State space discovery in spatial representation circuits with persistent cohomology

Louis Kang*[1,2], Boyan Xu†[3], and Dmitriy Morozov‡[4]

[1]Redwood Center for Theoretical Neuroscience, University of California, Berkeley
[2]Neural Circuits and Computations Unit, RIKEN Center for Brain Science
[3]Department of Mathematics, University of California, Berkeley
[4]Computational Research Division, Lawrence Berkeley National Laboratory

October 1, 2020

Abstract

Persistent cohomology is a powerful technique for discovering topological structure in data. Strategies for its use in neuroscience are still undergoing development. We explore the application of persistent cohomology to the brain's spatial representation system. We simulate populations of grid cells, head direction cells, and conjunctive cells, each of which span low-dimensional topological structures embedded in high-dimensional neural activity space. We evaluate the ability for persistent cohomology to discover these structures and demonstrate its robustness to various forms of noise. We identify regimes under which mixtures of populations form product topologies can be detected. Our results suggest guidelines for applying persistent cohomology, as well as persistent homology, to experimental neural recordings.

1 Introduction

The enormous number of neurons that compose brain circuits must coordinate their firing to operate effectively. This organization often constrains neural activity to low-dimensional manifolds, which are embedded in the high-dimensional phase space of all possible activity patterns [1, 2, 3, 4]. In certain cases, these low-dimensional manifolds exhibit nontrivial topological structure [5]. This structure may be imposed externally by inputs that are periodic in nature, such as the orientation of a visual stimulus or the direction of an animal's head. It may also be generated internally by the network itself; for example, the grid cell network constructs periodic representations of physical space which outperform non-periodic representations in several ways [6, 7, 8, 9, 10, 11, 12]. In either case, detecting and interpreting topological structure in neural data would provide insight into how the brain encodes information and performs computations.

One promising method for discovering topological features in data is persistent cohomology [13, 14, 15]. By tracking how the shape of the data changes as we examine it across different scales—thickening data points by growing balls around them—persistent cohomology detects prominent topological features in the data, such as loops and voids. This knowledge helps to identify the low-dimensional manifolds sampled by

*louis.kang@riken.jp
†boxu@berkeley.edu
‡dmitriy@mrzv.org

the data, and in particular to distinguish between tori of different intrinsic dimensions. Furthermore, it enables parametrization of the data and navigation of the underlying manifolds.

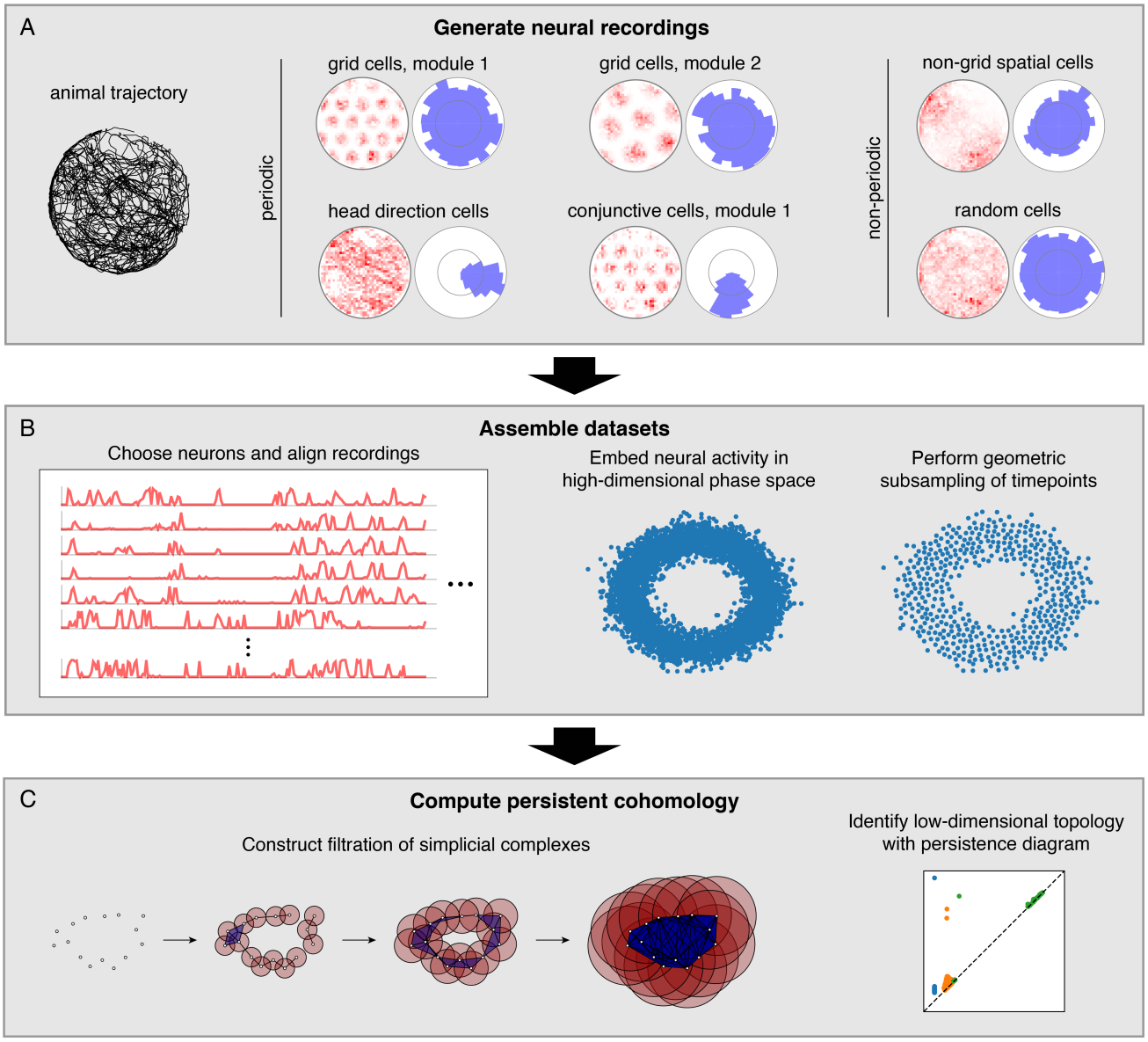

Figure 1: Pipeline for simulations and data analysis. (**A**) We generate activities for multiple neural populations along an experimentally recorded rat trajectory. For each population, we plot activity maps as a function of position (left) and direction (right) for one example neuron. (**B**) Then we choose neurons for topological analysis and form a high-dimensional vector of their firing rates at each timepoint along the trajectory. For computational tractability, we eliminate the most redundant points using a geometric subsampling algorithm. (**C**) We compute persistent cohomology on these subsampled timepoints to identify low-dimensional topological structure.

