bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity.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-ND 4.0 International license.

Disk-Drive-Like Operations in the Hippocampus

Wilten Nicola[1,2,3*], David Dupret[4], and Claudia Clopath[5]

[1] *University of Calgary, Cumming School of Medicine, Department of Cell Biology and Anatomy*
[3] *Hotchkiss Brain Institute*
[2] *University of Calgary, Faculty of Science, Department of Physics and Astronomy*
[*] *Corresponding Author: wilten.nicola@ucalgary.ca*
[4] *Medical Research Council Brain Network Dynamics Unit, Nuffield Department of Clinical Neurosciences, University of Oxford*
[5] *Imperial College London, Department of Bioengineering*

October 5, 2022

Abstract

The rapid computation of re-playable memories within the hippocampus in the form of spike sequences is a near computer-like operation. Information can be encoded once during the initial experience, and replayed numerous times after in a compressed-time representation [1–8]. Theta oscillations, sharp-wave ripples, and attractor dynamics have been posited to collectively play a role in the formation and replay of memories. However, the precise interplay between these dynamical states remains elusive. Here, we show that the memory formation dynamics and operations of the hippocampus are not just computer-like, but map directly onto the dynamics and operations of a disk-drive. We constructed a tripartite spiking neural network model where the hippocampus is explicitly described as a disk drive with a rotating disk, an actuator arm, and a read/write head. In this Neural Disk Drive (NDD) model, hippocampal oscillations map to disk rotations in the rotating disk network while attractor dynamics in the actuator arm network point to "tracks" (spike assemblies) on the disk. The read/write head then writes information onto these tracks, which have temporally-structured spikes. Tracks can be replayed during hippocampal ripples for consolidation. We confirmed the existence of interneuron-ring-sequences, predicted by the rotating disk network, in experimental data. Our results establish the hippocampus as a brain region displaying explicit, computer-like operations. Based on the known interactions between the hippocampus and other brain areas, we anticipate that our results may lead to additional models that revisit the hypothesis that the brain performs explicit, computer-like operations.

1 Introduction

The metaphor that the brain operates as a computer has been pervasive in neuroscience since Jon von Neumann's pioneering work in the 1950's [9–19]. At the near simultaneous dawn of computer science and electrophysiologial-based, computational neuroscience, Von Neumann postulated that the nearly discrete action potentials fired by neurons were comparable to the digital binary units or bits in the vacuum tubes and transistors of early computers [9]. Such a metaphor, if made into a concrete model, could help reach a comprehensive understanding of, and formulate predictions on, the nature of the computational operations underlying brain functions.

Unfortunately, von Neumann's efforts were limited by the state of knowledge of the brain at the time. However, the last decades of neuroscientific work have now shed important insights into the cellular substrates and network dynamics at the nexus of brain and behaviour, laying the foundational knowledge about how some neuronal regions specialize and adapt to perform specific operations [20–24]. Notably, the hippocampus of the mammalian brain holds mnemonic information used to inform behaviour [1–8, 20, 25–46]. Likewise, modern computers hold information for further operations using dedicated components: Hard Disk Drives (HDDs) [47].

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Accordingly, we test a more direct version of von Neumann's "brain-as-a-computer" analogy by establishing a theoretical framework where hippocampal operations and dynamics are mapped directly onto those of a computer's disk drive in the Neural Disk Model of Hippocampal Function. The NDD model is a tri-partite network where each network maps onto the core components of a disk drive: the rotating disk network, the read/write head network, and the actuator arm network. The NDD model also successfully maps theta oscillations and sharp-wave ripples to disk-rotations while attractor dynamics act as the actuator arm, pointing to individual tracks or cell assemblies. The zoo of observed hippocampal replays (forward replay, reverse replay, extended replay, fragmented replay) can interpreted as specific data-accessing events in the NDD model. Finally, we detected the interneuron-ring sequences in multi-unit recordings from rats, as predicted by the rotating disk network.

2 Results

2.1 Mapping Hippocampal Behaviours to Disk Drive Dynamics

We start by first describing hippocampal dynamics. The spike times of hippocampal pyramidal cells and interneurons are organized on multiple timescales by a collection of network oscillations that are observed as rhythmic fluctuations in the local field potentials (LFPs) and correlate with behavioural states and memory processing stages (Figure 1A). Sequences of said spikes are observed on two time scales: temporally compressed spike sequences during hippocampal sharp-waves ripples (SWRs, Figure 1A-B) and temporally dilated forms of these spike sequences during hippocampal theta oscillations (Figure 1A,C). The SWR LFP event is a 125–250 Hz oscillatory event lasting approximately 100 ms and supporting memory consolidation [1–8, 26, 48–53]. The theta oscillation, on the other hand, is a 5–12 Hz oscillation that dominates hippocampal LFPs, organising temporally structured firing activity of pyramidal cells in support of learning during active exploration (Figure 1C) [20, 25, 27, 44, 54, 55]. During rest/sleep, theta-nested neural patterns are compressed and replayed in sharp wave/ripples (SWRs) [1–8,56–62]. The relation between compressed sharp-wave sequences and theta sequences may occur through an oscillatory-interference mechanism where one oscillator controls spike times during SWRs, and a second oscillator dilates SWR-sequences into theta sequences by creating an interference pattern that dilates the sequential content of the carrier waves into the envelope phases (Figure 1C) [63]. This postulated interference mechanism provides a mechanistic explanation for hippocampal phase precession [20,44,64–72], and explicitly links theta sequences during a single cycle of the theta oscillation to compressed spike sequences during a SWR [63]. Indeed, spike sequences during a cycle of the theta oscillation are also a compressed representation of the firing fields of individual cells, and are known to have comparable compression rates to replay sequences in SWRs [48]. Hebbian plasticity allows for one-shot learning of new sequences by using existing theta-sequences as a compressible temporal backbone (Figure 1D). Different populations of neurons within a theta sequence can be selected by biasing currents to store potential information (Figure 1E) allowing for discrete memories to be stored in different populations of pyramidal neurons [73].

We discovered that collectively, these operations could be explicitly mapped onto those performed by a computer's Hard Disk Drive (HDD). To start constructing our brain-machine mapping of hippocampal dynamics, we first catalogued the individual components and operations of a disk drive and mapped these operations to the hippocampal equivalent (Figure 1F). The central operations of an HDD are performed by three components: an Actuator Arm (AA), a Rotating Disk (RD), and a Read/Write head (R/W) (Supplementary Material Section 1, Supplementary Figure S1, Supplementary Video 1, Figure 1E). The actuator arm points to a segment of the disk referred to as a track, which is further subdivided into sectors (Figure 1F). The read/write head, which is located on the apex of the actuator arm, writes new information in the form of bits, or reads previously stored bits on tracks and sectors. These three mechanical components are described by three dynamically evolving parameters that control the disk drive: the location of the actuator arm (ψ_A, A for "Arm"), the disk rotation speed (θ_S, s for "spinning") and the read/write speed of the head ($\theta_{R/W}$, R/W for "read/write", Figure 1F). As the disk spins in a single revolution (θ_S), a sequence of sectors on a track appears directly beneath the R/W head on the actuator arm (Supplementary Video 1, Figure 1G). This sequence appears on the fast, intrinsic time-scale of the disk rotation speed (θ_s). The sequence of sectors can be read from or written to in

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

a single disk rotation (Supplementary Video 2, Supplementary Video 3). This mimics the compressed time-scale sequences which occur during hippocampal SWRs [1–8, 56, 73].

The sequence of sectors can also be accessed on a slower time-scale, by allowing the disk to perform slightly more than a full revolution for each read/write cycle of $\theta_{R/W}$ ($\theta_{R/W} < \theta_S = \theta_S - \epsilon_\theta$, Figure 1H, Supplementary Video 3). In this operating mode, the read/write head accesses each subsequent sector on a slower time scale which mimics the dilated or behavioural time-scale sequences that occur during hippocampal theta oscillations. Bit-sequences can be written to these sectors during the slower-access mode and then rapidly replayed after (Figure 1I, Supplementary Video 2). This is similar to the time-compressible, one-shot learning of spike-sequences observed in the hippocampus ([46, 48]). Finally, different tracks are accessible by the actuator arm when the actuator arm changes its location (Figure 1J). The location of the actuator arm, ψ_A behaves similarly to a line attractor. So long as no force is impinged on the actuator arm, the location ψ_A is constant and thus all actuator arm positions are stable. For an actual actuator arm, the range on ψ_A is restricted to some interval within $[0, \pi]$. Applying a force to the actuator arm moves ψ_A to a new track. This mimics how bias currents within the hippocampus, as a result of place, context, or other stimuli, can select different populations of cells during replays [73]. These biasing currents may be regulated or produced by attractor dynamics [74–78].

2.2 Constructing the Rotating Disk, Read/Write Head, and Actuator Arm Networks

Having qualitatively linked the dynamics of the hippocampus to those of a disk drive, we wondered if an explicit neural model could be constructed of the individual disk-drive components. Such a model, if constructed, would merge two modelling paradigms into a concrete device: interfering oscillators and attractor dynamics (see [79] for a complementary grid-cell model). To that end, we trained recurrent spiking neural networks to display the dynamics of a disk drive as a tripartite network with a rotating disk network, a read/write head network, and an actuator arm network [63, 80]. We started by modelling the three networks individually to reproduce disk-drive dynamics constrained to the hippocampal parameter ranges.

First, we considered the rotating disk network (Figure 2A-I). The RD network was trained with techniques in machine learning (FORCE training [80, 81]) to cycle in a sequence around a ring with frequency θ_S where θ_S was 9 Hz. This cyclical behaviour acts as the disk rotation of the NDD model with disk rotations mapping to SWRs (Figure 2A). A single isolated cycle around the ring corresponds to a single isolated SWR with a duration of approximately 100 ms (${\theta_S}^{-1}$). This rotational sequence is generated by asymmetrically coupling the interneurons in the rotating disk network on the ring while the interneurons are receiving a super-threshold excitatory current (Figure 2B). The interneurons that are currently firing in the sequence inhibit interneurons that have fired just before, thereby maintaining the sequential firing structure on the ring. This interneuron ring serves as the rotator for the entire disk. Subsets of pyramidal neurons serve as individual tracks in the rotating disk network. If the collection of neurons within a track receive enough excitation, they can fire unique sequences, commonly elicited during hippocampal SWRs (Figure 2C-D). The excitation comes from recurrent coupling between the pyramidal neurons with a strongly coupled subset of track initiators (Materials and Methods). The initiator neurons bind the subset of pyramidal neurons into a track and are activated stochastically (consistent with [82]) while the sequences within a track are regulated by the interneuron rotator. These initiators may be related to high-firing rate, low spatial specificity CA1 pyramidal neurons [83, 84].

Next, we found that pyramidal cells in the rotating disk network that constitute at track can be exposed sector(s)-at-a-time when a second oscillation with a slower frequency ($\theta_{R/W} = 8$ Hz) is applied to the interneurons in the rotating disk network (Figure 2E). This frequency difference creates an interference that activates subsets of the full SWR-sequence of spikes (a track) in the rotating disk network but as subsets of a theta-sequence (Figure 2G-H). This is the slower operating mode where sectors can be accessed for reading/writing on a slower time-scale (Supplementary Video 3, Supplementary Video 4). We further observed that the interference between the θ_S and $\theta_{R/W}$ oscillations manifests as an interference pattern in the voltage of individual neurons (Figure 2) [85] and the interference produces internally generated theta sequences [25, 86, 87]. These internally generated theta sequences are used for reading or writing to tracks. Therefore, the rotating disk network accurately mimics both the operations of a spinning disk and the hippocampus as it consists of a series of pyramidal neurons arranged

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

into tracks and sectors, which could be accessed on slow (behavioural) or fast (neural) time scales, and an interneuron ring which "spins" the entire disk as the rotator thereby forming spike sequences.

With the rotating disk network constructed, we then focused on modelling the actuator arm (Figure 2J-L). The actuator arm of a conventional disk drive operates much like a continuous line attractor. The actuator itself produces a physical force that moves the arm to a specified position, thereby selecting a track (Figure 2J). Once the force on the arm stops, the arm stays in its new position over the track and is stable. In that way, all possible positions of the actuator arm are stable while an applied force can rapidly switch the arm's position, thereby forming a continuous line attractor.

Given the correspondence between an actuator arm and an attractor network, we trained a recurrent spiking network with FORCE training [80,81] to mimic the dynamics of a line attractor [88] to serve as the actuator arm network (Supplementary Figure S2, Figure 2K). The actuator arm network receives two inputs: the desired position (ψ_A) and velocity ($\frac{d\psi_A}{dt}$) of the actuator arm. We trained the actuator arm network to integrate the velocity-like input to estimate the desired position ($\hat{\psi_A}$). During training, the desired position, ψ_A was dropped during random periods to force the actuator arm network to produce a position estimate with integration alone (Supplementary Figure S2). After training, we found that the resulting actuator arm network produced isolated firing fields of pyramidal neurons, consistent with observed hippocampal place fields (Figure 2K-L, Supplementary Figure S3). However, here $\hat{\psi_A}$ refers to the estimated position of the actuator arm on the disk, rather than the estimated position of an animal in physical space *per se*. Further, we observed that pyramidal neurons in the actuator arm network exhibited bumps of activity, reporting the dynamics of an attractor network (Figure 2L) [85]. Finally, we tested the line-attractor nature of the trained actuator arm network. When no velocity inputs and position inputs are applied, the actuator network retains its last known position as a stable state, similar to how a physical actuator arm stays in the last position on a real HDD if the actuator no longer provides a force on the arm (Supplementary Figure S4). Thus, the actuator arm network behaves as both a line-attractor network, and the actuator arm of a disk drive, while simultaneously producing hippocampal features such as place-field like firing in the individual cells.

Next, we wondered how the actuator arm network displayed its line attractor dynamics. To investigate this, we sorted all neurons (pyramidal and interneuron) according to their place (ψ_A) preference (Supplementary Figure S3, Figure 2K). After sorting, we found that the actuator arm network maintained stable states through a self-disinhibitory motif. Pyramidal neurons with similar actuator arm position preferences excited both themselves and interneurons with similar ψ_A preferences. The interneurons then inhibited pyramidal neurons with dissimilar ψ_A preferences.

With the rotating disk and actuator arm networks created, we next investigated how information is written onto, and read from the hippocampal "tracks" in the rotating disk network network. For a physical disk drive, the read/write head on the actuator arm reads and writes bits of data (0's or 1's) on individual tracks as different magnetic field directions (e.g. $\uparrow$ or $\downarrow$) on a ferromagnetic material. We thus sought to identify how the read/write head network could "write" bits of information.

We thus mapped the functional capability of a R/W head in a disk drive to a plausible biological candidate: Hebbian plasticity (Figure 2D). We constructed a R/W spiking neural network that contains plastic excitatory synaptic weights from all pyramidal neurons in the RD network and uses these plastic weights to encode bits of information (Figure 2M). If enough excitatory weights from a track/sector connect onto a neuron in the R/W head, then that neuron spikes. The spiking of neurons in the read/write head (post-synaptic) paired with Hebbian plasticity to spikes in the rotating disk network (pre-synaptic) is how the read/write head network reads or writes bits of information.

Bits of information are sent to the R/W network from external sources for encoding, causing spiking in R/W pyramidal neurons. When the R/W neural spiking coincides with spiking from the active track in the RD network, Hebbian plasticity encodes bits of information in the AA$\rightarrow$R/W pathway. The bits of information are written when the spinning disk continuously cycles (θ_S), and the read/write oscillation ($\theta_{R/W}$) is active. As these oscillations sequentially expose the sectors of an individual track, bits are written discretely to isolated sectors (Figure 2N, Supplementary Video 3). We found that this discretization is visible as discrete assemblies when the information is subsequently replayed either when $\theta_{R/W}$ is active or in full disk rotations (Figure 2N-O, Supplementary Video 4) during SWRs, and in fact corresponds to the discrete sectors exposed during the theta-oscillation. This discretization of replays

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

may be linked to the observation that replays can also "jump" from location to location discretely [89]. This result shows that the read/write head in the Neural Disk Drive Model can use Hebbian plasticity in a similar fashion to the read/write head on a physical HDD to encode bits of information onto a plastic medium. Finally, we remark that it may also be possible that different phases of the $\theta_{R/W}$ oscillation are used separately for the reading and writing of bits [90–93].

2.3 The Tri-Partite Neural Disk Drive Model

Next, we investigated if the rotating disk, Actuator Arm, and read/write head networks that we constructed separately could be assembled together and function synergistically to store information as the final Neural Disk Drive (Figure 3). We simulated these three constituent networks coupled with three sets of cross-network synaptic weights (Figure 3A). The first set of (AA→RD) connections link the actuator arm and rotating disk networks, selecting tracks and thus subsets of pyramidal neurons to access. The second set of (RD→R/W) connections associated the RD network to the R/W head network, and act as the storage media to store bits of information (Figure 3B). The third set of (AA→R/W) connections, from the AA network to the R/W network, triggers synfire-chain like spiking in the R/W network. This sequence of spikes represents the information to be acquired during the writing state/theta oscillation, and replayed during the reading stage/SWR. These three (AA, RD and R/W) networks acting together collectively operate as the hippocampal Neural Disk Drive (Figure 3C-F). The R/W head network served to record/replay information as spike sequences; that is, leveraging the computer analogy, sequences of bits are written/read onto a track (Figure 3C). The track is rotated by the ring of interneurons that mediates the disk dynamics (Figure 3D). The actuator arm network serves to perform path-integration and selects the specific track on the rotating disk network.

Together, these networks operate synergistically. For example, we found that the AA network can create (theta) sequences longer than those contained within a single track by switching between tracks (Supplementary Figure S5). The AA network can also bias which track becomes replayed during disk rotation (Supplementary Figure S5). Finally, the zoo of hippocampal data-accessing events are well explained by a disk-drive model of hippocampal operations. By disabling the rotating disk network interneurons, replays can be converted into sequence-less reactivations [94] (Supplementary Figure S6). Pre-plays, where the sequences during SWRs are observed before navigation [49–53,95], are pre-existing or old data written to tracks before they are accessed during Reading/Writing phases (Supplementary Figure S6). Fragmented replays [6], where replay trajectories jump in state-space correspond to misalignment between the initial phase of the disk and the start of a bit sequence (Supplementary Figure S6, Supplementary Video 5). Reverse replays based on dedicated sequence-reversion pools of interneurons were previously considered [63]. In this scenario, the disk spins backwards to reverse the order of bit/spike sequences (Supplementary Video 6). Replays of trajectories that were not previously experienced by an animal ([96, 97]) can also be constructed by reading out multiple tracks that were never sequentially accessed during the writing phase. These tracks can also be individually reversed within a multiple-track replay event [96, 97].

