_2)}_{n^{(1)}I^{(osc_b)}} = -\sigma^{(p_2)}_{n^{(2)}I^{(osc_a)}} = \sigma^{(p_3)}_{n^{(2)}I^{(osc_a)}} = \sigma^{(p_3)}_{n^{(1)}I^{(osc_b)}}$, otherwise 0. For the diagonal elements, component two has zero variance for (is unconnected to) $\mathbf{I}^{(s_a)}$ and component three has zero variance for $\mathbf{I}^{(s_b)}$. This gives a coupling function of the form:

$$g(\theta, \phi) = \sin(\theta - \phi)\frac{a}{\sqrt{b + c\sin(\frac{\pi}{2}s(t))^2}} + \cos(\theta - \phi)\frac{a}{\sqrt{b + c\cos(\frac{\pi}{2}s(t))^2}},$$

for constants a, b, c that depend on the exact values of the variances and covariances of the mixture components. For large c and small b, the coupling function will change from $a\sin(\theta - \phi)$ to $a\cos(\theta - \phi)$ as $s(t)$ changes from 0 to 1. A: The coupling function of a network (N=2056 units) with connectivity drawn from mixture of Gaussians as described above, for $s(t) = 0$ and $s(t) = 1$. The coupling function translates between these two states as a consequence of position input $s(t)$. B: Different trials on which $s(t)$ is tonically presented at different values lead the the network locking to a unique phase difference with respect to the reference oscillation. In phase-space, this appears as the stable cycle shifting along the surface of a torus.

bioRxiv preprint doi: https://doi.org/10.1101/2023.04.11.536352; this version posted April 11, 2023. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC 4.0 International license.

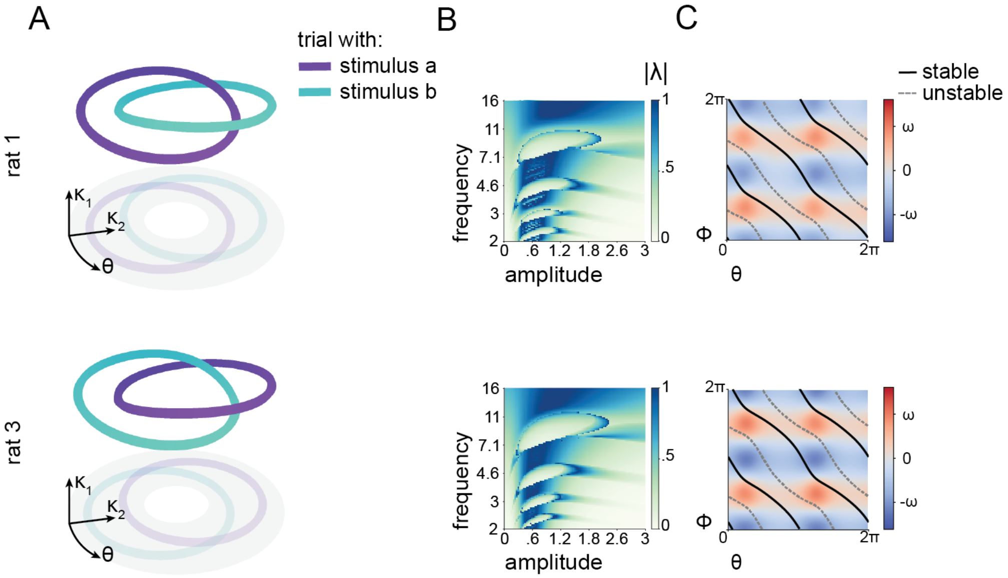

Fig S9. Results are consistent over data from multiple rats. Dynamics (A), bifurcation analysis (B) and coupling function (C) extracted from models trained with LFP data from rat 1 (top row) or rat 3 (bottom row) look qualitatively similar to that of rat 2 (Fig. 2-4), with only slight quantitative differences.