We characterize how persistent cohomology can discover topological structure in neural data through

simulations of the brain's spatial representation system. This system contains several neural populations whose activity exhibits nontrivial topology, which we term *periodic neural populations* (Fig. 1A). Grid cells fire when an animal reaches certain locations in its environment that form a triangular lattice in space [16]. In each animal, grid cells are partitioned into 4–10 modules [17]. Within each module, grid cells share the same scale and orientation but their lattices have different spatial offsets. Modules appear to increase in scale by a constant ratio and exhibit small differences in orientation [17, 18]. Head direction cells fire when an animal's head is oriented in a certain direction relative to its environment [19]. They respond independently of the animal's position. Finally, conjunctive grid $\times$ head direction cells respond when an animal is located at the vertices of a triangular lattice and is oriented in a certain direction [20]. Like grid cells, conjunctive cells are also believed to be partitioned into modules.

We also consider neural populations whose activity exhibits trivial topology, which we will term *non-periodic neural populations* (Fig. 1A). Place and non-grid spatial cells are part of the spatial representation system, and they fire in one or multiple regions of the environment [21, 22, 23]. These two populations are found in different brain regions, and the former tend to have sharper spatial selectivity compared to the latter. Finally, we simulate neurons with irregular activity that exhibits no spatial tuning. We imagine these *random cells* may be responding to non-spatial stimuli or representing internal brain states.

Our use of simulated data enables us to comprehensively explore the capabilities of our methodology. With complete control over the data, we can identify features that improve topological discovery and features that disrupt it. We can also freely generate datasets with varied quantities and proportions of different neural populations. A greater number of neurons embeds underlying activity manifolds in higher dimensions, which strengthens the signal. However, experimental limitations impose bounds to this number. Our simulations allow us to evaluate persistent cohomology in regimes currently accessible by experiments, as well as in regimes that may soon become experimentally tractable due to advances in recording technology [24]. Moreover, by combining neurons from different populations, we demonstrate the ability of persistent cohomology to succeed without the pre-processing step of neural classification.

2 Results

2.1 Overview of methods and persistence diagrams

In this work, we simulate neural populations within the spatial representation system, prepare the simulated data for topological analysis, and compute persistent cohomology to discover topological structure within the data (Fig. 1). We will now briefly describe each of these three stages; a complete explanation is provided in the Methods section.

To generate neural recordings, we define tuning curves as a function of position and direction. For each grid module, we first create a triangular lattice in space. Each grid cell has peaks in its positional tuning curves at a randomly chosen offset from each lattice point. Its directional tuning curve is uniform. Head direction cells have peaks in their directional tuning curves at a randomly chosen angle and have uniform positional tuning curves. Conjunctive cells have positional tuning curves like grid cells and directional tuning curves like head direction cells. We describe tuning curves for the non-periodic neural populations in the Methods section.

These tuning curves are applied to an experimentally extracted trajectory of a rat exploring its circular enclosure, producing an activity, or firing rate, for each neuron at 0.2 s intervals. This simulates the simultaneous recording of a large number of neurons from the medial entorhinal cortex and the binning of their spikes into firing rates. The time series span 1000 s, or 5000 data points. Figure 1A shows examples of these time series data mapped back onto spatial coordinates.

Next, we choose a subset of these neurons and pre-process it for topological data analysis (Fig. 1B). We form a vector of neural activities at each timepoint, which produces a point cloud in high-dimensional phase space. We wish to subsample it for computation tractability while maintaining as much evidence of topological structure embedded within it. To do so, we use a geometric subsampling algorithm that roughly eliminates the most redundant points (see the Methods section for a complete description), reducing the 5000 timepoints down to 1000.

Finally, we apply persistent cohomology to this subsampled point cloud (Fig. 1C). We describe this technique colloquially here, in terms of its dual, persistent homology. Both produce the same persistence diagrams, but we use cohomology throughout the paper both because it is faster to compute and because it allows us to parametrize the data, as described in the next subsection. See the Methods section for a precise description.

From the point cloud, we form a Vietoris–Rips filtration, which is a nested sequence of simplicial complexes. Each complex consists of all cliques in the near-neighbor graph, which contains all edges between points at distance at most r apart. As the threshold r increases, more edges enter the graph, and more cliques enter the Vietoris–Rips complex. Throughout this process, cycles (e.g., 1-dimensional loops) appear and get filled in the complex (Fig. 2A). There is a unique way to pair the distance thresholds at which cycles are born and die.

All such birth and death distances are collected into a persistence diagram (Fig. 2B). The points farthest from the diagonal correspond to the most persistent cycles that appear for the longest range of distance thresholds. They recover topological structure in the space sampled by the point cloud, which corresponds to the processes underlying the data—in our case, the spatial representation networks and external inputs. Persistent (co)homology is stable: the persistent points will remain in the diagram if we make small changes to the data, such as selecting slightly different timepoints or perturbing their values by a small amount of noise. The points closest to the diagonal would appear even if the processes underlying the data lack topological structure, and they are usually interpreted as noise.

The process we described above keeps track of cycles of different dimension (Fig. 2C). Besides the loops (1-dimensional cycles), it tracks connected components (0-dimensional cycles), voids (2-dimensional cycles) and higher-dimensional topological features, which lack a colloquial name. The number of independent k-cycles is called the k-th Betti number and is a topological invariant of a space. We can infer the topology of a dataset by comparing the number of persistent k-cycles to the k-th Betti numbers of conjectured ideal spaces, such a circle or a torus. Note that for every dataset, the 0-(co)cycle corresponding to the entire point cloud will never die, so we consider its death distance to be infinity.