Thus, the Actuator Arm, rotating disk, and read/write head networks work synergistically to record information in a tripartite Neural Disk Drive, with core features of the hippocampal network being mapped to well-defined operations and components of a HDD.

2.4 Detecting the Interneuron Ring Sequences Predicted by the NDD Model

This Neural Disk Drive model proposes that the hippocampus uses both network oscillations and attractor dynamics to implement the operations performed by the rotating disk and the actuator arm of a computer disk drive. Thus, we tested this theoretical prediction using empirical data, probing evidence for one of the dynamical hallmarks of the NDD model: interneurons serving as the rotational backbone of neural firing patterns. To proceed, we first considered in vivo hippocampal ensemble recordings performed from rats trained to learn and remember three reward locations on a cheeseboard maze (Materials and Methods, Figure 4A). On each day of this memory task, animals develop an effective navigation path to reach the reward locations throughout the learning trials (Figure 4B). We used these animal trajectories to analyze hippocampal patterns formed by sequences of spikes (Figure 4C-D) with a Maximal Likelihood Estimation (MLE) based alignment algorithm (see Supplementary materials). The

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

MLE algorithm aligns the repeating firing fields internally to each other across learning trials (rather than to place) by applying time-shifts across trials through each firing field (Figure 4D, Supplementary Figure S7). The time shifts are then used to optimize an objective function, the maximum-likelihood estimator for the spike density parameterized by said time shifts. We found that aligning a single neuron simultaneously aligns an entire community of neurons within the learning session (Figure 4E). This included both pyramidal neurons and interneurons (Figure 4F), unveiling a repeating interneuron sequence that appeared to be stable across learning trials (Figure 4G, Supplementary Figure S7). Interneurons in both mice and rats are known to have differing phase preferences with respect to the theta oscillation in the LFP. Indeed, we found a similar result (Figure 4H-K) with interneurons spanning the $[0, 2\pi]$ range in both rats (Figure 4H-I) and mice (Figure 4J-K). However, phase preference firing of bursts in interneurons during theta oscillations, which has been previously reported ([98–100]) does not generate ring-sequences alone. Indeed, stochastic simulations of neurons with phase preferential firing of bursts in a population will not generate a ring sequence (Supplementary Figure S8). Our analysis shows that that hippocampal interneurons form the neural ring-sequences predicted by the NDD model across species and tasks.

3 Discussion

Analogies and metaphors with devices have been attempted towards obtaining useful descriptions of the brain since antiquity [14, 16–18]. Often, these analogies are a product of their times, manifesting brain function as a likeness or similarity to the dominant technological innovation of the era. Indeed, the computer analogy for brain function was proposed by Von Neumann shortly after both the first electronic computer, ENIAC (Electronic Numerical Integrator and Computer), was unveiled [101] and Hodgkin and Huxley successfully modeled the squid giant axon [102]. However, these brain analogies are often limited to observations of behaviour, and rarely directly linking the computations, components, and function of a device to the anatomy and physiology of any particular brain area [15]. This is where our contribution differs from the past by leveraging decades of in vivo experiments in the hippocampus.

We have constructed and simulated a model of hippocampal function as a Neural Disk Drive, by merging two prior modelling paradigms, attractors [74–78] and interference models [20] into one singular device. The network was constructed and simulated with 3 components: an actuator arm (Attractor), a rotating disk (Oscillator) and a read/write Head (for storing memories). This model was sufficient to reproduce some of the core behaviours of hippocampal neurons; path-integration and navigation, theta sequences and phase precession, compressed sequences and sharp-wave ripples, and the zoo of hippocampal replays. Finally, we verified one of the predictions of this model; a rotating ring-sequence of interneurons which constitute the rotator of the neural disk drive.

If the hippocampus does utilize oscillations and attractor dynamics similarly to how these dynamics are used in a disk drive, two natural lines of inquiry emerge. The first line of inquiry emerges from the saturation of tracks. A track on a physical disk drive can only hold so many bits. In fact, for a sufficiently large file, multiple tracks must be used to store the entire file (Supplementary Video 7). Thus, if the hippocampus stores information as a disk drive, does multi-track storage also occur? Evidence for the affirmative to this hypothesis is present in the literature in the form of so-called extended replays, where a trajectory is replayed as multiple, sequential sharp-wave-ripple complexes [28, 29]. These ripple complexes are separated by 150 ms intervals, with an inter-sharpwave-interval distribution displaying prominent peaks at multiples of 150 ms, hinting at a disk-rotation mechanism controlling SWR generation [63].

The second line of inquiry consists of the total storage space of the hippocampus, if it does indeed operate as a disk drive. In the NDD model, the storage medium used to write bits onto the RD network are the synaptic weights coupling the RD network onto the R/W head. Anatomically, these weights may be the Schaeffer Collateral connections linking CA3 to CA1, which constitutes a matrix containing at most $N_{CA3}N_{CA1}$ connections, where N_{CA3}, N_{CA1} are the number of neurons in CA3, CA1. If we think of the weights as entirely binary, and consider reasonable values of $N_{CA3} = N_{CA1} = O(10^5)$ [103], we arrive at $\approx O(10^9)$ bytes or $O(1)$ GB of storage as an upper-bound for rats. Human estimates for N_{CA3} and N_{CA1} are larger ($O(10^6)$), leaving $O(10^2)$ GB [104]. Thus, if the hippocampus does act as a disk drive, it is one of fairly limited storage, as 100 GB translates into roughly 100 hours

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

of video and audio with 720p resolution. Additionally, these were upper bounds that were derived assuming all CA3 and CA1 neurons and that the neurons operate perfectly to transmit and store bits without redundancy. Given the low values of even these optimistic bounds, the neural disk drive model suggests that the hippocampus is limited in the amount of data it can store, possibly hinting that the hippocampus stores some type of data-compressed representation of complex memories [105–109]. Alternatively, it is also possible that the low amounts of storage are only used to store recent events, prior to consolidation [110–112].