2.2 Persistent cohomology for periodic neural populations

Each periodic neural population spans a particular topological space. We recover these relationships when we compute persistent cohomology of our simulated data (Fig. 2D–F). Each grid cell is active at one location

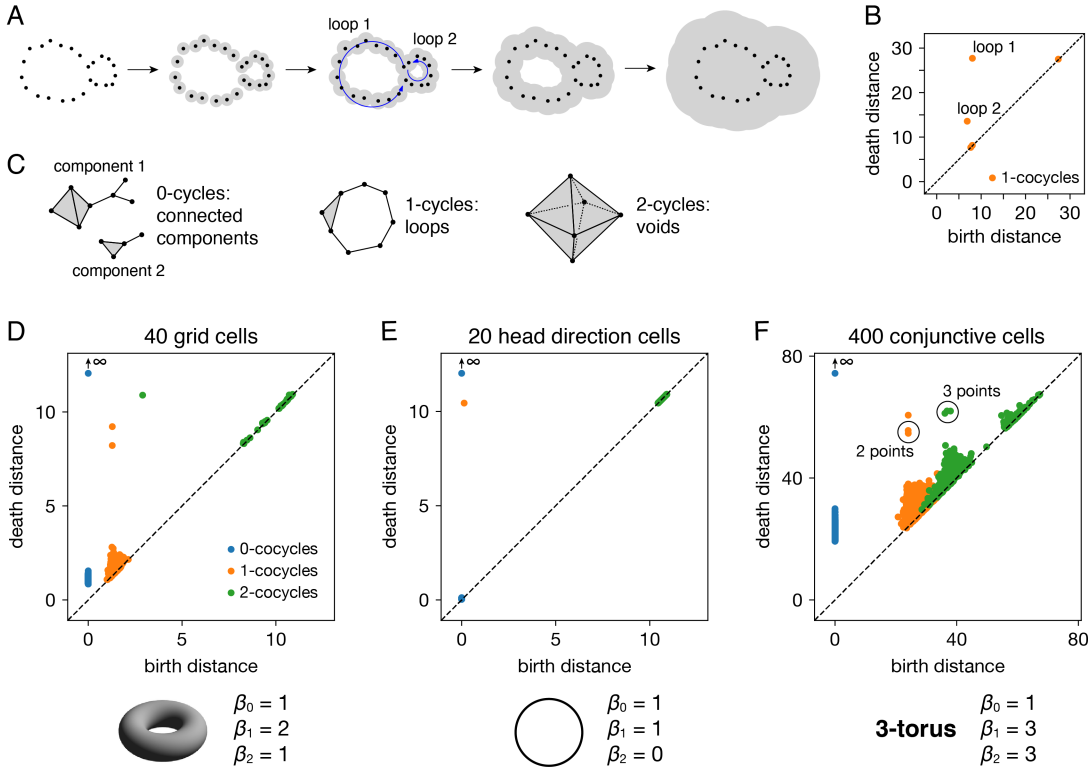

Figure 2: Persistence diagrams for periodic neural populations. (**A**–**C**) Interpreting persistence diagrams. (**A**) Persistent (co)homology involves generating a filtration of a dataset in which we connect points within ever-expanding distances. One-dimensional loops will be born and then die at various distances. (**B**) These values are plotted in a persistence diagram where each point corresponds to a single loop. The points located farthest from the diagonal are the most persistent and correspond to significant topological features. The points located close to the diagonal correspond to sampling noise. (**C**) This process can be performed for features of arbitrary dimension k, called k-(co)cycles, where $k = 0$, 1, and 2 correspond respectively to connected components, loops, and voids. (**D**–**F**) Persistence diagrams for periodic neural populations (top) and identified topological spaces (bottom). We compare the number of persistent k-(co)cycles to the k-th Betti numbers β_k of different topological spaces to infer the underlying topological structure of the dataset. (**D**) Grid cells from module 1 exhibit one persistent 0-cocycle, two persistent 1-cocycles, and one persistent 2-cocycle, which corresponds to a torus. (**E**) Head direction cells exhibit one persistent 0-cocycle, one persistent 1-cocycle, and no persistent 2-cocycles, which corresponds to a circle. (**F**) Conjunctive cells exhibit one persistent 0-cocycle, three persistent 1-cocycles, and three persistent 2-cocycles, which corresponds to a 3-torus.

in a rhombic unit cell that is tiled over 2D space (Fig. 3A). Grid cells within a single module share the same unit cell but differ in their active location, so each grid module spans a torus [5]. Similarly, head direction cells span a circle and each conjunctive cell module spans a 3-torus. The correspondence between our results and predicted topological spaces validates the basic capabilities of our methods.

Persistent cohomology can not only discover topological structure in neural data, but it can also decode information embedded within this structure. Using the grid cell population presented in Fig. 2D as an exemplar, we can assign circular coordinates [15, 25] from the two persistent 1-cocycles (Fig. 3B). This periodic topological space should represent the rhombic unit cell of the grid module, which tiles physical

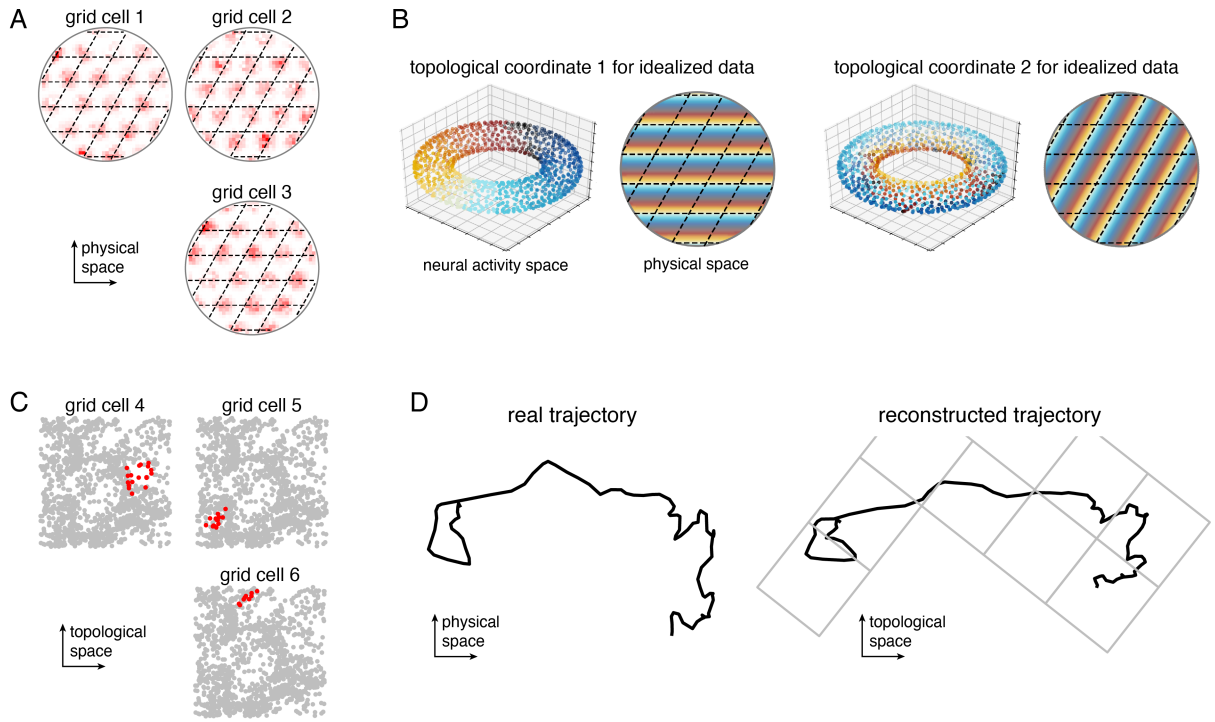

Figure 3: Correspondence between spatial and topological coordinates for a module of grid cells. (**A**) Positional activity maps for several grid cells in module 1. The module tiles physical space with a rhombic unit cell (black dashed lines), and each grid cell has its activity peaks shifted by a different offset within the unit cell. (**B**) Circular coordinates that define a topological space (left) and their mapping onto physical space (right) for the two persistent 1-cocycles of a grid module, illustrated for an idealized dataset. (**C**) Firing fields in the topological space of neural activities. We identify data points (red) for which a particular grid cell is the most active in the population. These points are clustered in topological space, similarly to how grid cells exhibit localized firing fields in physical space. (**D**) Reconstructing a segment of animal trajectory in the topological space of neural activities. We trace the time series of neural activity through topological space and unfold it along the circular coordinates (right). This tracing agrees well with the segment of real trajectory through physical space (left).

space. To explore this relationship, we project the entire time series of neural activities onto these coordinates. For each neuron, we find the data points for which that neuron is the most active within the population. These points are clustered in topological space (Fig. 3C), which means that persistent cohomology can indeed recover the firing fields of grid cells in topological space. We then analyze the population dynamics. The animal's trajectory in physical space corresponds to a trajectory through the point cloud in activity space. We find that the projection of this trajectory on the circular coordinates matches well with the real animal trajectory (Fig. 3D). Thus, persistent cohomology can decode the behavior of an animal in a recording of its neural activity.

The ability of persistent cohomology to discover topological structure depends on the number of neurons in the dataset, or equivalently, the dimension of the time series embedding. Again using the grid cell population as an exemplar, we form multiple datasets with randomly selected neurons to measure the success rate of persistent cohomology as a function of neuron count (Fig. 4A). To measure success, we only use the first