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

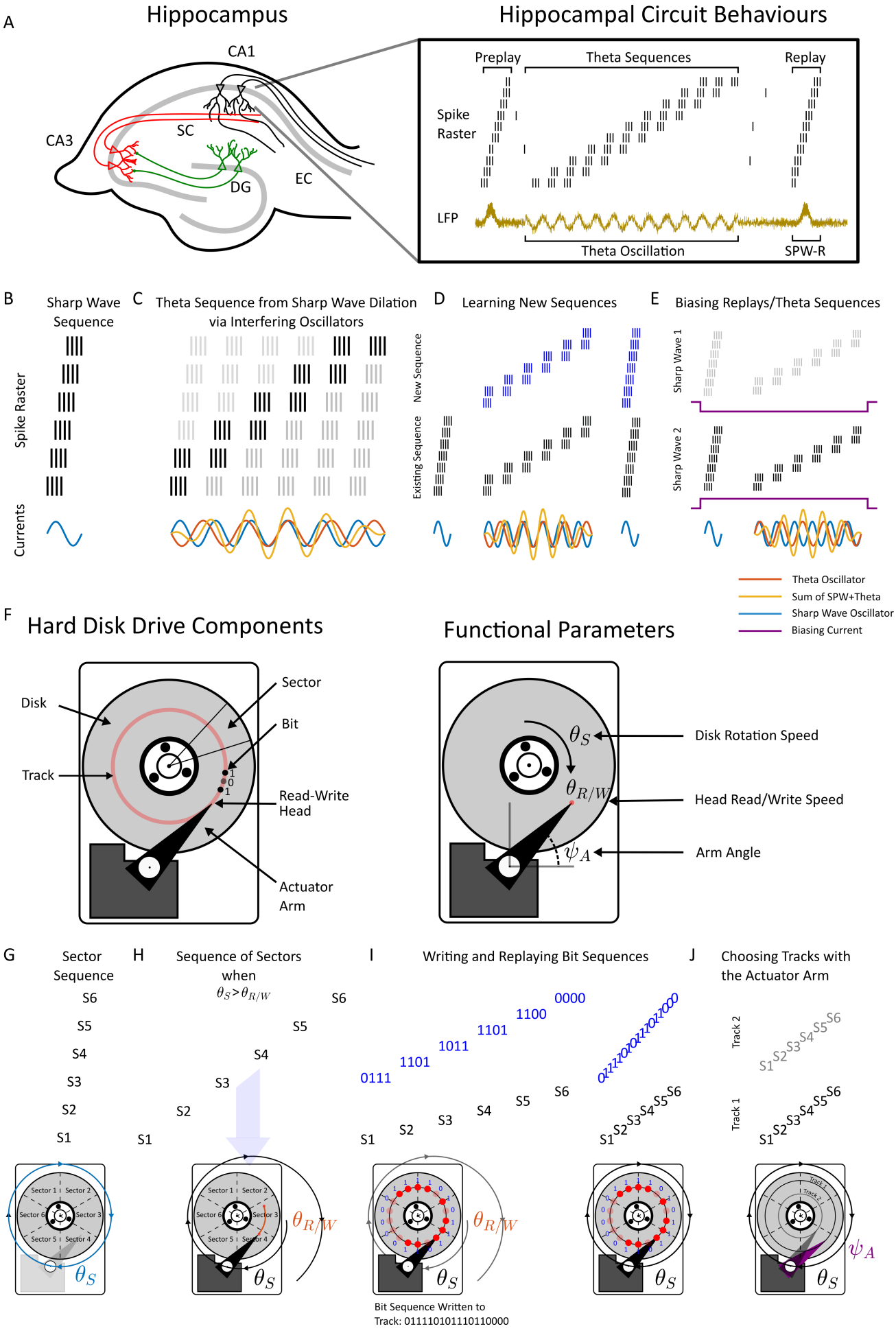


Figure 1

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

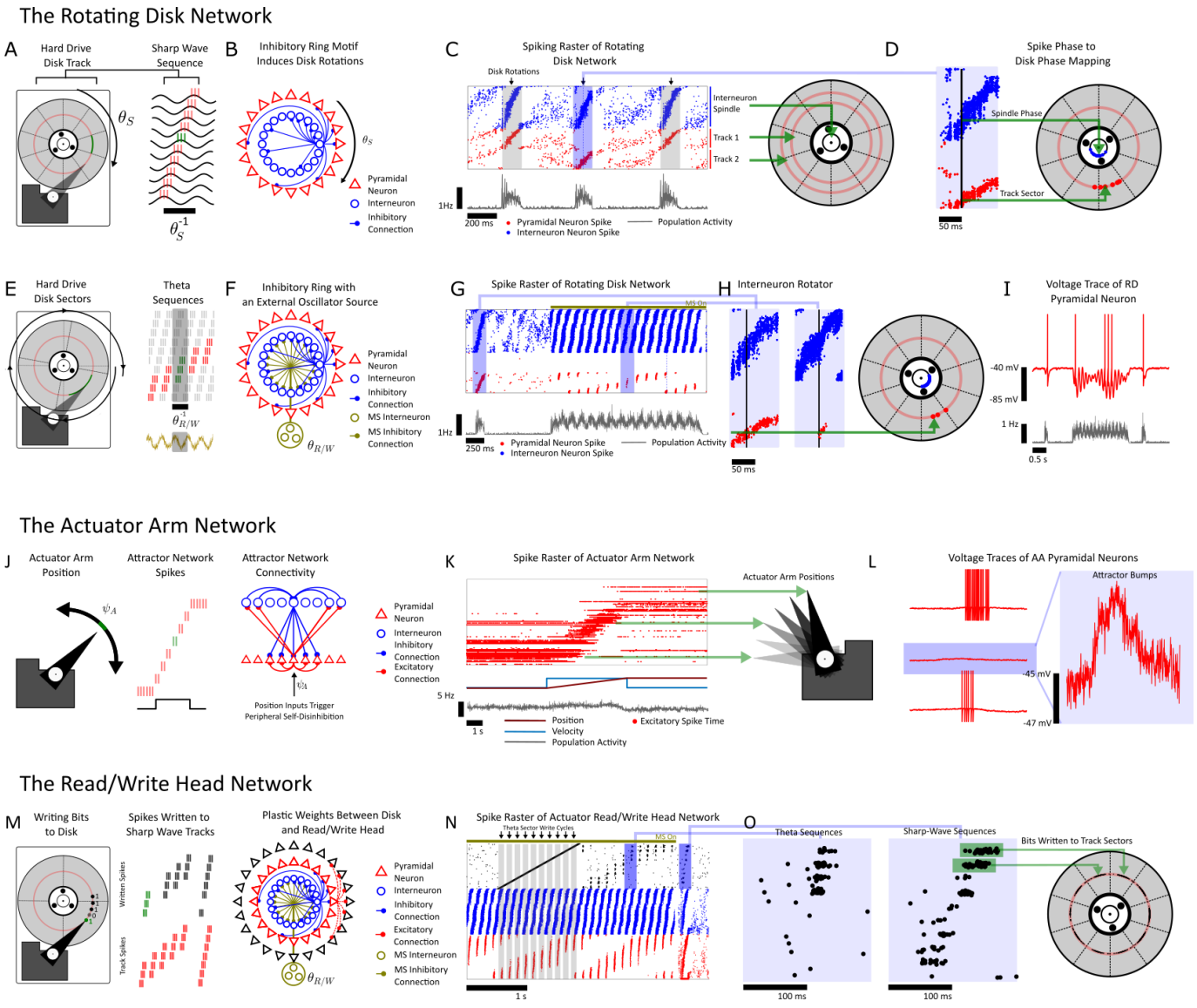


Figure 2

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

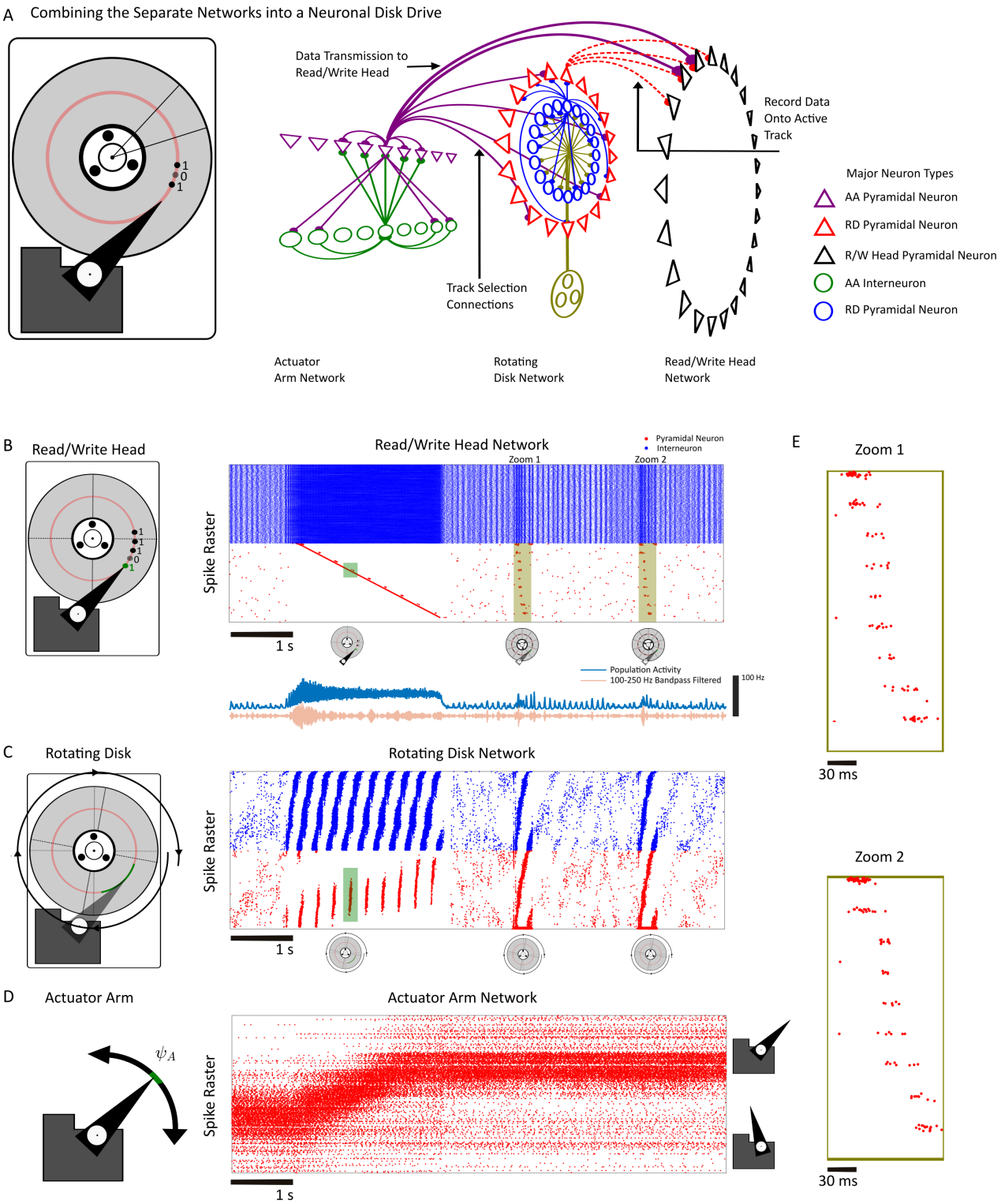


Figure 3

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

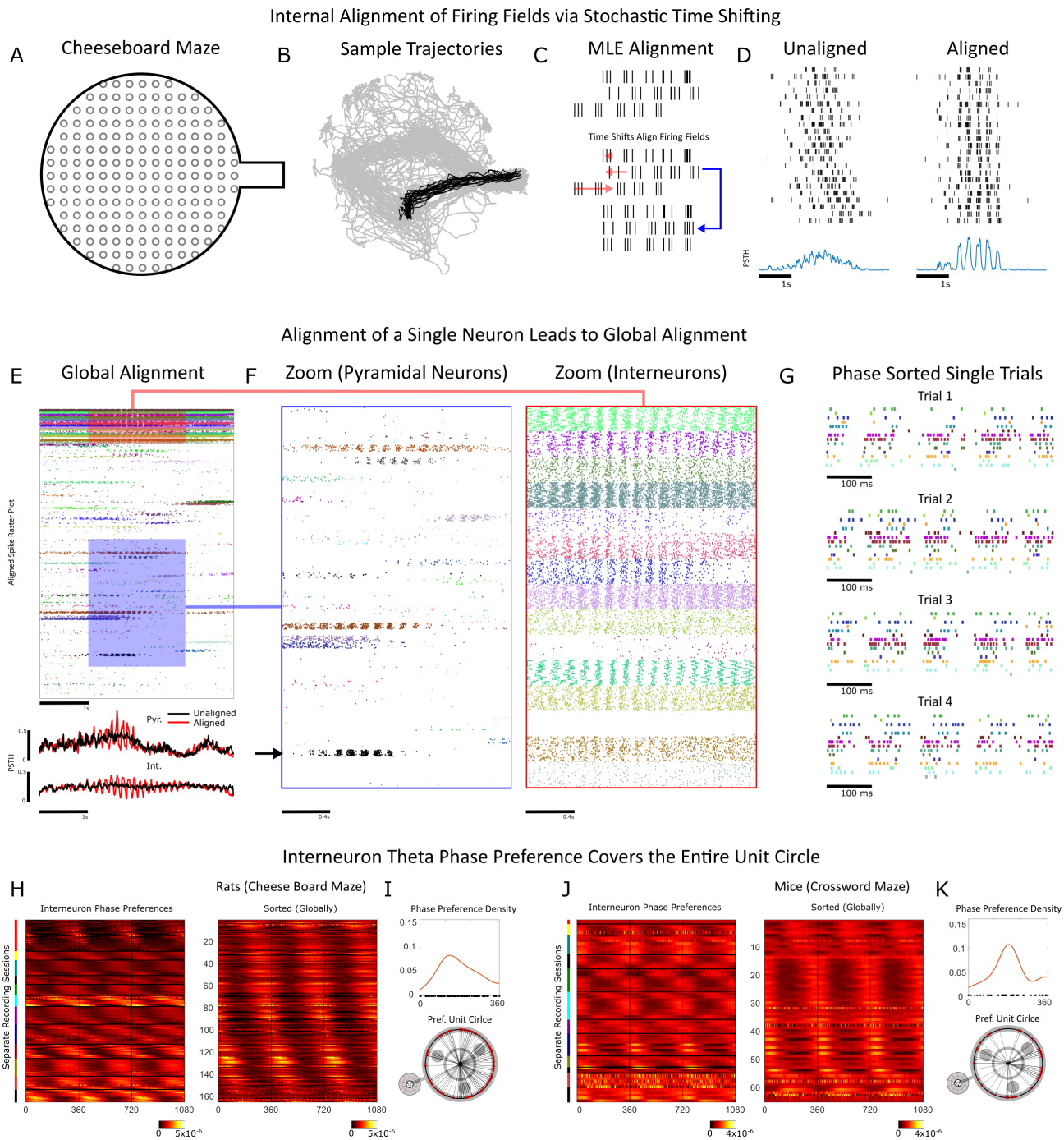


Figure 4

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

List of Supplementary Videos

To dynamically describe both the function of disk drives and the relation between hippocampal dynamics and disk drives, a series of animations were prepared (see below). All animations can be viewed and downloaded from https://www.nicolacomputationalneurosciencelab.com/publications.

Supplementary Video 1: The Components and Operations of a Disk Drive. A supplementary video that displays how the motions of the read/write head, rotating disk, and actuator arm stores and accesses information.

Supplementary Video 2: Writing on the Slow Time Scale. A supplementary video that shows how a disk drive can write with a frequency ($\theta_{R/W}$) that is slightly slower than the disk rotation speed (θ_S). This frequency difference creates hippocampal phase precession in the Neural Disk Drive model.

Supplementary Video 3: Replays via Single Disk Rotations. A supplementary video that shows how a trajectory can be replayed in the Neural Disk Drive model with a single disk rotation.

Supplementary Video 4: Theta Sequences via Multi-Sector Access. A supplementary video that shows the emergence of theta-sequences from a tracks worth of data. Here, the read/write head accesses multiple sectors in a single disk rotation sequentially, thereby creating theta-sequences with phase precessing spikes.

Supplementary Video 5: Fragmented Replay via Disk-Phase Misalignment. A supplementary video showing how trajectories can "jump" via the misalignment of the initial phase or sector of the disk with the start of a trajectory.

Supplementary Video 6: Reverse replay via Counter-Clockwise Disk Rotations. A supplementary video showing how trajectories can be replayed backwards through time by spinning the disk in the opposite rotation (e.g. counter-clockwise) of the initial recording (e.g. clockwise).

Supplementary Video 7: Extended Replay via Multi-Track Access. A supplementary video showing how long trajectories can be decoded by using multiple tracks, with an actuator arm switching between successive tracks.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Figure Captions

Figure 1: The links between hippocampal dynamics and disk drive operations

(A) The anatomical (CA3, CA1, Dentate Gyrus (DG), and Entorhinal Cortex (EC) sub-regions of the hippocanmpus (left) along with observed hippocampal behaviours (right). Sequences of spikes occur on short time scales as preplays/replays during hippocampal sharp-wave-ripples (SWRs), and on longer time scales as theta sequences. The components of a theta sequence during a theta oscillation are subsets of the entire preplay/replay. Preplays occur before the initial observation of a theta sequence, while replays occur after the initial observation of a theta sequence. The theta oscillation is an 8-12 Hz oscillation displayed in the local-field potential (LFP) while SWRs are 150-250 Hz high-frequency oscillations in CA1, which coincide with a large-deflection (the sharpwave) in CA3. **(B)** In a model of hippocampal replay, sharp-wave sequences are controlled by an intra-hippocampal oscillation. The oscillation is transiently activated for single cycles to trigger SWR replays. **(C)** By adding a secondary oscillation, SWR sequences become temporally dilated into theta sequences via an interference-based mechanism. **(D)** Dilated sharp-wave spike sequences can form a basis to learn new sequences, in a one-shot, instantly compressible format. **(E)** Different sharp-wave sequences can be elicited by a biasing current to pools of pyramidal neurons. **(F)** The hardware components (left) and functional parameters (right) of a computer disk drive, which operates by encoding data onto a rapidly spinning disk subdivided into tracks and sectors. The read/write head on the apex of the actuator armmoves to encode bits onto different tracks. The disk drive has three functional parameters: the Disk Rotation speed θ_S, the read/write speed of the head $\theta_{R/W}$, and the actuator armangle ψ_A.

Figure 2: Mapping hippocampus circuit components onto hard disk drive components: the rotating disk, actuator armand read/write head networks.

(A) The rotating disk (RD) network contains SWR sequences that instantiate tracks in the Neural Disk Drive (NDD) model of the hippocampus function. **(B)** An asymmetric ring of inhibition induces a rotating SWR sequence with rotational speed θ_S. **(C)** Different subsets of RD excitatory neurons instantiate tracks in the NDD model. **(D)** The interneuron ring acts as the primary rotator of the NDD, while subsets of the RD SWR tracks act as the sectors. **(E)**The theta oscillations $\theta_{R/W}$ sequentially expose sectors of a track to write-to. This forms theta sequences, which in the NDD model, are sequentially exposed tracks of a SPR track. **(F)** An oscillatory inhibitory signal with a frequency of $\theta_{R/W}$ triggers sector exposure during a theta oscillation. **(G)** During a RD SWR, all sectors of a track are activated without $\theta_{R/W}$. During theta oscillations (with $\theta_{R/W}$), the sectors are sequentially activated. **(H)** Zoom of the spike sequences in the RD network overlaid explicitly onto a disk. **(I)** RD dynamics are present as theta oscillations in both the LFPs and in the membrane potential of individual pyramidal neurons. **(J)** The actuator arm(AA) features dynamics of an attractor network. A torque on the AA causes it to change angular position ψ_A, which activates different subsets of AA neurons. This forms sequences of instantaneous rates, rather than individual spikes. When the inputs correspond to velocity and place, these ongoing firing fields can be interpreted as current place fields. **(K)** A simulation of the FORCE trained AA network with the position and velocity of the animal as inputs (below). The AA pyramidal neurons (red) are sorted according to their place preference. **(L)** The voltage traces for 3 randomly selected pyramidal neurons in the AA network. The voltage traces show "ramps of activity", which allows the neurons to modulate their ongoing firing rates to collectively encode the position of the actuator arm. The position of the actuator arm can be decoded from a linear combination of these rates. **(M)** Once a track is selected by the AA, the read/write (R/W) head can write new sequences onto the track. This corresponds to updating the Schaffer collateral weights in the network to store new sequential information. **(N)** Spike raster plot for the RD and R/W head networks. The written sequences become discretized into assemblies of coactive neurons **(O)** The discretized assemblies projected onto the sectors of a disk.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Figure 3: Assembling the tripartite Neural Disk Drive model.

(A) The three constituent (RD, AA, and R/W) networks are connected together to construct a tripartite network. **(B)** The RD, AA and R/W networks correspond to discrete regions and pathways within the hippocampus circuit. **(C)** The R/W network is simulated with a writing component consisting of theta oscillations ($\theta_{R/W}$) and a reading phase during which SWRs occur (θ_S). **(D)** The RD network during the writing phase and two disk rotations. The interneuron ring performs the disk rotations in both the writing and reading phase. **(E)** The AA network, with neurons sorted according to place preference. The AA transitions from one location to the next. **(F)** Two zooms of the replayed sequences that occurred during RD SWRs.

Figure 4: Probing the hippocampal in-silico NDD with in-vivo ensemble recordings.

(A) Schematics of the cheeseboard and the crossword mazes where rodents (rats and mice, respectively) learn to navigate to reward locations while hippocampal neuron ensembles and LFPs are recorded. **(B)** Example repeated navigational trajectories (black) used for firing pattern analyses, superimposed on animal's whole path (gray) in these mazes. **(C)** The trajectories are aligned with a maximum likelihood estimation (MLE) algorithm that maximizes the probability of recorded spiking with time-shifts to each temporal firing field. **(D)** The firing fields, temporally unaligned (left) and aligned (right). **(E)** Aligning a single recorded pyramidal neuron produces a global temporal alignment across the ensemble of multiple recorded neurons. **(F)** Zoom of the aligned recorded pyramidal neurons (left) and interneurons (right). The black arrow denotes the pyramidal neuron used to align the entire population. **(G)** A sequence of interneuron ring-like spiking. **(H)** Temporal firing fields for a population of interneurons. **(I)** The phase preference of firing for all interneurons is broadly distributed. **(J)** Similarly, the firing fields for a population of interneurons recorded in mice performing another memory task (the crossword maze). **(K)** The phase preference for all interneurons is once again, broadly distributed over $[0, 2\pi]$.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Materials Availability

This study did not generate new unique reagents.

Data availability

The datasets used in this will be made available via the MRC BNDU Data Sharing Platform (https://data.mrc.ox.ac upon reasonable request.

Funding

W.N. and D.D are supported by the New Frontiers in Research Fund Canada (Award NFRFE-2019-00159). D.D is supported by the Biotechnology and Biological Sciences Research Council (BBSRC) UK (Award BB/S007741/1) and the Medical Research Council UK (Awards MC_UU_12024/3, MC_UU_00003/4 and MR/W004860/1). C.C. is supported by the BBSRC (BB/N013956/1), Wellcome Trust (200790/Z/16/Z), the Simons Foundation (564408) and EPSRC (EP/R035806/1).

Materials and Methods

The Actuator Arm Network

The actuator arm network consists of 2000 excitatory (AA_E), and 2000 inhibitory (AA_I) leaky-integrate-and-fire neurons:

$$\begin{aligned}\tau_m \frac{d\dot{v}_j^{AA_E}}{dt} &= -v_j^{AA_E} + RI^{AA_E} + R\sum_{j=1}^{N_E^{AA}} \omega_{ij}^{AA_E,AA_E} r_j^{AA_E}(t) + R\sum_{j=1}^{N_I^{AA}} \omega_{ij}^{AA_E,AA_I} r_j^{AA_I}(t) \\ &+ \kappa_j^{AA_E}\psi_A(t) + \gamma_j^{AA_E}\frac{d\psi_A(t)}{dt}, \quad j = 1, 2, \dots N^{AA_E} \end{aligned} \tag{1}$$

$$\begin{aligned}\tau_m \frac{d\dot{v}_j^{AA_I}}{dt} &= -v_j^{AA_I} + RI^{AA_I} + R\sum_{j=1}^{N_E^{AA}} \omega_{ij}^{AA_I,AA_E} r_j^{AA_E}(t) + R\sum_{j=1}^{N_I^{AA}} \omega_{ij}^{AA_I,AA_I} r_j^{AA_I}(t) \\ &+ \kappa_j^{AA_I}\psi_A(t) + \gamma_j^{AA_I}\frac{d\psi_A(t)}{dt}, \quad j = 1, 2, \dots N^{AA_I} \end{aligned} \tag{2}$$

where v^{AA_E} and v^{AA_I} denotes the voltage for an actuator arm excitatory and actuator arm inhibitory neuron, respectively. The parameters for all neurons/synapses can be found in Table 1. Once the voltage for a neuron reaches a threshold, v_{thresh}, the voltage is reset to v_{reset}.

$$v^{AA_{E/I}}(t^-) = v_{thresh}, \quad v^{AA_{E/I}}(t^+) = v_{reset} \tag{3}$$

Every spike is followed by an absolute refractory period, τ_{ref} during which the neuronal dynamics are quenched at the reset value. The parameters $I^{AA_E/I}$, denote the bias currents to the neurons in the E/I sub-populations, respectively. The membrane time constant, τ_m controls the integration dynamics of each neuron. The parameter $R = 1 \cdot 10^9\ \Omega$ serves as the resistance. The weight matrices $\boldsymbol{\omega}^{AA_E,AA_E}$,$\boldsymbol{\omega}^{AA_E,AA_I}$,$\boldsymbol{\omega}^{AA_I,AA_E}$, $\boldsymbol{\omega}^{AA_I,AA_I}$ denote the coupling from E to E, I to E, E to I and I to I populations, respectively. These weights are trained with the FORCE algorithm, and described below. The inputs $\psi_A(t)$ and $\frac{d\psi_A}{dt}$ denote the desired position of the actuator arm, and the velocity of the desired position of the actuator arm. The inputs are multiplied by a set of input weights, $\boldsymbol{\kappa}^{AA_{E/I}}$, and $\boldsymbol{\gamma}^{AA_{E/I}}$ for the actuator arm position/velocity, respectively.

The variables $\boldsymbol{r}^{AA_{E/I}}$ are the convolved spike times for the E/I actuator arm neurons:

$$\dot{r}_j^{AA_{E/I}} = -\frac{r_j^{AA_{E/I}}}{\tau_d} + h_j^{AA_{E/I}} \tag{4}$$

$$\dot{h}_j^{AA_{E/I}} = -\frac{h_j^{AA_{E/I}}}{\tau_r} + \frac{1 \cdot pA \cdot ms^2}{\tau_r \tau_d} \sum_{t_{jk}^{AA_{E/I}} < t} \delta(t - t_{jk}^{AA_{E/I}}) \tag{5}$$

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

where $t_{jk}^{AA_{E/I}}$ denotes the kth spike fired by the jth neuron in the actuator arm excitatory/inhibitory sub-population. The parameters τ_d and τ_r denote the decay and rise times, respectively of the sub-population of neurons. The AA network, and indeed, all networks considered were integrated with a simple Forward-Euler method and a step size of $dt = 5 \times 10^{-5} s$.

FORCE Training the Weights of the Actuator Arm Network

The weight matrices $\boldsymbol{\omega}^{AA_E,AA_E}$,$\boldsymbol{\omega}^{AA_E,AA_I}$,$\boldsymbol{\omega}^{AA_I,AA_E}$, $\boldsymbol{\omega}^{AA_I,AA_I}$ decompose as the sum of a static component, and a learned component:

$$\boldsymbol{\omega}^{AA_E,AA_E} = \boldsymbol{\omega}_0^{AA_E,AA_E} + (\boldsymbol{\eta}^E)_+(\boldsymbol{\phi}^E)_+ + (\boldsymbol{\eta}^E)_-(\boldsymbol{\phi}^E)_- \quad (6)$$

$$\boldsymbol{\omega}^{AA_E,AA_I} = \boldsymbol{\omega}_0^{AA_E,AA_I} + (\boldsymbol{\eta}^E)_-(\boldsymbol{\phi}^I)_+ + (\boldsymbol{\eta}^E)_+(\boldsymbol{\phi}^I)_- \quad (7)$$

$$\boldsymbol{\omega}^{AA_I,AA_I} = \boldsymbol{\omega}_0^{AA_I,AA_I} + (\boldsymbol{\eta}^I)_+(\boldsymbol{\phi}^I)_- + (\boldsymbol{\eta}^I)_-(\boldsymbol{\phi}^I)_+ \quad (8)$$

$$\boldsymbol{\omega}^{AA_I,AA_E} = \boldsymbol{\omega}_0^{AA_I,AA_E} + (\boldsymbol{\eta}^I)_+(\boldsymbol{\phi}^E)_+ + (\boldsymbol{\eta}^I)_-(\boldsymbol{\phi}^E)_-. \quad (9)$$

The functions

$$(x)_+ = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}, \quad (x)_- = \begin{cases} x, & x \leq 0 \\ 0, & x > 0 \end{cases}$$

are applied to the components of the matrices $\boldsymbol{\eta} = [\boldsymbol{\eta}^E, \boldsymbol{\eta}^I]$, $\boldsymbol{\phi} = [\boldsymbol{\phi}^E, \boldsymbol{\phi}^I]$ to enforce Dales Law. The matrices $\boldsymbol{\eta}$ are referred to as the encoders, and help determine the tuning properties of neurons with respect to the estimated actuator arm position $\hat{\psi}_A$. For each neuron in the AA network, the encoder for that neuron (a row of $\boldsymbol{\eta}$), is randomly generated and sparse. The encoder is an $N \times k$ matrix where n_{sup} is the dimension of the supervisor of the network, likewise for the decoder. The encoder is enforced to be sparse: each neuron has a single element in its encoder that is non zero, and randomly set to $\pm W$, where $W = 10$pA. The input weights, $\boldsymbol{\kappa}^{\psi_A}$ and $\boldsymbol{\kappa}^{\gamma}$ were randomly generated with from a uniform $[-1, 1]$ distribution, for all neurons in the AA network.

The decoders, $\boldsymbol{\phi}$ are learned with FORCE training, which we describe below.