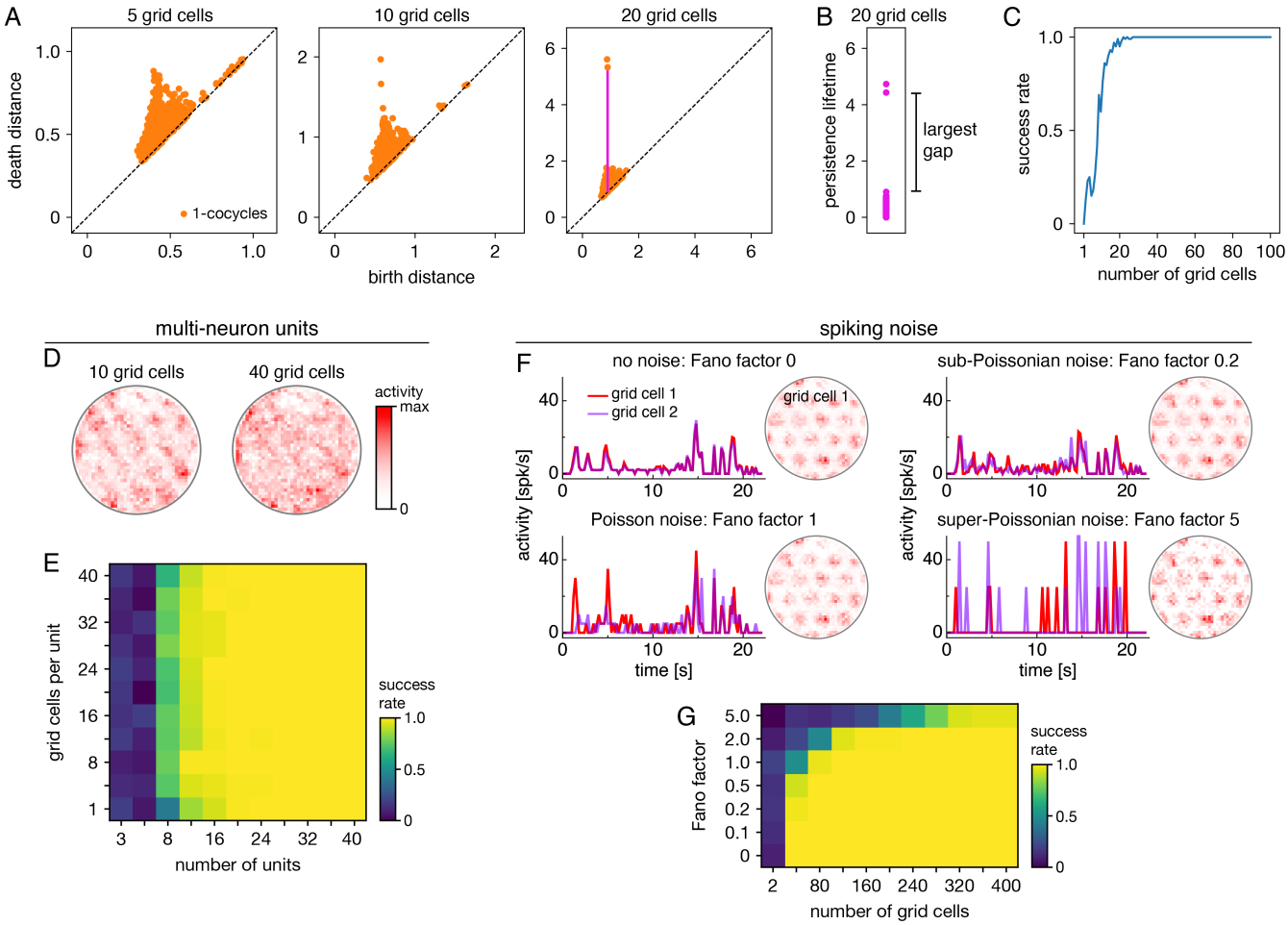

Figure 4: Success rates of persistent cohomology for grid cells. (**A**–**C**) Defining a success metric using the *largest-gap heuristic*. (**A**) As the number of grid cells increases, two persistent 1-cocycles emerge. To define persistence precisely, we consider the persistence lifetime of each 1-cocycle, which is the difference between its birth and death distances (length of the magenta line). (**B**) We identify the largest gap in persistence lifetimes and define persistent cocycles as those above this gap. (**C**) Since a grid module has a toroidal topology for which $\beta_1 = 2$, we define success as a persistence diagram with two persistent 1-cocycles. We determine success rates by generating multiple sets of neurons. Reliable discovery can be achieved around approximately 20 neurons. (**D,E**) Linearly combining activities from multiple neurons before applying persistent cohomology. (**D**) Example positional activity maps for multi-neuron units. (**E**) The success rate is largely independent of the number of neurons per multi-neuron unit. (**F,G**) Introducing Poisson-like spiking noise with different amounts of variability, as measured by the Fano factor. (**F**) For each Fano factor, time series for two grid cells with very similar offsets (left) and positional activity map of the first (right). (**G**) Persistent cohomology requires more neurons to achieve success with higher Fano factors.

cohomology group H^1, which contains 1-cocycles. We define successful discovery of the grid cell torus as a persistence diagram with two persistent 1-cocycles, and we define what it means to be persistent precisely using the *largest-gap heuristic*. We calculate the lifetime of each cocycle, which is the difference between its death and birth and corresponds to the vertical distance to the diagonal of its point in the persistence diagram (Fig. 4A,B). We find the largest gap in the lifetimes and consider points above this gap to be

persistent (Fig. 4B). Figure 4C shows that reliable discovery of the torus using this heuristic can be achieved with ≈20 grid cells.

Persistent cohomology can succeed for mixed signals. Separation of raw electrode recordings into single-neuron spike trains may not always be possible or desired. To address this scenario, we form multi-neuron units by linearly combining time series of neural activity across different grid cells. The mixing coefficients are drawn from a uniform random distribution and then normalized. Example activity maps of these multi-neuron units as a function of position are shown in Fig. 4D. The combination of many neurons destroys the classic responses exhibited by individual grid cells. Yet, multi-neuron units retain topological information associated with the grid module that can be recovered by persistent cohomology (Fig. 4E). The success rate for discovering toroidal topology is remarkably independent of the number of grid cells in each unit.

Persistent cohomology can also succeed in the presence of spiking noise. To simulate such noise, we use our generated activity as a raw firing rate that drives a Poisson-like random process (see Methods). We construct this process to have different Fano factors, which is the variance in the random process for a given firing rate divided by the firing rate. When the Fano factor is 1, the random process is Poisson. Figure 4F shows activity time series for two grid cells that have very similar tuning curves and thus very similar raw firing rates, which can be seen in the noise-free condition (top left). Higher Fano factors lead to more variability both across time for each neuron and across neurons. Persistent cohomology can still recover the toroidal topology of the grid module, though more neurons are required for higher Fano factors (Fig. 4G). In the mammalian cortex, Fano factors lie around ∼1.0–1.5 [26]. Applying this regime to our simulations implies that ≈80 grid cells are required for reliable topological discovery.

2.3 Persistent cohomology for mixtures of neural populations

Persistent cohomology can discover topological structure in mixtures of neural populations. When neurons are recorded from a periodic neural population and a non-periodic neural population, the latter adds additional dimensions to the point cloud embedding, but the topological structure contained within the former should persist. We test if persistent cohomology can recover this information in mixed datasets with neurons from both a periodic population (either grid or conjunctive) and a non-periodic population (either non-grid spatial or random). Reliable discovery of the grid module torus is possible when the number of spatial or random cells is less than twice the number of grid cells (Fig. 5A,B). Detection of the 3-torus formed by conjunctive cells requires more neurons, but it can also be reliably achieved in the presence of non-periodic populations (Fig. 5C,D). Thus, persistent cohomology demonstrates robustness to the inclusion of non-periodic populations. The size of the non-periodic population that can be tolerated increases with the size of the periodic population.

When neurons are recorded from multiple periodic neural populations, their structures are preserved along separate subspaces in high-dimensional activity space. We explore this scenario by forming mixed datasets with neurons from two periodic populations. When the two populations respond to unrelated signals—such as grid and head direction cells—the combined topological space should be the Cartesian product of those of the separate populations. Indeed, that persistent cohomology can discover the resultant 3-torus at intermediate mixing ratios (Fig. 6A,B). If one population contributes many more neurons—and thus embedding dimensions—than the other, we instead detect the corresponding single-population structure (Fig. 6B).

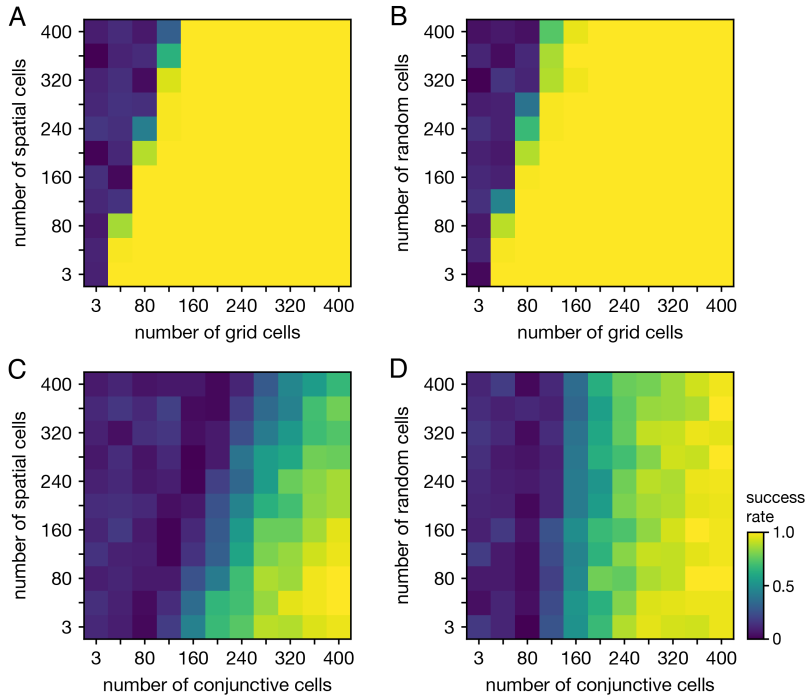

Figure 5: Persistent cohomology in combinations of periodic and non-periodic neural populations. Success is defined by discovering the number of persistent 1-cocycles expected from the periodic population, which is two for grid cells and three for conjunctive cells. (**A**) Grid cells from module 1 and non-grid spatial cells. (**B**) Grid cells from module 1 and random cells. (**C**) Conjunctive cells from module 1 and non-grid spatial cells. (**D**) Conjunctive cells from module 1 and random cells.

When the two populations respond to related signals—such as grid and conjunctive cells—the activity space of one is contained in the activity space of the other. Grid cells and conjunctive cells from the same module encode position with the same toroidal structure; they both tile space with the same rhombic unit cell of neural activity. In addition, the conjunctive population encodes direction with a circular topology. Thus, the mixed dataset should span a 3-torus, which can be detected by persistent cohomology (Fig. 6C). For reliable discovery, at least ≈120 conjunctive cells and at least ≈240 total neurons are required. However, discovery of the product topology is disrupted if the number of grid cells exceeds the number of conjunctive cells by more than a factor of ≈1.5. Thus, persistent cohomology can best detect product topologies when the mixed dataset is not dominated by one population.

Finally, we consider the case of mixing grid cells from multiple modules. Grid modules have different rhombic unit cells with different scales and orientations, so they map the same physical space onto different topological coordinates. Thus, a mixed dataset from two different modules should exhibit the product topology of two 2-tori, which is the 4-torus. However, we are unable to reliably discover this structure using the grid modules illustrated in Fig. 1A; they are too sparse. To produce a point cloud that embeds the toroidal structure for one grid module, the animal trajectory should densely sample its rhombic unit cell. This is achieved since the enclosure contains many unit cells. However, to produce a point cloud that embeds the 4-torus formed by two grid modules, the animal trajectory should densely sample all combinations of unit cells. This is not achieved by the grid modules illustrated in Fig. 1A because the enclosure contains too