The Supervisor and Inputs to the Actuator Arm Network

The input to the actuator arm network is a randomly generated signal, $\psi_A(t)$, and its derivative, $\frac{d\psi_A}{dt}$. The signal is generated with a bounded, double-filtered noisy process. The first filter corresponds to the acceleration $a(t)$ while the second filter corresponds to the velocity $\psi_A(t)$:

$$\tau_\psi \frac{da}{dt} = -a + \zeta(t) \quad (10)$$

$$\tau_\psi \frac{d\psi_A}{dt} = -\psi_A + a(t) \quad (11)$$

where $\psi_A(t) = \pm 1$ causes a reset, as if colliding with a boundary, to the velocity and acceleration of the actuator arm position $\frac{d\psi_A}{dt} = 0$, $a(t) = 0$. Further, the velocity of the actuator arm is also limited such that if $|\frac{d\psi_A}{dt}| \geq 1$, the velocity is fixed to ± 1, to prevent arbitrarily fast motion of the actuator arm. The variable $\zeta(t)$ is a white noise process with mean 0 and standard deviation of 10^{-3}. The actuator arm position is unit-less, while the velocity and acceleration are s^{-1} and s^{-2}.

The position and velocity inputs $(\psi_A(t), \frac{d\psi_A}{dt})$ are provided to the actuator arm network during training. However, the position component is dropped stochastically (set to 0) for intervals that are randomly generated. These intervals are a minimum of 1 second long, and a maximum of 51 seconds long, with the interval itself drawn from the uniform distribution $U([1, 51])$. Once the position dropping interval ends, the position is turned back on instantly, for a random period of time. This random interval with position is also uniformly generated from the distribution $U([1, 51])$. In the intervals where the position is dropped, the network must rely on velocity and the last known position of the system to "integrate" and estimate the desired actuator arm position.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

The supervisor, $\boldsymbol{s}(t)$ to the actuator arm network is a $n_{sup} = 200$ dimensional vector that is a non-linear transform of the position:

$$s_i(t) = \exp\left(-\frac{(x_i - \psi_A(t))^2}{\sigma_x}\right) \tag{12}$$

Each component of the supervisor acts as an activity bump when $\psi_A(t)$ passes near x_i, where x_i is the center of the bump. The centers are uniformly distributed on the interval $[-1, 1]$. The variable σ_x controls the width of the bump, with $\sigma_x = 0.3$ used during training. The velocity component is not contained in the supervisor in any way.

Recursive Least Squares

The decoders, $\boldsymbol{\phi}$ are determined dynamically to minimize the squared error between the approximant and intended dynamics, $\boldsymbol{e}(t) = \hat{\boldsymbol{s}}(t) - \boldsymbol{s}(t)$. The Recursive Least Squares (RLS) technique updates the decoders to solve this problem in real-time:

$$\boldsymbol{\phi}(t) = \boldsymbol{\phi}(t - \Delta t) - \frac{\boldsymbol{e}(t)\boldsymbol{P}(t - \Delta t)\boldsymbol{r}(t)}{1 + \boldsymbol{r}(t)^T\boldsymbol{P}(t - \Delta t)\boldsymbol{r}(t)} \tag{13}$$

$$\boldsymbol{P}(t) = \boldsymbol{P}(t - \Delta t) - \frac{\boldsymbol{P}(t - \Delta t)\boldsymbol{r}(t)\boldsymbol{r}(t)^T\boldsymbol{P}(t - \Delta t)}{1 + \boldsymbol{r}(t)^T\boldsymbol{P}(t - \Delta t)\boldsymbol{r}(t)} \tag{14}$$

and $\boldsymbol{r}(t) = (\boldsymbol{r}^{AA_E}(t), \boldsymbol{r}^{AA_I}(t))^T$. The Recursive Least Squares Algorithm and FORCE training is described in greater detail in [80, 81]. The actuator arm network is initialized with $\boldsymbol{\phi}(0) = \boldsymbol{0}$, $\boldsymbol{P}(0) = \boldsymbol{I}_N\lambda$, where $\boldsymbol{I}_N$ is an N-dimensional identity matrix, and λ controls the learning rate of RLS. The value $\lambda = 0.5dt$ was used, where $dt = 5 \times 10^{-5}s$ was the simulation integration step size. To implement Dale's law as in equations (6)-(9), we decompose $\boldsymbol{\phi}$ into $\boldsymbol{\phi}^E$ and $\boldsymbol{\phi}^I$ with $\boldsymbol{\phi}^E = (\boldsymbol{\phi})_+$ and $\boldsymbol{\phi}^I = (\boldsymbol{\phi})_-$. The training parameters for the RD network. For the rotating disk network, a value of $\lambda = 0.05dt$ was used.

The Rotating Disk Network

The rotating disk network is a modification of the so-called "SHOT-CA3" network from [63]. As in the actuator arm network, the rotating disk network consists of coupled leaky integrate-and-fire neurons:

$$\begin{aligned}\tau_m\dot{v}_i^{RD_I} &= -v_i^{RD_I} + RI^{RD_I} + R\sum_{j=1}^{N_{RD_I}}\omega_{ij}^{RD_I,RD_I}r_j^{RD_I}(t) \\ &+ RI^{GABA}(1 + \cos(2\pi\theta_{RW}t)), \quad i = 1, 2, \ldots N_{RD_E}\end{aligned} \tag{15}$$

$$\tau_m\dot{v}_i^{RD_E} = -v_i^{RD_E} + RI^{RD_E} + R\sum_{j=1}^{N_{RD_I}}\omega_{ij}^{RD_E,RD_I}r_j^{RD_I}(t), \quad i = 1, 2, \ldots N_{RD_I} \tag{16}$$

where RD_E, RD_I denote the excitatory and inhibitory populations of the rotating disk network. The neurons receive a constant background current I^α for $\alpha = RD_E, RD_I$. The RD_I neurons receive an oscillatory input where θ_{RW} is the input frequency, and $\kappa > 1$ determines the tonic level of inhibitory drive. The INP-MS has amplitude $I^{GABA} = -10$ pA for $i = 1, 2, \ldots N_I$. The network consists of $N_{RD_E} = N_{RD_I}$ spiking neurons.

The weight matrices $\boldsymbol{\omega}^{RD_I,RD_I}$, $\boldsymbol{\omega}^{RD_E,RD_I}$, and supervisor used to generate them with FORCE training are described in further detail below, and in the specific methods for individual figures. They decompose similarly to the actuator arm network weights in equations (6)-(9). The supervisor used to train the RD network is a bank of oscillators:

$$s_{RD}(t) = \cos(2\pi\theta_S t + \beta_i^{RD}) \tag{17}$$

where ϕ_i is a uniformly distributed $[0, 2\pi]$ phase preference for each oscillator. The phase preferences β_i^{RD} are randomly drawn from a uniform distribution on the interval $[0, 2\pi]$. All weight matrices that we consider are dimensionless with the units of current (pA) carried by the synaptically filtered spike trains $\boldsymbol{r}(t)$ (see Equation (5)).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

The Read/Write Head Network

The read/write Head network is a population of leaky integrate-and-fire 2000 inhibitory (RW_I) and 2000 excitatory (RW_E) leaky-integrate-and-fire neurons:

$$\begin{aligned}
\tau_m \dot{v}_i^{RW_I} &= -v_i^{RW_I} + RI^{RW_I} + R\sum_{j=1}^{N_{RW_E}} \omega_{ij}^{RW_I,RW_E} r_j^{RW_E}(t) + R\zeta_i(t), \quad i = 1, 2, \dots N_{RW_I} \\
\tau_m \dot{v}_i^{RW_E} &= -v_i^{RW_E} + RI^{RW_E} + R\sum_{j=1}^{N_{RW_I}} \omega_{ij}^{RW_E,RW_I} r_j^{RW_I}(t) \\
&+ R\sum_{j=1}^{N_{RD_E}} \omega_{ij}^{RW_E,RD_E} r_j^{RD_E}(t) + RI_{syn,i}^{RW_E,bits}(t), \quad i = 1, 2, \dots N_{RW_E}
\end{aligned}$$

where $\zeta_i(t)$ is an independent white noise term with mean 0 and standard deviation $\sigma = 0.2$ pA. This noise term prevents the pathological synchronization of interneurons. The weight matrices $\boldsymbol{\omega}^{RW_E,RW_I}$, $\boldsymbol{\omega}^{RW_E,RW_I}$ are untrained, and described below. The bits sent to the RW_E neurons are expressed as time dependent currents $I_i^{RW_E,bits}(t)$. Finally, the weight matrix $\boldsymbol{\omega}^{RW_E,RD_E}$. The matrix $\boldsymbol{\omega}^{RW_E,RW_I}$ is a random matrix with each element drawn from a uniform distribution $[W_{RW_E,RE_I}, 0]$ where $W_{RW_E,RE_I} = -\frac{0.04}{N_{RW_I}}$. The matrix $\boldsymbol{\omega}^{RW_I,RW_E}$ is also randomly generated, on the interval $[0, W_{RW_I,RW_E}]$ where $W_{RW_I,RW_E} = \frac{25}{N_{RW_E}}$.

The weights from the rotating disk excitatory neurons to the read/write head excitatory neurons are learned with a Hebbian-plasticity based learning rule ([63]):

$$\Delta\boldsymbol{\omega}^{RW_E,RD_E} = \epsilon \boldsymbol{r}^{RW_E(t)} \left(\boldsymbol{r}^{RD_E}(t)\right)^T \tag{18}$$

For efficiency in the numerical simulations, the update rule (18) is applied every 15 dt time steps, rather than every time step. The parameter ϵ acts as a learning rate for the synaptic weight adjustments and controls how rapidly the weights are adjusted.

Maximum Likelihood Alignment

Identifying Repeating Trajectories

To maximally align the spike times, we first selected a trajectory component from the rats navigating the cheeseboard maze from [38], which was restricted to 4 seconds in duration and contained large, linear movements along the maze within those 4 seconds. The initial selected trajectory $\boldsymbol{p}_0 = (x_0(t), y_0(t))$, for $t \in [t_0, t_0 + 4]$ was then used as a motion template, where other $\boldsymbol{p}_j = x_j(t), y_j(t)$, $t \in [t_j, t_j + 4]$ were found by proximity to the initial template via the L_2 norm:

$$d_j = F(t_j) = \|p_j - p_0\|^2 \tag{19}$$

The trajectory components were found by treating t_j as a continuous variable, τ and floating τ over the entire interval, τ:

$$F(\tau) = \int_0^4 \left[(x(t+\tau) - x_0(\tau))^2 + (y(\tau) - y_0(\tau))^2\right] dt$$

Then, a peak detector algorithm (*findpeaks*, MATLAB 2020a), was used to detect local minima in $F(\tau)$ (maxima in -$F(\tau)$). Only the top 70% of these peaks were used, as $F(\tau)$ may contain local minima that are dissimilar from the initial trajectory x_0. The set of minima of $F(\tau)$, correspond to the discrete times $\tau_1^*, \tau_2^*, \dots \tau_m^*$.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Parameter Value	RD-E	RD-I	AA-E	AA-I	R/W-E	R/W-I
N	2000	2000	2000	2000	4000	4000
t_{ref}	2 ms	2 ms	2 ms	2 ms	2 ms	2 ms
t_m	10 ms	10 ms	10 ms	10 ms	10 ms	10 ms
I^{α}		-25 pA/-40 pA	-40 pA	-40 pA	-41 pA/-42 pA	-40 pA
v_{reset}	-65 mV	-65 mV	-65 mV	-65 mV	-65 mV	-65 mV
$v_{threshold}$	-40 mV	-40 mV	-40 mV	-40 mV	-40 mV	-40 mV
τ_d	20 ms	20 ms	20 ms	20 ms	20 ms	20 ms
τ_r	2 ms	2 ms	2 ms	2 ms	2 ms	2 ms

Table 1: The parameters used for the rotating disk (RD), actuator arm (AA) and read/write (R/W) Head networks, unless otherwise specified by the supplementary methods for each figure. The bias currents, I^{α}, $\alpha = RD, AA, R/W$ vary to change the operational modes of the different sub-networks. Note that the nominal values of the bias currents may differ in specific figures/subfigures.

Performing the Maximum Likelihood Alignment

With the trajectory alignment times τ_j^*, $j = 1, 2, \ldots m$ determined, a single neuron was selected for target alignment. The spikes for the m trials of that neuron were selected, and a kernel density estimator was constructed, $\rho_{ner}(t)$,where $t \in [0, 4]$. The bandwidth of the Gaussian kernel was taken to be 5 milliseconds. Then, for the m trials, a random $m \times 1$ vector of m time shifts was generation. Each scalar component of this vector would would shift all the spikes within one of the $j = 1, 2, \ldots m$ trials by a constant amount, τ_j^{shift}, $j = 1, 2, \ldots m$. The goal of these random time-shifts is to determine the time-shift vector $\boldsymbol{\tau}$ which would minimize the following quantity, commonly referred to as the cross entropy:

$$L(\boldsymbol{\tau}^{shift}) = \int_0^4 \rho_{ner}(t) \log\left(\rho_{ner}(t)\right)\, dt \tag{20}$$

Minimizing the cross entropy is mathematically equivalent to maximizing the log-likelihood-function with the vector of shift times $\boldsymbol{\tau}^{shift}$ serving as the parameters. This procedure is commonly referred to as Maximum-Likelihood Estimation (MLE) of parameters. We remark that alternative methods can also be used to align the spike times [113].

To minimize the cross entropy, we employed an iterative stochastic gradient descent based algorithm. At every time point, the best shift vector, $\boldsymbol{\tau}_{opt}^{shift}$ so far is perturbed:

$$\boldsymbol{\tau}_p^{shift} = \boldsymbol{\tau}_{opt}^{shift} + \gamma \cdot \boldsymbol{p} \tag{21}$$

where γ is a variable learning rate, and $\boldsymbol{p}_n$ is an $m \times 1$ normally distributed random vector, from a standard normal distribution. The optimal time shift is then updated as:

$$\left[\boldsymbol{\tau}_{opt}^{shift}\right]_{n+1} = \left[\boldsymbol{\tau}_{opt}^{shift}\right]_n - \gamma \left(L(\boldsymbol{\tau}_{opt}^{shift}) - L(\boldsymbol{\tau}_p^{shift})\right) \boldsymbol{p} \tag{22}$$

which serves to estimate the gradient of L as a function of $\boldsymbol{\tau}^{shift}$ and descend the gradient in one-step. This stochastic gradient algorithm is run for $n = 10^4$ iterations for all animals, with an initial $\gamma = 10^{-1}$. Every 10^3 time steps, γ is halved. This slows down the learning rate of this stochastic algorithm for longer times, and yields more precise solutions to $\boldsymbol{\tau}_{opt}^{shift}$.

With the MLE alignment parameters $\boldsymbol{\tau}_{opt}^{shift}$ determined for a single neuron on $[t_j, t_j + 4]$, for the $j = 1, 2, \ldots m$ trials, the spike times are then shifted by the j the component τ_{opt}^{shift}. The τ_{opt}^{shift} determined for aligning a single neuron is used for all neurons in the spike-raster plot with spikes son $[t_j, t_j + 4]$.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Specific Methods for Figures/Supplementary Figures

Figure 2

In Figure 1E-F, the rotating disk network is simulated for a total time of 10 seconds with the θ_{RW} oscillator input off. For Figure 1I-K, the θ_{RW} oscillatory input is turned on for 2 seconds in the interval $[3, 5]$s. All neurons in the rotating disk network additionally receive a white noise current with mean 0 and standard deviation 0.5 pA, to mimic the stochastic firing/SWR initiation observed during slow-wave sleep. The background currents to all neurons, except to the initiators were -40 pA, while the initiator neurons were set to -40.1 pA. Larger or smaller currents to the initiators controls the SWR average rate, while larger/smaller currents to the RD_I neurons controls both the SWR average rate, and the shape of the inter-SWR-interval distributions [63]. For all simulations in Figure 1, the RD_E neurons were split into two tracks. The tracks were assigned by first constructing a random permutation of the rows of $\boldsymbol{\omega}^{RD_E,RD_I}$, splitting the permutation into two sets, and then sorting the permutation with respect to the θ_S phase preference. This is mathematically equivalent to splitting the population of RD_E neurons into two sets, and sorting them on the unit circle. Each track had 50 initiator neurons, and 950 neurons representing the sectors (phases) of the disk. The initiators in each track connect to all other intra-track initiators with a strong recurrent weight ($\omega_i^{RD_{initiator},RD_{initiator}} j = \frac{1.1}{N_{int}}$), and connect to all other excitatory track neurons, and rotating disk neurons with the weight $\omega^{RD_E,RD_{initiator}} = \omega^{RD_I,RD_{initiator}} = \frac{0.1}{N_{int}}$. To implement inter-track competition, the excitatory neurons within a track project to rotating disk interneurons with randomly drawn weights from the interval $[0, 0.2\frac{1}{N_{AA_E}-N_{int}}]$. In Figure 1I, we biased one of the tracks towards being preferentially activated by increasing the bias currents to the track neurons, similarly to the impact the actuator arm network would have. For Figure 1I, In the reading mode, all initiators in Track 1 had a bias of -40.3 pA, while those in Track 2 had a bias of -40.5 pA. All remaining track neurons had a bias of $I^{AA_E} = -40.25pA$ and $I^{AA_I} = -40.5pA$. All excitatory neurons in all tracks and initiators also have a spike-frequency adaptation variable, $u_i(t)$, which increases by an amount of $d = 18$ pA for every spike fired by that neuron and decays with a time constant of 50 ms. The adaptation current is negatively weighted, and serves to slow down or even eliminate repetitive spiking. This adaptation variable can stop a disk-rotation/sharp-wave-ripple. In the writing-mode/theta-oscillation mode, the bias currents for all non-initiator excitatory neurons is increased to $-6pA$. For Figure 1K-L, the total simulation time is 25 seconds, with a non-zero, constant velocity ($\frac{d\psi_A}{dt} = 0.4$ given in the interval $[10, 15]$ seconds. The AA_E in Figure 1K neurons were sorted according to their phase preference, as determined in Supplementary Figure 3. For clarity, only a subset of AA_E neurons (10%) are plotted. In Figure 1N, the R/W head network is simulated for 15 seconds, with 5.1 seconds in the write mode. The information written to the network is a synfire-chain of spikes elicited in the read/write head neurons by an external current. The external current is an additional 20 pA applied to each neuron for 40 ms, in sequence. In the read/SWR-mode, the bias currents to the read/write head neurons are $I^{R/W_E} = -41pA$, $I^{R/W_I} = -40pA$. In the write/theta-oscillation mode, the bias currents are unchanged for the R/W neurons, while $I^{RD_E} = -6pA$ and $I^{RD_I} = 15pA$. For both the read and write modes, the bias current for the initiators was -40.3 pA.