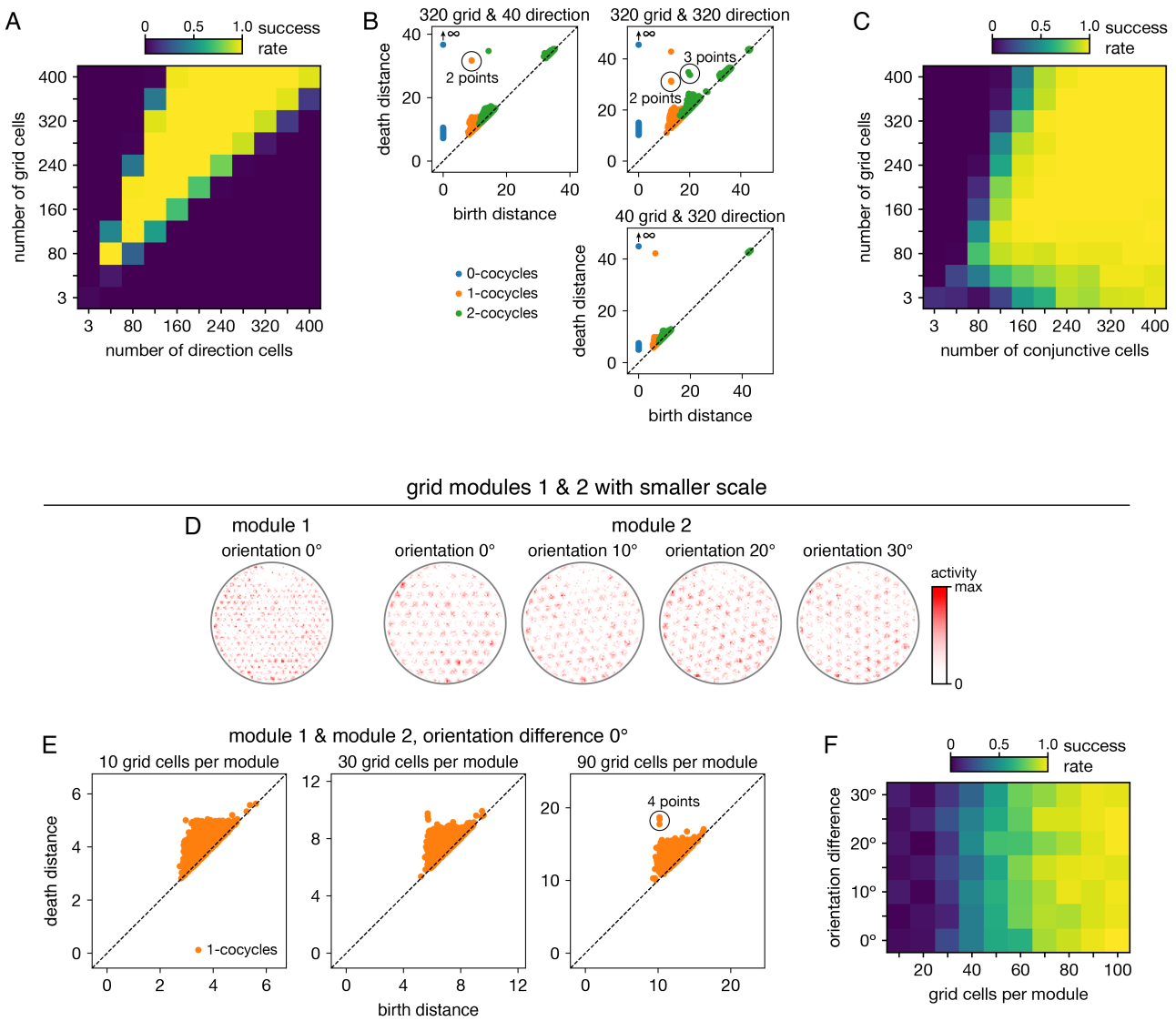

Figure 6: Persistent cohomology in combinations of periodic neural populations. Success is defined by discovering the number of persistent 1-cocycles expected from the product topology. (**A,B**) Discovering the 3-torus for grid cells from module 1 and head direction cells. (**A**) Success rates. (**B**) Example persistence diagrams. If one population contributes many more neurons than the other, persistent cohomology detects the topology of the predominant population. (**C**) Success rates for discovering the 3-torus for grid cells and conjunctive cells, both from the same module 1. (**D–F**) Combining grid cells from two modules with smaller scales. (**D**) Positional activity maps for example neurons from module 1 and four different orientations of module 2. (**E**) Example persistence diagrams. (**F**) Success rates for discovering the 4-torus.

few rhombic unit cells for them to overlap in many different configurations.

Thus, we generate two grid modules separated by the same scale ratio as in Fig. 1A, but with one-fourth of its scale (Fig. 6D). In addition, we explore different relative orientations between the modules by generating different orientations for module 2. Notably, these scale ratios and orientation differences are not chosen such that the two rhombic unit cells would share a simple geometric relationship with each other [27], which

would limit their possible overlap configurations. As we include more neurons from both modules into our dataset, we see that four persistent 1-cocycles eventually emerge from the points close to the diagonal that represent sampling noise (Fig. 6E). The success of persistent cohomology is independent of the orientation difference between the two modules (Fig. 6F).

3 Discussion

We demonstrate that persistent cohomology can discover topological structure in neural recordings with as few as tens of neurons from a periodic neural population (Fig. 4). From this structure, it can decode the trajectory of the animal using only the time series of neural activities (Fig. 3). It is robust to noise within these recordings and to the inclusion of neurons from non-periodic populations (Figs. 4 and 5). Furthermore, it can also discover more complex topological structures formed by combinations of periodic neural populations if each population is well-represented within the dataset (Fig. 6). These conclusions have been obtained through the analysis of spatial representation circuits, but we hope that they may guide the use of persistent (co)homology in other neural systems as well.

We have characterized the capabilities of persistent cohomology using simple simulated data, but we expect our results to generalize to real neural data. The inputs to our analysis pipeline are firing rates over 0.2 s time bins, which averages over many neurophysiological processes, including major neural oscillations found in the hippocampal region [28]. Moreover, we have found that persistent cohomology is robust to variations in firing rate that may arise from spiking or other sources of biological noise. A key requirement for generalization is a separation of two timescales. The macroscopic timescale at which topological structures are explored—here, the time required to traverse a rhombic unit cell of a grid module or 360° of head direction—must be much longer than the microscopic timescale at which neuronal activity is generated. This enables us to coarse-grain over the activity and describe it by a firing rate.

The application of persistent (co)homology to neuroscience data is still in its developing stages. Notable lines of work include: dimensionality reduction for manifold decoding in head direction cells [29, 30]; simulations of hippocampal place cells in spatial environments with nontrivial topology [31, 32, 33, 34, 35]; analysis of EEG signals, for classification and detection of epileptic seizures [36, 37] and for construction of functional networks in a mouse model of depression [38]; inferring intrinsic geometric structure in neural activity [39]; and detection of coordinated behavior between human agents [40]. There is potential for persistent (co)homology to provide insight to a wide range of neural systems. Topological structures generally can be found wherever periodicities exist. These periodicities can take many forms, such as the spatial periodicities in our work, temporal regularities in neural oscillations, motor patterns, and neural responses to periodic stimuli.

The toolbox of topological data analysis has more methods beneficial to the analysis of neural data. The methods described in this paper, including geometric subsampling, are sensitive to outliers. This problem can be addressed within the same framework of persistent cohomology by using distance-to-measure function [41]. In practice, this would translate into a slightly more elaborate construction [42] of the Vietoris–Rips complex. Furthermore, our analysis pipeline benefits from having neural activity embedded in a high dimensional space, i.e., from having many more neurons than the intrinsic dimensions of the recovered tori. It is possible to adapt this technique to the regime of limited neural recordings (even to a single neuron) by using time-

delay embeddings [43]. However, for spatial populations, such a technique would require control over the smoothness of the animal's trajectory, which may not be feasible in practice.

Our results also suggest research directions in topological data analysis. Throughout the paper we relied on 1-dimensional persistent cohomology to infer whether we recovered a particular torus. But that is a relatively weak method: many topological spaces have cohomology groups of the same dimension. Although the trajectories that we recover via circular coordinates serve as a convincing evidence that we are indeed recovering the tori, it is possible to confirm this further by exploiting cup product structure in cohomology, which is a particular kind of a topological operation that turns cohomology into a ring. Computing a "persistent cup product" would provide additional evidence about the structure of the recovered spaces.