Figure 3

The actuator arm, rotating disk, and read/write head networks were combined into a single network with a 10 second long simulation time. The actuator arm network was identical to all previously considered (Figure 2, supplementary figures) implementations. The signal actuator arm position was generated as the integral of a constant signal, $\dot{\psi}_A = \frac{2}{dt(t_2-t_1)}$, over the interval $[t_1, t_2]$ with $\dot{\psi}_A = 0$ outside this interval. The desired actuator arm position is initialized with $\psi_A(t) = -1$ for $t < t_1$. This initialization results in a linearly ramping ψ_A from -1 to 1 on $[t_1, t_2]$, and $\psi_A = 1$ for $t_2 > 1$. The values $t_1 = 1$, and $t_2 = 2.2$ were used. The rotating disk was as in Figure 1, only with a single track and a single population of initiators. The intra-initiator weight was also stronger, $\omega_i^{AA_{initiator},AA_{initiator}} j = \frac{1.4}{N_{int}}$, which was necessary to generate replays in the read/write head. The adaptation step size was also increased to $d = 22pA$ in all rotating disk pyramidal neurons, to terminate sharp-wave bursts. The actuator arm projects random, sparse excitatory connections to all non-initiator neurons in the rotating

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

disk network. The weights are drawn from a uniform distribution with mean $[0, \frac{0.1}{\sqrt{N^{AA_E}}}]$ with probability $p = 0.1$ and 0 otherwise. The actuator arm also projects to a set of 500 leaky-integrate-and-fire neurons which act to generate the synfire chain of spikes, which serves as the bits to be written in this example. This network has identical parameters as all other neurons, only with a chain of connectivity between these neurons to trigger the synfire chain of spiking. Each neuron in the chain is connected to the next with a synaptic weight of 0.2. These neurons receive a strong hyperpolarizing bias current of $-60.09pA$. A 20 pA current pushes these cells in the superthreshold firing regime where the current is slightly over-threshold (-40.09 pA). All other bias currents are identical to prior implementations in Figure 2 for the isolated networks in reading and writing modes. The learning rate, ϵ, was 1.2×10^{-5}.

Figure 4

The spikes in figure 4H-K were mapped onto LFP phases, with the LFP taken as be the channel with the most spikes detected. The raw signal was further band-pass filtered with a butterworth filter with a $[4, 12]$ Hz window. The phase from the LFP was computed first via a Hilbert transform, and then the application of the arctangent function (*atan2* in MATLAB 2020a). Only epochs of theta oscillations, as defined in [38] were used in our final analysis. All spikes were then transformed into phases via a linear interpolant (*interp1* in MATLAB 2020a). The spikes were subsequently duplicated from $[0, 2\pi]$ into the intervals $[2\pi, 4\pi]$ and $[4\pi, 6\pi]$, with a simple histogram with binds of width $6\pi/100$ used to compute the spike-base histograms on $[0, 6\pi]$. The neurons were subsequently ordered in sequence according to increasing mean-phase. An identical protocol was applied to the rat data also.

Supplementary Figure 2

The total simulation time is 400 seconds, with the first second of simulation time used to initialize the chaotic spiking neural network. Recursive Least Squares (RLS) is turned on for the next 350 seconds and subsequently turned off. The last 49 seconds of simulation time are used for testing the network performance. The network was initialized in the balanced inhibitory regime: $\boldsymbol{\omega}^{AA_E,AA_E} = 0, \boldsymbol{\omega}^{AA_I,AA_E} = 0$, that is, all initial excitatory weights were set to 0, and all excitatory weights were learned. This initial state lets us constrain the firing rates of pyramidal cells to arbitrarily low rates. The bias currents were $I^{AA_E} = -22pA$, and $I^{AA_I} = -20pA$. The initial inhibitory weights were randomly generated with exactly 90% sparsity . Each of the inhibitory neurons made 200 connections, with the connection strength set to $\frac{G}{\sqrt{200\times0.1}}$, where $G = -0.1$. We found that this was sufficient to initialize the network into a chaotic spiking (prior to learning), and therefore serve as an adequate reservoir for RLS. RLS was only applied every 20 time steps ($\Delta t = 20$) for efficiency, as the weight updates in equations (13)-(14) are $O(N^2)$ in time complexity.

Supplementary Figure 3

The trained AA network was simulated for 200 seconds, with a randomly generated actuator arm position $\psi_A(t)$ and $\dot{\psi_A}(t)$, generated similarly to the initial training supervisor. The actuator arm signal, ψ_A remained on for all 200 seconds to establish the neural preference of firing to the ψ_A coordinate. The locations corresponding to each spike times were determined with a linear interpolant (*interp1*, MATLAB 2020a). For each neuron, a histogram was constructed with 41 bins of size 0.05, distributed uniformly over the $[-1, 1]$, which is the operating range of $\psi_A(t)$. For each AA_E and AA_I neuron, the maximum of the spike-ψ_A histogram was determined. The maxima were sorted in ascending order (Figure SF 3A), and the AA neurons were re-ordered in an identical fashion to expose the weight structures (Figure SF 3B-C). The bias currents were $I^{AA_E} = -22pA$, and $I^{AA_I} = -20pA$.

Supplementary Figure 4

The trained AA network was simulated for 50 seconds, with a randomly generated actuator arm position $\psi_A(t)$ and $\dot{\psi_A}(t)$, generated similarly to the initial training supervisor. The bias currents were $I^{AA_E} = -22pA$, and $I^{AA_I} = -20pA$. The position input ψ_A was turned off (set to 0) in Figure SF 4D, while

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

the velocity input was turned off in Figure SF4E after 25 seconds. In all three simulations, the same random initial seed is used to generate ψ_A and $\dot{\psi}_A$.

Supplementary Figure 5

For both simulations, the rotating disk network was subdivided into two tracks with 50 initiators each. In the reading/sharp-wave mode, the bias currents were $I^{AA_E} = -22pA$, and $I^{AA_I} = -20pA$, and $I^{RD_E} = -40.5pA$, $I^{RD_I} = -40pA$. The initiators had a bias current of -40.7 pA. In the writing/theta-oscillation mode, the BIAS currents were the same for the actuator arm networks, while $I^{RD_E} = -9.5pA$, $I^{RD_I} = 15pA$. The initiators in the writing mode were kept off with a hyperpolarizing current ($-60pA$). In the reading mode, the initiators received connections from the AA neurons. The weights were given by the scaled, FORCE-trained decoder, $\boldsymbol{\omega}^{RD_{initiator},AA_E} = 0.5\boldsymbol{\phi}^{AA}$ for the actuator arm position variables $\hat{s}_i(t)$. The first tracks 50 initiators received 10 duplicates of the first 5 components of $\hat{s}_i(t)$, while the second tracks initiators received 10 duplicates of the last 5 components of $\hat{s}_i(t)$. In the writing/theta oscillation mode, the track neurons in the rotating disk network received scaled decoders as weights, $\boldsymbol{\omega}^{RD_E,AA_E} = 12\boldsymbol{\phi}^{AA}$. In this case, the 200 components of $\boldsymbol{\phi}$ were duplicated 5 times each, scaled up, and provided to the rotating disk network as inputs from the actuator arm excitatory neurons.

Supplementary Figure 8

The neuron consisted of 20 Poisson neurons which generate spikes stochastically as part of an inhomogeneous Poisson process. First, in each theta cycle, the individual neurons were modeled as having a phase preference in generating bursts with the preference given by:

$$pp_j(t) = \frac{1}{2\pi} + \frac{1}{4\pi} cos(2\pi\theta_{pp}t + \phi_j^{pp}), \quad \phi_j^{pp} = 2\pi j/19, \quad j = 0, 1, 2, \ldots 19, t \in [0, 2\pi] \tag{23}$$

where $\theta_{pp} = 8Hz$. In each cycle, a cell is allowed to fire a single burst with a phase-centre drawn from the probability distribution $pp_j(t)$. The phase centre for the jth neuron on the ith cycle, χ_{ij} is then used to generate a burst of spikes with probability:

$$\lambda_j(t) = G_{pp}\left(\exp\left(-\frac{(t-\chi_{ij})^2}{\sigma_{pp}}\right)\right), \quad t \in [2\pi(i-1), 2\pi i] \tag{24}$$

where $\sigma_{pp} = 0.1$ controls the width of a burst, and $H(x)$ is the Heaviside function, and $G_{pp} = 0.4$ sets the spikes emitted per burst. The value of G_{pp} was selected so that approximately 10 spikes per burst were emitted. As measured empirically, the cells fired on average 13.11 spikes per cycle, which was estimated after 2000 cycles of firing.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Supplementary Materials

Supplementary Section S1: Disk Drive Dynamics

A Hard Disk Drive (HDD) is a computer component dedicated to storing information. The simplest type of HDD consists of two primary sub-components: A rotating disk and an actuator arm (Figure S1A, [47]). The actuator arm moves across the disk to different regions as the disk is spinning at a constant rotational velocity. The disk is spun by a central spindle, which we will refer to as the "rotator" in the main text to avoid confusion with *sleep spindles*. The head of the actuator arm then writes information onto the disk by magnetizing a layer of ferromagnetic material. The magnetic field imprinted on the disk can be oriented in different directions, such as up or down, thereby allowing the encoding of binary information or bits as these magnetic field directions. The head of the actuator arm can then read data from the disk after the writing procedure is complete. For a fixed position of the actuator arm, the region on the disk that the head can access (through disk rotation) is called a track or cylinder (Figure S1A).

The physical components of the HDD give rise to 3 parameters that describe its function: the disk rotation speed θ_S, the head writing speed θ_W, and the angle of the actuator arm ψ_A (Figure S1B). The disk rotation speed is the number of revolutions per second that the disk makes, while the heard writing speed is the number of bits per second that the disk writes at. Finally, the actuator arm angle dictates which track the head writes information to. Thus, disks operations can be summarized by three parameters: two oscillation frequencies and a position variable for the actuator arm. These variables map nicely onto the dominant models of hippocampal function: dual oscillators and attractor networks.

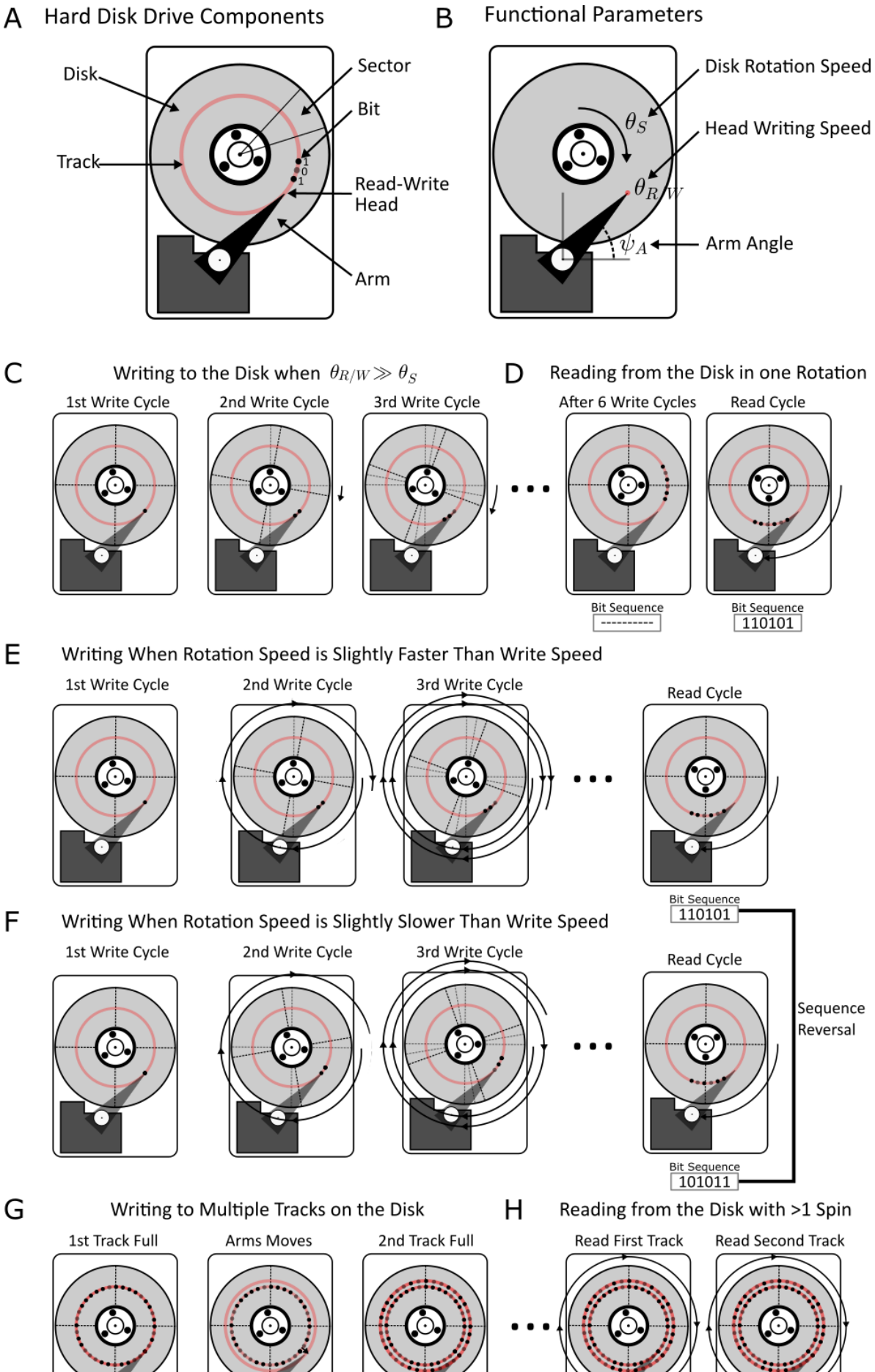
For a standard HDD, we expect that the write speed is significantly faster than the disk rotation speed ($\theta_{R/W} \gg \theta_S$, Figure S1C). In this regime, each write cycle of $\theta_{R/W}$ takes place when the disk has advanced to the next sector on the track. Thus, the exact sequence of bits is maintained and encoded onto the disk on the corresponding sequence of sectors. Then, after writing has concluded, the sequence of bits in a single track can be read in a single revolution of the disk (Figure S1D). Thus, hard drives preserve the sequence of bits, and maximize the number of bits written to a track by ensuring that the write speed is significantly faster than the disk rotation speed. We will refer to this as the Hard Drive "nominal" parameter regime, as this is how a typical HDD stores information.

However, the sequence of bits can still be encoded with slower writing speeds. In particular when the disk writing speed ($\theta_{R/W}$) is slightly slower than the disk spinning speed (θ_S), $\theta_{R/W} = \theta_S - \epsilon$, sequences of bits can still be written to a track (Figure S1E). In this regime, the disk makes slightly more than a full revolution in between write cycles. This ensures that the head writes to the next sector of the track, even under the constraint that the writing speed and disk rotation speeds are similar. Once again, a single revolution of the disk can subsequently read all the information in a track. We refer to this operating range, when $\theta_{R/W} = \theta_S - \epsilon$ for a small frequency difference ϵ as the Hard Drive "precession" parameter regime. Typical HDDs do not operate in this regime.

When the writing speed is slightly faster than the rotation speed ($\theta_{R/W} \approx \theta_{R/W} + \epsilon$), the disk makes slightly less than a full revolution in between write cycles. This implies that rather than writing to the next sector of the track, the head writes the next bit to the previous sector of the track (Figure S1F). The effect of this operating regime is to encode information in the reverse order that it was observed. A single revolution in a read cycle can then read the entire bit sequence in reverse. We refer to this as the Hard Drive "recession" parameter regime. As in the HDD precession regime, typical HDDs do not operate in this regime.

Finally, when the head has saturated a track with a sufficiently long sequence of bits, the only recourse to store more information is to change ψ_A and encode more information onto new tracks (Figure S1G). Thus, multiple read cycles are now necessary to read out sequences that can not fit on a single track (Figure S1H). In particular, the disk must complete multiple full rotations, with the actuator arm switching between tracks in each rotation.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

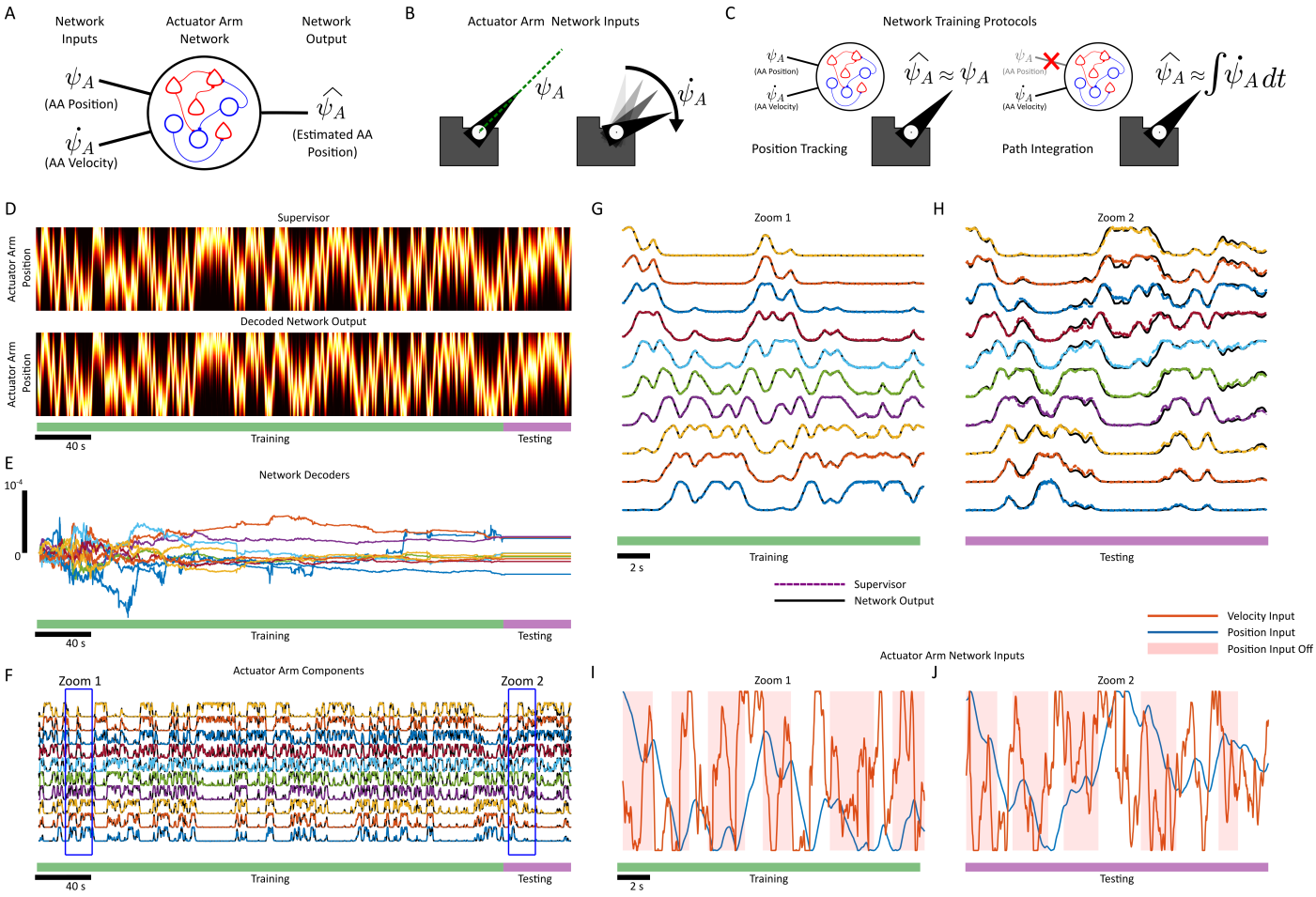


bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 1: The Operations of a Hard Disk Drive

(A) A Hard Disk Drive (HDD) schematic with its basic components. The rotating disk is used to store information onto tracks, which correspond to a circular segment of the disk. A sector on the disk is a wedge shaped region formed by any two radii. The actuator arm points to discrete tracks, which are the regions of the disk a stationary actuator arm has access by virtue of disk rotation alone. Bits are encoded onto the rotating disk by a read/write head on the apex of the actuator arm. The bits are encoded by inducing magnetic fields of different directions onto a ferromagnetic material on the disk surface. **(B)** The functional parameters of an HDD that describe its state: the disk rotation speed (θ_S), the angular position of the actuator arm, ψ_A, and the head writing speed, $\theta_{R/W}$. **(C)** When the write speed $\theta_{R/W}$ is substantially faster than the disk rotation speed, bits are written continuously as the disk spins. **(D)** Written bits on a single track can be read in a single revolution of the disk by the read/write head. **(E)** Bit sequences can also be written when $\theta_{R/W} = \theta_S - \epsilon$, where the writing to a track corresponds to slightly greater than a full revolution of the disk. The written information can be read in a single disk rotation, as in **(D)**. **(F)** When the disk rotation speed is slightly slower than the writing speed, the disk has advanced slightly less than a full revolution when a bit is written. This results in the bits being encoded and readout in the reverse order with which they were encoded. **(G)** When a track is saturated with bits, the actuator arm must move to the next track on the disk to write additional information. **(H)** Long sequences of bits must be written to multiple tracks. In order to access this information, multiple disk rotations are required with the actuator arm adjusting between tracks. This results in multiple read cycles/disk rotations required to readout information.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

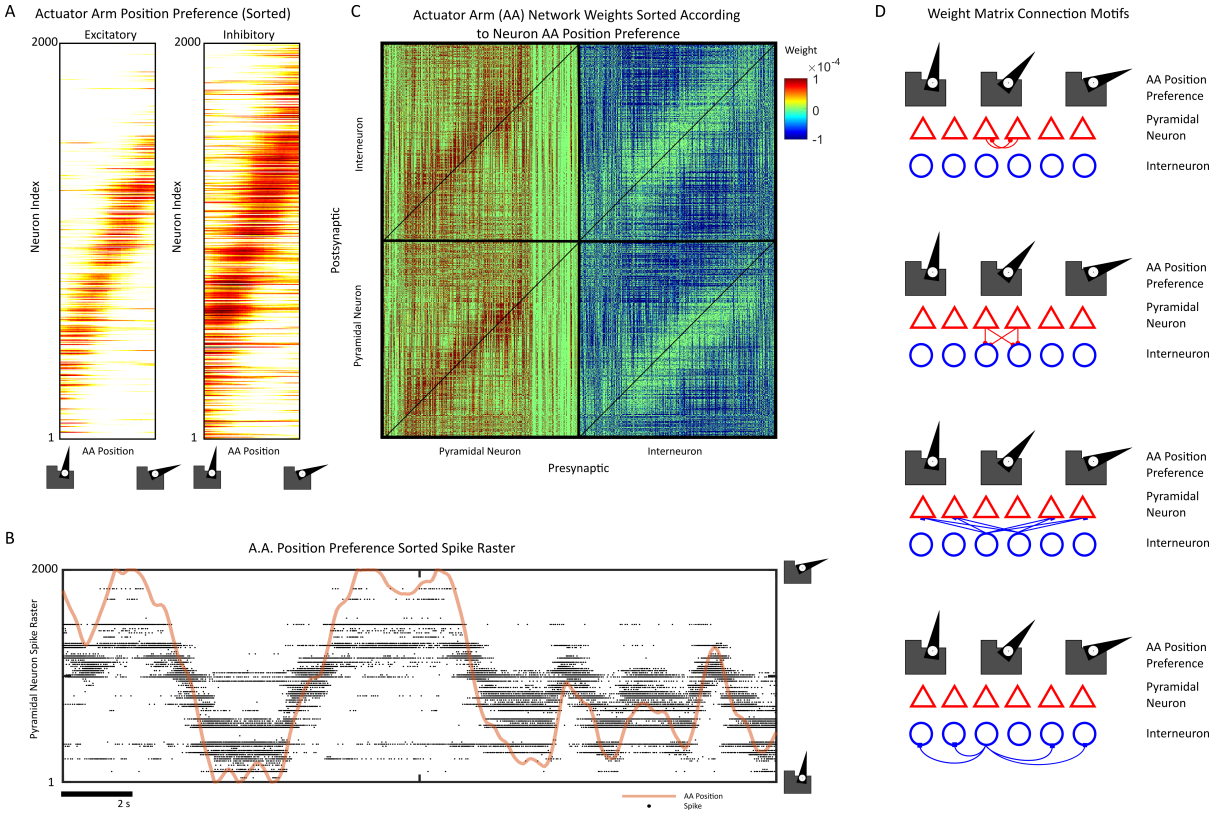


bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 2: FORCE Training the actuator armNetwork

(A) A network of 2000 leaky integrate-and-fire pyramidal neurons and 2000 leaky-integrate-and-fire interneurons is collectively trained to estimate the position of the actuator arm, $\hat{\psi_A}$, on a disk drive. **(B)** The network receives two inputs, a position variable ψ_A, and a velocity variable $\dot{\psi_A}$. **(C)** The position variable and velocity variables can set the position of the actuator arm in the actuator armnetwork. When both inputs are on, the network outputs an actuator arm position ($\hat{\psi_A}$) as determined by its position input ψ_A. When the position input is off, the network is trained to integrate the velocity input $\dot{\psi_A}$. **(D)** The supervisor used to train the actuator arm(AA) network consists of 100 components that encode actuator arm position, forming a high-dimensional supervisor [80]. The components are sorted according to the supervisor-encoding preferences with respect to the actuator arm position ψ_A. As a result, the supervisor forms a bump of activity, indicating the actuator arm position $\hat{\psi_A}$. The training signal is randomly generated, with the actuator arm position being randomly dropped during training. FORCE training was as applied for 230 seconds (green), after the first second of simulation time to remove any transients. FORCE training was turned off in the last 19 seconds (purple) of the simulation. The supervisor (top) and decoded network output (bottom) are shown as heat maps. **(E)** A subset of the decoders, $\boldsymbol{\phi}_i$, $i = 1, 2, \ldots N$ during (green) and after (purple) FORCE training. (**F**) A subset of the 100 supervisor components from (A)-(B). The network output is plotted as coloured dashed lines, while the supervisor components are in solid black. **(G)** A 10 second zoom of subset of the supervisor and network components while the FORCE training is on. **(H)** A 10 second zoom of the subset of the supervisor and network components while FORCE training is off. **(G)** The inputs (AA velocity in blue, AA position in orange) to the network during training. The red shading denotes the position input was turned off during this time period. **(H)** The inputs (AA velocity in blue, AA position in orange) to the network after training. The red shading denotes the position input was turned off during this time period.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

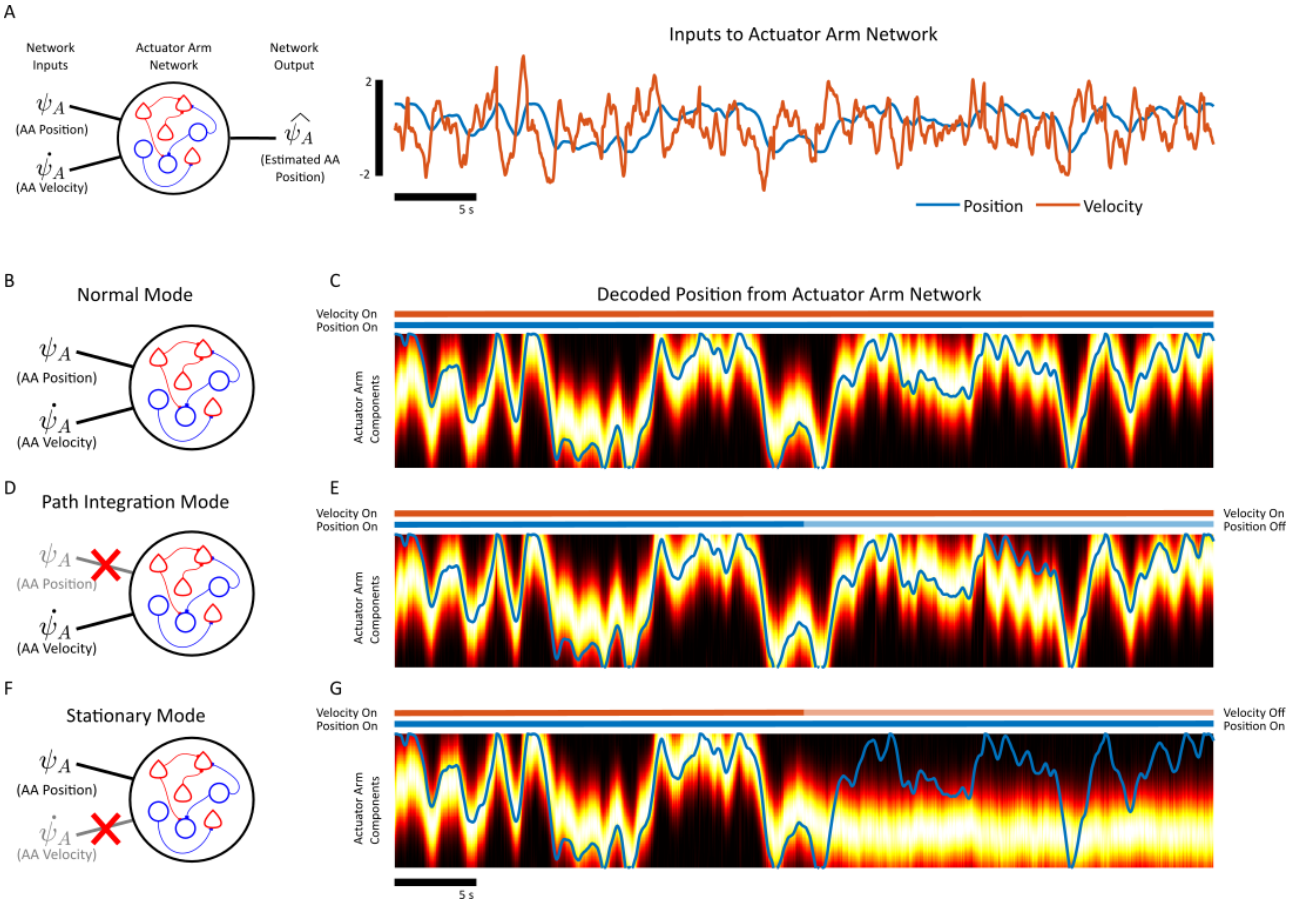


bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 3: Weight Structure of the Force Trained actuator armNetwork

(A) The 2000 pyramidal neurons (left) and 2000 interneurons (right) are plotted in order of their firing preference with respect to the actuator arm position ψ_A. The preference is estimated by binning the spikes into a histogram on the actuator arm range $\psi_A \in [-1, 1]$. Both interneurons and pyramidal neurons display firing preferences with respect to ψ_A, although the pyramidal neuron preferences are narrower. **(B)** A spike raster plot of the pyramidal neurons sorted according to the firing preference in ψ_A. The position is scaled and overlaid for comparison. **(C)** The ψ_A-preference sorted weight matrices for the pyramidal and inhibitory neurons. Excitatory connections, which are exclusively made by pyramidal neurons are plotted by red, while inhibitory connections, are plotted in blue. All EE, EI, II, and IE weight matrices show clear banding along the main diagonal. Note that the green vertical band is caused by a sub-population of pyramidal neurons that did not fire strongly during navigation, and as a result, do not display strong place-preferences and thus continue the banding structure. These neurons were also sorted to the top 400 pyramidal neurons due to the MATLAB *sort* function. **(D)** The four connection motifs created by FORCE training and shown in the banding structure. Pyramidal neurons tend to excite pyramidal neurons with similar ψ_A preference. Pyramidal neurons excite interneurons with similar ψ_A preference. Interneurons inhibit pyramidal neurons with different ψ_A preference. Interneurons inhibit interneurons with different ψ_A preference.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

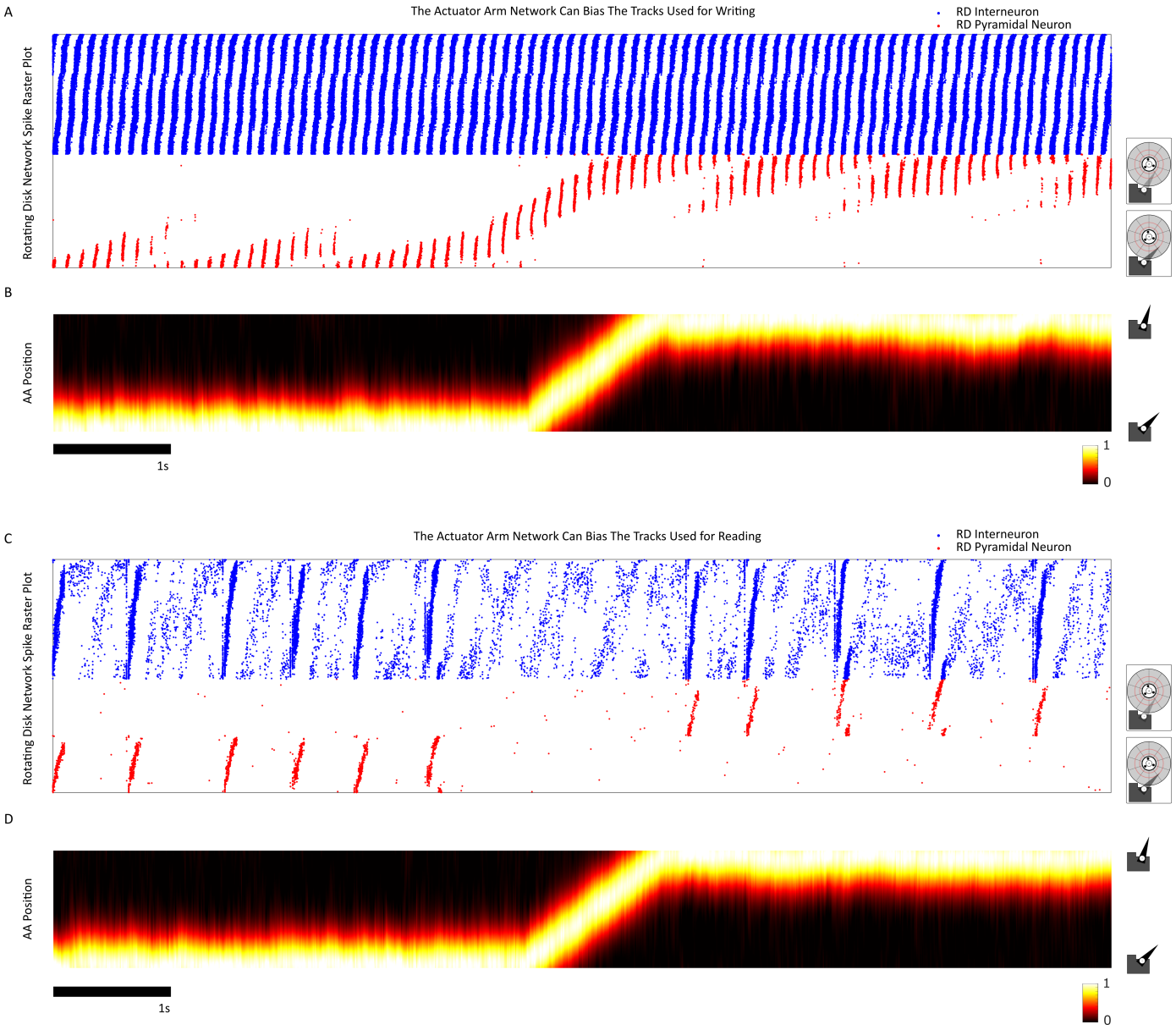


bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 4: Operating Modes of the actuator armNetwork

(A) The actuator arm network receives two signals, a velocity signal $\dot{\psi_A}$ (orange) and a position signal ψ_A (blue). The position signal denotes the desired position of the actuator arm. The velocity signal is randomly generated, while the position signal is the integral of the velocity signal. **(B)** Under "normal operating mode", the position signal and velocity signal are both applied. **(C)** The position of the actuator arm $\hat{\psi_A}$ closely tracks the desired position ψ_A for a 50 second simulation of the AA network **(D)** In the path integration mode, the position signal ψ_A is not present. **(E)** The position signal is dropped in the last 25 seconds of simulation. The actuator arm network still tracks the desired position of the actuator arm by integrating the velocity signal $\dot{\psi_A}$. **(F)** In the stationary mode, the velocity signal $\dot{\psi_A}$ is dropped while the position signal is still applied. **(G)** The velocity signal is dropped in the last 25 seconds of the simulation. The network interprets this drop as a velocity of 0, and as such, retains an approximation of the last known position of the actuator arm.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.



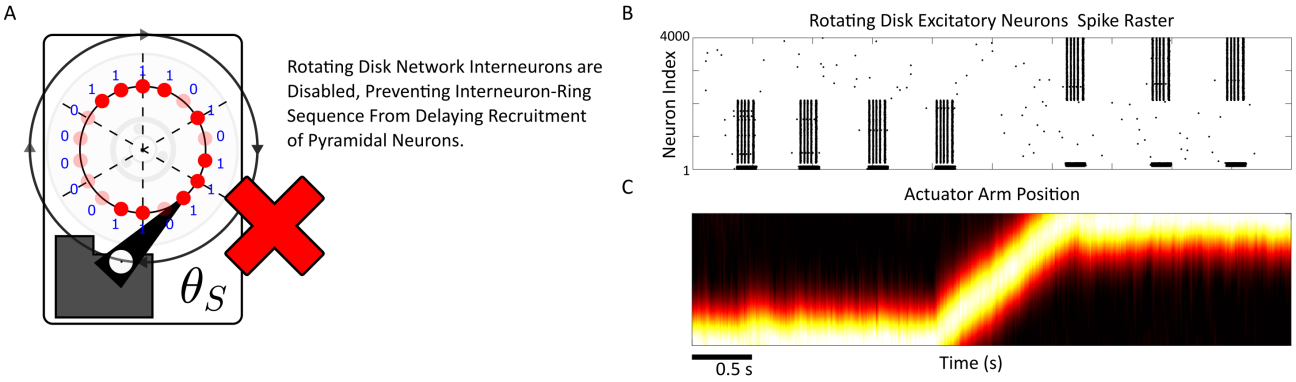
bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 5: The actuator arm network can select tracks on the rotating disk Network

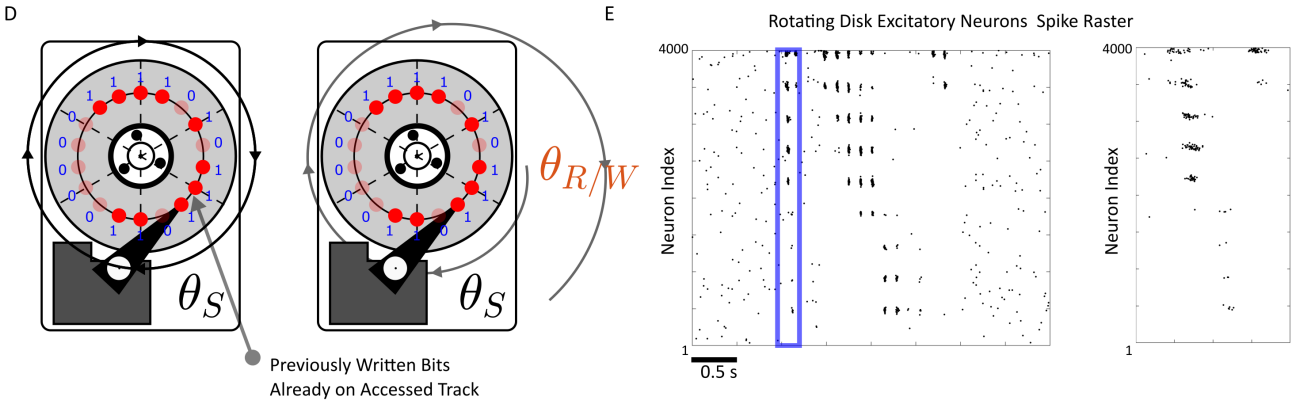
(A) The spike raster plot for the excitatory (red) and inhibitory (blue) spikes for the rotating disk network while both oscillations θ_S and $\theta_{R/W}$ are operating. The rotating disk network is separated into two tracks with 950 neurons each. **(B)** The decoded position of the actuator arm. **(C)** The spike raster plot for the rotating disk network when only the θ_S oscillation is transiently induced by noise. **(D)** The position of the actuator arm.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

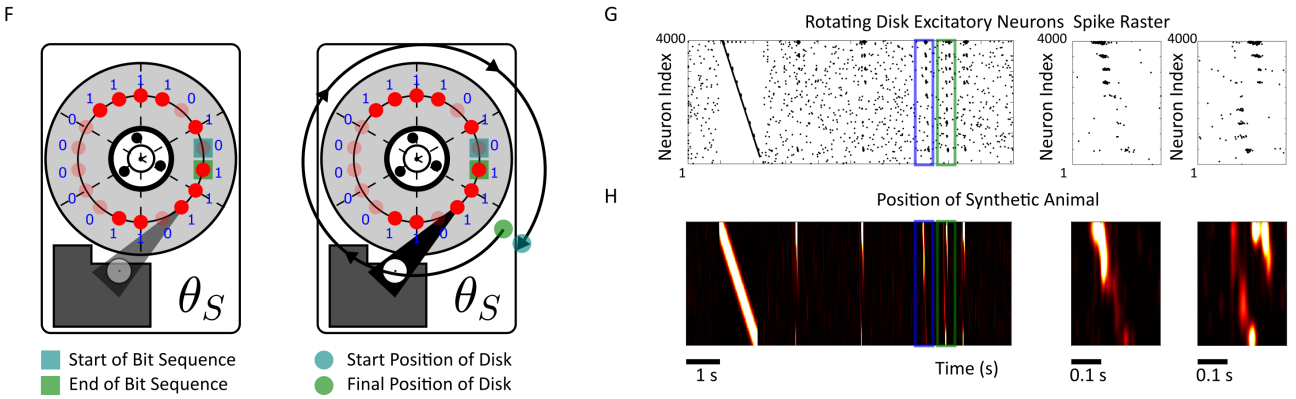
Reactivation: Disabling the Rotating Disk Network



Preplays: Selecting Tracks with Pre-Existing Information



Fragmented Replays: Start of Disk Rotation does not Coincide with Start of Sequence



bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

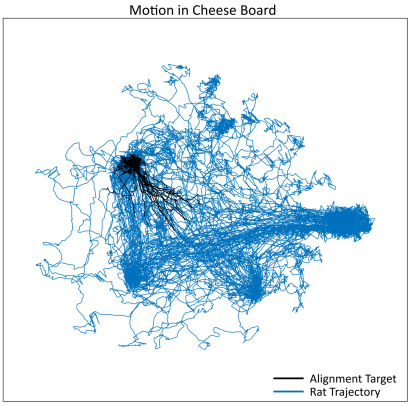
Supplementary Figure 6: The Zoo of Hippocampal Data Access Methods

(A) By disabling the rotating disk network interneurons, the spikes in the pyramidal neurons in the rotating disk network lose sequential content. **(B)** The simulated RD/AA network from Supplementary Figure 5A with the RD interneurons receiving a strong hyperpolarizing current to stop their recruitment. The pyramidal neurons now fire synchronized bursts in an event closer to reactivation rather than replay. **(C)** The position of the actuator arm, which can serve to bias which assembly becomes reactivated. **(D)** If a track already contains pre-written information, then new information can no longer be written to this track without the pre-existing information being deleted. Pre-plays may correspond to this scenario where tracks (SWRs) contain pre-existing information and are accessed during animal navigation. **(E)** In the NDD model, when information is already stored on a track/SWR, a compressed sequence (blue) occurs during a sharp-wave prior to sequential theta sequences during navigation. **(F)** For every sequence written to a track on a disk drive, there is a sequence start and end bit. If the disk rotation does not start with the sequence start bit, the sequence is accessed in a fragmented order. **(G)** Simulation of the NDD model with two separate replay events highlighted in blue and green. The first sequence, blue, corresponds to activating a track when the sequence starts. The second sequence, in green, corresponds to a mismatch between when the learned sequence starts and with which initial phase the rotating disk network is activated. **(H)** The decoded position of a synthetically generated animal. Replays are also decoded by a simple linear decoder, with a normal replay on the left (blue), and a fragmented replay on the right (green).

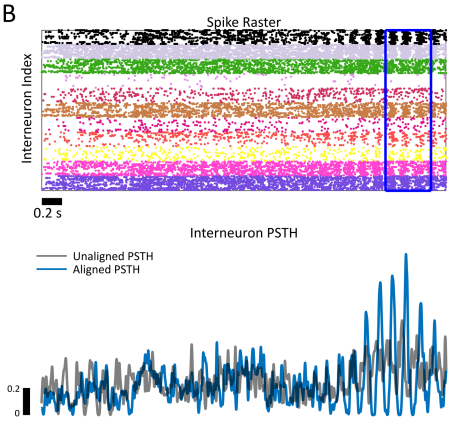
bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

A

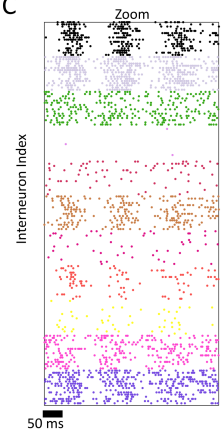
Animal 1



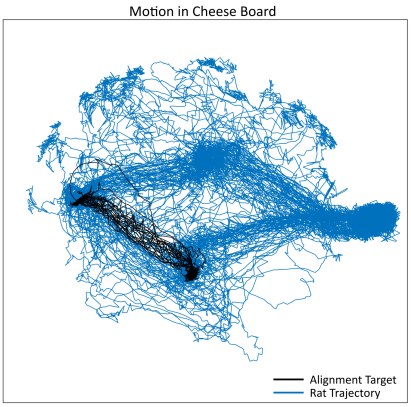
B

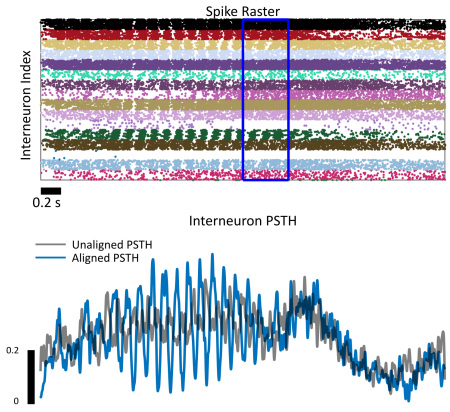


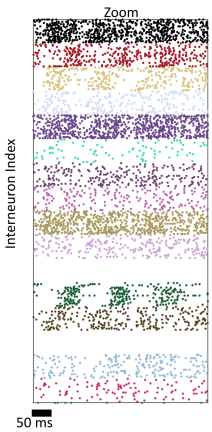
C



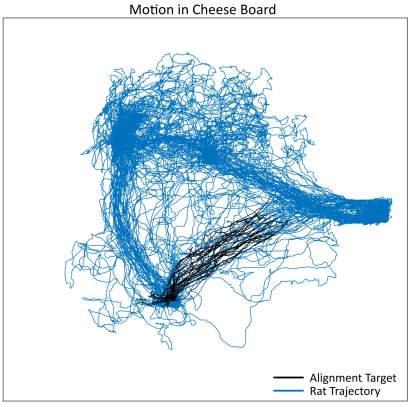
Animal 2

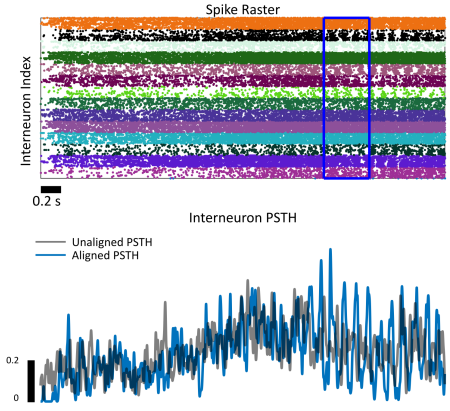


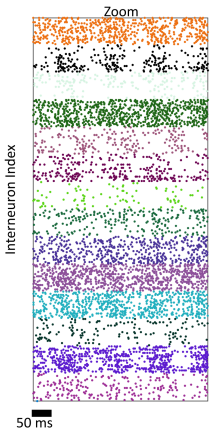




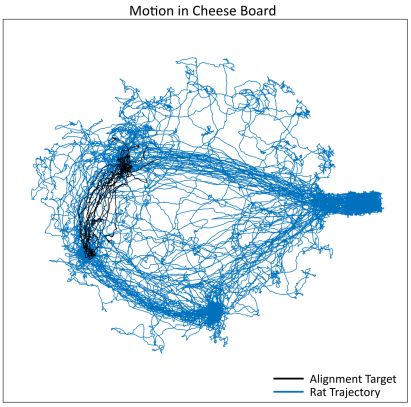
Animal 3

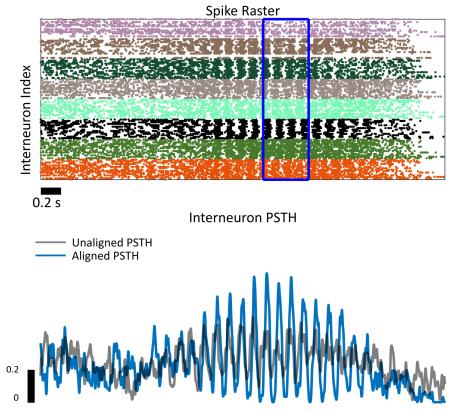


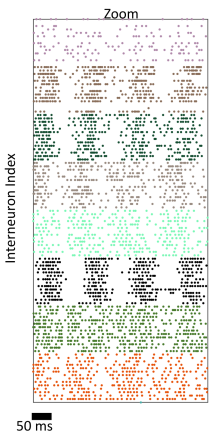




Animal 4





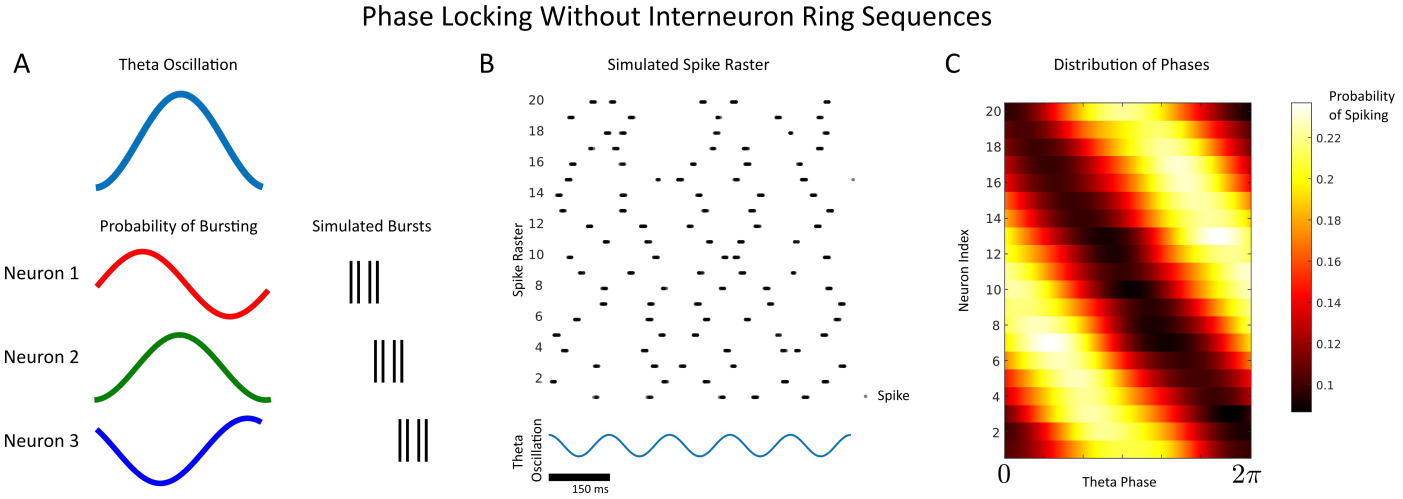


bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Figure 7: Time-Shift Alignment of Interneurons for 4 Different Animals

(A) Motion trajectories used for each of the four animals as targets for maximum likelihood estimation based alignment of the spikes. **(B)** The interneurons in the recorded animal. The black interneuron was used as the target for alignment, with its determined time-shifts applied to all other interneurons. The kernel density estimate of the aligned (blue) and un-aligned (black) spike density for all interneurons is show. **(C)** A zoom of the interneurons during periods of high theta-locking in the spike-density. The zoomed segment corresponds to the blue box in (B).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.



bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

Supplementary Figure 8: Preferred-Phase Firing without Interneuron Ring Sequences

(A) In a theta-modulated poisson-spiking model, each neuron fires a burst of spikes with a theta modulated probability. The theta oscillation is used as a clock to force every cell to fire a burst, with the bursts elicited with a specific phase preference. **(B)** Despite phase-preferential firing of the neurons in this simulated system, a ring of bursts is not elicited. **(C)** Despite the lack of a ring of bursting, each neuron in the network exhibits phase locking of their spikes onto the theta oscillation. The phase locking forms a continuous ring, without interneuron-ring-sequences being elicited.

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

References

[1] Singer, A. C., Carr, M. F., Karlsson, M. P. & Frank, L. M. Hippocampal swr activity predicts correct decisions during the initial learning of an alternation task. *Neuron* **77**, 1163–1173 (2013).

[2] Singer, A. C. & Frank, L. M. Rewarded outcomes enhance reactivation of experience in the hippocampus. *Neuron* **64**, 910–921 (2009).

[3] Farooq, U., Sibille, J., Liu, K. & Dragoi, G. Strengthened temporal coordination within pre-existing sequential cell assemblies supports trajectory replay. *Neuron* **103**, 719–733 (2019).

[4] Gillespie, A. K. *et al.* Hippocampal replay reflects specific past experiences rather than a plan for subsequent choice. *Neuron* **109**, 3149–3163 (2021).

[5] Denovellis, E. L. *et al.* Hippocampal replay of experience at real-world speeds. *Elife* **10**, e64505 (2021).

[6] Liu, S., Grosmark, A. D. & Chen, Z. Methods for assessment of memory reactivation. *Neural computation* **30**, 2175–2209 (2018).

[7] Jai, Y. Y. *et al.* Distinct hippocampal-cortical memory representations for experiences associated with movement versus immobility. *Elife* **6**, e27621 (2017).

[8] Rothschild, G., Eban, E. & Frank, L. M. A cortical–hippocampal–cortical loop of information processing during memory consolidation. *Nature neuroscience* **20**, 251–259 (2017).

[9] Von Neumann, J. *The computer and the brain* (Yale University Press, 2012).

[10] Searle, J. R. Is the brain a digital computer? In *Proceedings and addresses of the american philosophical association*, vol. 64, 21–37 (JSTOR, 1990).

[11] Shagrir, O. Brains as analog-model computers. *Studies In History and Philosophy of Science Part A* **41**, 271–279 (2010).

[12] Eliasmith, C. Is the brain analog or digital? (2000).

[13] Kempermann, G. Why new neurons? possible functions for adult hippocampal neurogenesis. *Journal of neuroscience* **22**, 635–638 (2002).

[14] Kirkland, K. L. High-tech brains: a history of technology-based analogies and models of nerve and brain function. *Perspectives in Biology and Medicine* **45**, 212–223 (2002).

[15] Brette, R. Brains as computers: metaphor, analogy, theory or fact? *Frontiers in Ecology and Evolution* **10**, 878729 (2022).

[16] Nelson, M. E. & Bower, J. M. Brain maps and parallel computers. *Trends in neurosciences* **13**, 403–408 (1990).

[17] Conrad, M. The brain-machine disanalogy. *BioSystems* **22**, 197–213 (1989).

[18] Searle, J. R. Is the brain a digital computer? In *Proceedings and addresses of the American Philosophical Association*, vol. 64, 21–37 (JSTOR, 1990).

[19] Shagrir, O. Why we view the brain as a computer. *Synthese* **153**, 393–416 (2006).

[20] O'Keefe, J. & Recce, M. L. Phase relationship between hippocampal place units and the eeg theta rhythm. *Hippocampus* **3**, 317–330 (1993).

[21] Hubel, D. H. & Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology* **160**, 106 (1962).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

[22] Moser, E. I., Kropff, E. & Moser, M.-B. Place cells, grid cells, and the brain's spatial representation system. *Annu. Rev. Neurosci.* **31**, 69–89 (2008).

[23] Bliss, T. V. & Lømo, T. Long-lasting potentiation of synaptic transmission in the dentate area of the anaesthetized rabbit following stimulation of the perforant path. *The Journal of physiology* **232**, 331–356 (1973).

[24] Bi, G.-q. & Poo, M.-m. Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of neuroscience* **18**, 10464–10472 (1998).

[25] Wang, Y., Romani, S., Lustig, B., Leonardo, A. & Pastalkova, E. Theta sequences are essential for internally generated hippocampal firing fields. *Nature neuroscience* **18**, 282–288 (2015).

[26] Buzsáki, G. Two-stage model of memory trace formation: a role for "noisy" brain states. *Neuroscience* **31**, 551–570 (1989).

[27] Buzsáki, G. Theta oscillations in the hippocampus. *Neuron* **33**, 325–340 (2002).

[28] Yamamoto, J. & Tonegawa, S. Direct medial entorhinal cortex input to hippocampal ca1 is crucial for extended quiet awake replay. *Neuron* **96**, 217–227 (2017).

[29] Davidson, T. J., Kloosterman, F. & Wilson, M. A. Hippocampal replay of extended experience. *Neuron* **63**, 497–507 (2009).

[30] Boyce, R., Glasgow, S. D., Williams, S. & Adamantidis, A. Causal evidence for the role of rem sleep theta rhythm in contextual memory consolidation. *Science* **352**, 812–816 (2016).

[31] Heusser, A. C., Poeppel, D., Ezzyat, Y. & Davachi, L. Episodic sequence memory is supported by a theta–gamma phase code. *Nature neuroscience* **19**, 1374 (2016).

[32] Lisman, J. & Buzsáki, G. A neural coding scheme formed by the combined function of gamma and theta oscillations. *Schizophrenia bulletin* **34**, 974–980 (2008).

[33] Lisman, J. E. & Idiart, M. A. Storage of 7+/-2 short-term memories in oscillatory subcycles. *Science* **267**, 1512–1515 (1995).

[34] Jensen, O. & Lisman, J. E. Novel lists of 7+/-2 known items can be reliably stored in an oscillatory short-term memory network: interaction with long-term memory. *Learning & Memory* **3**, 257–263 (1996).

[35] Nádasdy, Z., Hirase, H., Czurkó, A., Csicsvari, J. & Buzsáki, G. Replay and time compression of recurring spike sequences in the hippocampus. *Journal of Neuroscience* **19**, 9497–9507 (1999).

[36] Lee, A. K. & Wilson, M. A. Memory of sequential experience in the hippocampus during slow wave sleep. *Neuron* **36**, 1183–1194 (2002).

[37] Montgomery, S. M. & Buzsáki, G. Gamma oscillations dynamically couple hippocampal ca3 and ca1 regions during memory task performance. *Proceedings of the National Academy of Sciences* **104**, 14495–14500 (2007).

[38] Dupret, D., O'neill, J., Pleydell-Bouverie, B. & Csicsvari, J. The reorganization and reactivation of hippocampal maps predict spatial memory performance. *Nature neuroscience* **13**, 995–1002 (2010).

[39] McNamara, C. G., Tejero-Cantero, Á., Trouche, S., Campo-Urriza, N. & Dupret, D. Dopaminergic neurons promote hippocampal reactivation and spatial memory persistence. *Nature neuroscience* **17**, 1658–1660 (2014).

[40] Hasselmo, M. E., Wyble, B. P. & Wallenstein, G. V. Encoding and retrieval of episodic memories: role of cholinergic and gabaergic modulation in the hippocampus. *Hippocampus* **6**, 693–708 (1996).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

[41] Kunec, S., Hasselmo, M. E. & Kopell, N. Encoding and retrieval in the ca3 region of the hippocampus: a model of theta-phase separation. *Journal of neurophysiology* **94**, 70–82 (2005).

[42] Manns, J. R., Zilli, E. A., Ong, K. C., Hasselmo, M. E. & Eichenbaum, H. Hippocampal ca1 spiking during encoding and retrieval: relation to theta phase. *Neurobiology of learning and memory* **87**, 9–20 (2007).

[43] Hasselmo, M. E., Bodelón, C. & Wyble, B. P. A proposed function for hippocampal theta rhythm: separate phases of encoding and retrieval enhance reversal of prior learning. *Neural computation* **14**, 793–817 (2002).

[44] Skaggs, W. E., McNaughton, B. L., Wilson, M. A. & Barnes, C. A. Theta phase precession in hippocampal neuronal populations and the compression of temporal sequences. *Hippocampus* **6**, 149–172 (1996).

[45] McClelland, J. L., McNaughton, B. L. & O'reilly, R. C. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological review* **102**, 419 (1995).

[46] Diba, K. & Buzsáki, G. Forward and reverse hippocampal place-cell sequences during ripples. *Nature neuroscience* **10**, 1241 (2007).

[47] Goddard, W. A. & Lynott, J. J. Direct access magnetic disc storage device (1970). US Patent 3,503,060.

[48] Buzsáki, G. Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. *Hippocampus* **25**, 1073–1188 (2015).

[49] Dragoi, G. & Tonegawa, S. Preplay of future place cell sequences by hippocampal cellular assemblies. *Nature* **469**, 397–401 (2011).

[50] Dragoi, G. & Tonegawa, S. Distinct preplay of multiple novel spatial experiences in the rat. *Proceedings of the National Academy of Sciences* **110**, 9100–9105 (2013).

[51] Grosmark, A. D. & Buzsáki, G. Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences. *Science* **351**, 1440–1443 (2016).

[52] Ólafsdóttir, H. F., Barry, C., Saleem, A. B., Hassabis, D. & Spiers, H. J. Hippocampal place cells construct reward related sequences through unexplored space. *Elife* **4**, e06063 (2015).

[53] Ólafsdóttir, H. F., Barry, C., Saleem, A. B., Hassabis, D. & Spiers, H. J. Hippocampal place cells construct reward related sequences through unexplored space. *Elife* **4**, e06063 (2015).

[54] Lisman, J. & Buzsáki, G. A neural coding scheme formed by the combined function of gamma and theta oscillations. *Schizophrenia bulletin* **34**, 974–980 (2008).

[55] Hasselmo, M. E. What is the function of hippocampal theta rhythm?—linking behavioral data to phasic properties of field potential and unit recording data. *Hippocampus* **15**, 936–949 (2005).

[56] Wilson, M. A. & McNaughton, B. L. Reactivation of hippocampal ensemble memories during sleep. *Science* **265**, 676–679 (1994).

[57] Berners-Lee, A. *et al.* Hippocampal replays appear after a single experience and incorporate greater detail with more experience. *Neuron* **110**, 1829–1842 (2022).

[58] Johnson, A. & Redish, A. D. Hippocampal replay contributes to within session learning in a temporal difference reinforcement learning model. *Neural Networks* **18**, 1163–1171 (2005).

[59] Ecker, A. *et al.* Hippocampal sharp wave-ripples and the associated sequence replay emerge from structured synaptic interactions in a network model of area ca3. *Elife* **11**, e71850 (2022).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

[60] Maboudi, K. *et al.* Uncovering temporal structure in hippocampal output patterns. *Elife* **7**, e34467 (2018).

[61] Cutsuridis, V. & Hasselmo, M. Spatial memory sequence encoding and replay during modeled theta and ripple oscillations. *Cognitive Computation* **3**, 554–574 (2011).

[62] Chenkov, N., Sprekeler, H. & Kempter, R. Memory replay in balanced recurrent networks. *PLoS computational biology* **13**, e1005359 (2017).

[63] Nicola, W. & Clopath, C. A diversity of interneurons and hebbian plasticity facilitate rapid compressible learning in the hippocampus. *Nature Neuroscience* **22**, 1168 (2019).

[64] Harris, K. D. *et al.* Spike train dynamics predicts theta-related phase precession in hippocampal pyramidal cells. *Nature* **417**, 738–741 (2002).

[65] Tsodyks, M. V., Skaggs, W. E., Sejnowski, T. J. & McNaughton, B. L. Population dynamics and theta rhythm phase precession of hippocampal place cell firing: a spiking neuron model. *Hippocampus* **6**, 271–280 (1996).

[66] Jaramillo, J., Schmidt, R. & Kempter, R. Modeling inheritance of phase precession in the hippocampal formation. *Journal of Neuroscience* **34**, 7715–7731 (2014).

[67] Maurer, A. P. & McNaughton, B. L. Network and intrinsic cellular mechanisms underlying theta phase precession of hippocampal neurons. *Trends in neurosciences* **30**, 325–333 (2007).

[68] Thurley, K., Leibold, C., Gundlfinger, A., Schmitz, D. & Kempter, R. Phase precession through synaptic facilitation. *Neural Computation* **20**, 1285–1324 (2008).

[69] Ego-Stengel, V. & Wilson, M. A. Spatial selectivity and theta phase precession in ca1 interneurons. *Hippocampus* **17**, 161–174 (2007).

[70] Chadwick, A., van Rossum, M. C. & Nolan, M. F. Flexible theta sequence compression mediated via phase precessing interneurons. *Elife* **5** (2016).

[71] Chadwick, A., van Rossum, M. & Nolan, M. Modelling phase precession in the hippocampus. *BMC Neuroscience* **16**, 1–2 (2015).

[72] Hasselmo, M. E., Alexander, A. S., Dannenberg, H. & Newman, E. L. Overview of computational models of hippocampus and related structures: Introduction to the special issue (2020).

[73] Bendor, D. & Wilson, M. A. Biasing the content of hippocampal replay during sleep. *Nature neuroscience* **15**, 1439–1444 (2012).

[74] Wills, T. J., Lever, C., Cacucci, F., Burgess, N. & O'Keefe, J. Attractor dynamics in the hippocampal representation of the local environment. *Science* **308**, 873–876 (2005).

[75] Knierim, J. J. & Neunuebel, J. P. Tracking the flow of hippocampal computation: Pattern separation, pattern completion, and attractor dynamics. *Neurobiology of learning and memory* **129**, 38–49 (2016).

[76] Agmon, H. & Burak, Y. A theory of joint attractor dynamics in the hippocampus and the entorhinal cortex accounts for artificial remapping and grid cell field-to-field variability. *Elife* **9**, e56894 (2020).

[77] Steemers, B. *et al.* Hippocampal attractor dynamics predict memory-based decision making. *Current Biology* **26**, 1750–1757 (2016).

[78] Samsonovich, A. & McNaughton, B. L. Path integration and cognitive mapping in a continuous attractor neural network model. *Journal of Neuroscience* **17**, 5900–5920 (1997).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

[79] Bush, D. & Burgess, N. A hybrid oscillatory interference/continuous attractor network model of grid cell firing. *Journal of Neuroscience* **34**, 5065–5079 (2014).

[80] Nicola, W. & Clopath, C. Supervised learning in spiking neural networks with force trainnig. *NatureCommunications* **8** (2017).

[81] Sussillo, D. & Abbott, L. F. Generating coherent patterns of activity from chaotic neural networks. *Neuron* **63**, 544–557 (2009).

[82] Schlingloff, D., Káli, S., Freund, T. F., Hájos, N. & Gulyás, A. I. Mechanisms of sharp wave initiation and ripple generation. *Journal of Neuroscience* **34**, 11385–11398 (2014).

[83] Grosmark, A. D. & Buzsáki, G. Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences. *Science* **351**, 1440–1443 (2016).

[84] Gava, G. P. *et al.* Integrating new memories into the hippocampal network activity space. *Nature neuroscience* **24**, 326–330 (2021).

[85] Harvey, C. D., Collman, F., Dombeck, D. A. & Tank, D. W. Intracellular dynamics of hippocampal place cells during virtual navigation. *Nature* **461**, 941 (2009).

[86] Pastalkova, E., Itskov, V., Amarasingham, A. & Buzsáki, G. Internally generated cell assembly sequences in the rat hippocampus. *Science* **321**, 1322–1327 (2008).

[87] Wang, Y., Roth, Z. & Pastalkova, E. Synchronized excitability in a network enables generation of internal neuronal sequences. *Elife* **5** (2016).

[88] Eliasmith, C. A unified approach to building and controlling spiking attractor networks. *Neural computation* **17**, 1276–1314 (2005).

[89] Pfeiffer, B. E. & Foster, D. J. Autoassociative dynamics in the generation of sequences of hippocampal place cells. *Science* **349**, 180–183 (2015).

[90] Hasselmo, M. E., Bodelón, C. & Wyble, B. P. A proposed function for hippocampal theta rhythm: separate phases of encoding and retrieval enhance reversal of prior learning. *Neural computation* **14**, 793–817 (2002).

[91] Manns, J. R., Zilli, E. A., Ong, K. C., Hasselmo, M. E. & Eichenbaum, H. Hippocampal ca1 spiking during encoding and retrieval: relation to theta phase. *Neurobiology of learning and memory* **87**, 9–20 (2007).

[92] Siegle, J. H. & Wilson, M. A. Enhancement of encoding and retrieval functions through theta phase-specific manipulation of hippocampus. *elife* **3** (2014).

[93] Hasselmo, M. E. What is the function of hippocampal theta rhythm?—linking behavioral data to phasic properties of field potential and unit recording data. *Hippocampus* **15**, 936–949 (2005).

[94] Swanson, R. A., Levenstein, D., McClain, K., Tingley, D. & Buzsáki, G. Variable specificity of memory trace reactivation during hippocampal sharp wave ripples. *Current Opinion in Behavioral Sciences* **32**, 126–135 (2020).

[95] Azizi, A. H., Wiskott, L. & Cheng, S. A computational model for preplay in the hippocampus. *Frontiers in computational neuroscience* **7**, 161 (2013).

[96] Gupta, A. S., Van Der Meer, M. A., Touretzky, D. S. & Redish, A. D. Hippocampal replay is not a simple function of experience. *Neuron* **65**, 695–705 (2010).

[97] Wu, X. & Foster, D. J. Hippocampal replay captures the unique topological structure of a novel environment. *Journal of Neuroscience* **34**, 6459–6469 (2014).

bioRxiv preprint doi: https://doi.org/10.1101/2022.10.05.511000; this version posted October 7, 2022. 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-ND 4.0 International license.

[98] Klausberger, T. *et al.* Brain-state-and cell-type-specific firing of hippocampal interneurons in vivo. *Nature* **421**, 844–848 (2003).

[99] Lapray, D. *et al.* Behavior-dependent specialization of identified hippocampal interneurons. *Nature neuroscience* **15**, 1265–1271 (2012).

[100] Somogyi, P. & Klausberger, T. Defined types of cortical interneurone structure space and spike timing in the hippocampus. *The Journal of physiology* **562**, 9–26 (2005).

[101] Goldstine, H. H. & Goldstine, A. The electronic numerical integrator and computer (eniac). *Mathematical Tables and Other Aids to Computation* **2**, 97–110 (1946).

[102] Hodgkin, A. L. & Huxley, A. F. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology* **117**, 500 (1952).

[103] Boss, B. D., Turlejski, K., Stanfield, B. B. & Cowan, W. M. On the numbers of neurons on fields ca1 and ca3 of the hippocampus of sprague-dawley and wistar rats. *Brain research* **406**, 280–287 (1987).

[104] West, M. J. & Gundersen, H. Unbiased stereological estimation of the number of neurons in the human hippocampus. *Journal of Comparative Neurology* **296**, 1–22 (1990).

[105] McNaughton, B. L. Cortical hierarchies, sleep, and the extraction of knowledge from memory. *Artificial Intelligence* **174**, 205–214 (2010).

[106] Teyler, T. J. & DiScenna, P. The hippocampal memory indexing theory. *Behavioral neuroscience* **100**, 147 (1986).

[107] Teyler, T. J. & Rudy, J. W. The hippocampal indexing theory and episodic memory: updating the index. *Hippocampus* **17**, 1158–1169 (2007).

[108] Goode, T. D., Tanaka, K. Z., Sahay, A. & McHugh, T. J. An integrated index: engrams, place cells, and hippocampal memory. *Neuron* **107**, 805–820 (2020).

[109] Teyler, T. J. & DiScenna, P. The role of hippocampus in memory: a hypothesis. *Neuroscience & Biobehavioral Reviews* **9**, 377–389 (1985).

[110] Marr, D., Willshaw, D. & McNaughton, B. Simple memory: a theory for archicortex. In *From the Retina to the Neocortex*, 59–128 (Springer, 1991).

[111] Girardeau, G. & Zugaro, M. Hippocampal ripples and memory consolidation. *Current opinion in neurobiology* **21**, 452–459 (2011).

[112] Buzsáki, G. Two-stage model of memory trace formation: a role for "noisy" brain states. *Neuroscience* **31**, 551–570 (1989).

[113] Williams, A. H. *et al.* Discovering precise temporal patterns in large-scale neural recordings through robust and interpretable time warping. *Neuron* **105**, 246–259 (2020).