4 Methods

4.1 Generating neural recordings

4.1.1 Animal trajectory

We simulate the simultaneous recording of neurons from a rat as it explores a circular enclosure of diameter 1.8 m. We use 1000 s from a trajectory extracted from an experimental animal [16, 44]. This trajectory is provided as velocities sampled at 0.5 ms intervals along with the initial position. We perform subsampling to produce a position and direction at 0.2 s intervals as follows. The position of the animal is simply the average position within each 0.2 s time bin. The direction of the animal is the circular mean of the velocity vector angle within each 0.2 s time bin; we ignore the distinction between body direction and head direction.

4.1.2 Periodic neural populations

We generate tuning curves as a function of position and/or direction for each neuron. These localized tuning curves are based on a shifted and truncated cosine function:

$$f(z) = \begin{cases} \dfrac{1+\cos \pi z}{2} & |z| < \frac{1}{2} \\ 0 & |z| \geq \frac{1}{2}. \end{cases} \tag{1}$$

For each grid module, we set a scale l and an orientation ϕ. This defines a transformation matrix from the space of phases $\left[-\frac{1}{2}, \frac{1}{2}\right) \times \left[-\frac{1}{2}, \frac{1}{2}\right)$ to the rhombic unit cell of the grid module in physical space:

$$\mathsf{A} = l \begin{pmatrix} \cos\phi & \cos\left(\phi + \frac{\pi}{3}\right) \\ \sin\phi & \sin\left(\phi + \frac{\pi}{3}\right) \end{pmatrix}. \tag{2}$$

The inverse of this matrix A^{-1} maps the rhombic unit cell onto the space of phases. We also define $\|\cdot\|$ as the vector norm and $\langle a \rangle_m \equiv (a + m \mod 2m) - m$ as the shifted modulo function. The tuning curve of a grid cell as a function of position $\mathbf{x}$ is then

$$s_{\text{grid}}(\mathbf{x}; \mathbf{b}) = f\left(\tfrac{1}{0.45l} \left\| \mathsf{A} \langle \mathsf{A}^{-1}\mathbf{x} - \mathbf{b} \rangle_{1/2} \right\|\right) \tag{3}$$

Each grid cell is shifted by a uniformly random phase offset $\mathbf{b}$. The full width at half maximum of each grid field is $0.45l$.

The tuning curve of a head direction cell as a function of direction θ is

$$s_{\mathrm{dir}}(\theta; c) = f\big(4\langle\theta - c\rangle_\pi\big). \tag{4}$$

Each head direction cell is shifted by a uniformly random direction offset c. The full width at half maximum of the head direction field is $\pi/2$.

The tuning curve of a conjunctive cell is simply the product

$$s_{\mathrm{conj}}(\mathbf{x}, \theta; \mathbf{b}, c) = s_{\mathrm{grid}}(\mathbf{x}; \mathbf{b}) s_{\mathrm{dir}}(\theta; c). \tag{5}$$

Each conjunctive cell has uniformly random offsets $\mathbf{b}$ and c.

4.1.3 Non-periodic neural populations

For non-grid spatial cells, we generate tuning curves as a function of position for each neuron of the form

$$s_{\mathrm{spatial}}(\mathbf{x}) = \frac{1}{2}\Big(\mathrm{e}^{-\|\mathbf{x}-\mathbf{d}_1\|^2/2\sigma^2} + \mathrm{e}^{-\|\mathbf{x}-\mathbf{d}_2\|^2/2\sigma^2}\Big), \tag{6}$$

where $\sigma = 40\,\mathrm{cm}$ and the $\mathbf{d}_i$'s are each chosen uniformly randomly between $(0\,\mathrm{cm}, 0\,\mathrm{cm})$ and $(180\,\mathrm{cm}, 180\,\mathrm{cm})$.

For random neurons, we obtain activity time series by sampling from a distribution every 2 s, or 10 timepoints, and interpolating between using cubic polynomials. The distribution is Gaussian with mean 0 and width 0.5, truncated between 0 and 1.

4.1.4 From tuning curves to time series

To obtain activity time series for all populations except for random neurons, we apply the tuning curves to the subsampled trajectory. Whenever the velocity decreases below 5 cm/s, we set the activity to be 0. This threshold simulates the behavior of neurons in the hippocampal region that exhibit high activity during locomotion and low activity during idle periods [20, 45, 46].

4.1.5 Spiking noise for grid cells

The activities described above are dimensionless, and we typically do not need to assign a scale because we divide each time series by its mean before applying persistent cohomology. To create spiking noise, however, we must set the firing rate. We linearly rescale the rate given by Eq. 3:

$$\lambda = 0.4 + 7.6 s_{\mathrm{grid}}. \tag{7}$$

This sets the maximum firing rate to be 8 and creates a baseline rate of 0.4; with 0.2 s time bins, these values correspond to 40 Hz and 2 Hz, respectively. However, we still set the firing rate to 0 Hz when the animal's velocity decreases below 5 cm/s.

Using λ, we generate Poisson-like spiking noise with different levels of variability (Fig. 4). At each timestep, the noisy activity is given by

$$s_{\text{noisy}} = F \cdot X, \qquad X \sim \text{Pois}(\lambda/F). \tag{8}$$

F sets the Fano factor of the random process, which is its variance divided by its mean (for any given λ). The $F = 1$ case corresponds to a Poisson random process; $F < 1$ implies sub-Poissonian noise and $F > 1$ implies super-Poissonian noise.

4.1.6 Multi-neuron units for grid cells

We generate multi-neuron units (Fig. 4) by linearly combining activity time series from multiple grid cells. Each mixing coefficient is chosen from a uniform random distribution between 0 and 1. The activity is then normalized by the sum of squares of the mixing coefficients.

4.2 Neural activity maps

We construct activity maps for each neuron as a function of position or direction. To do so, we simply tally the total amount of activity in each positional or directional bin. Note that these maps do not depict firing rate because we do not divide by the occupancy of each bin; we decided against this in order to show the activity experienced through the animal trajectory.

4.3 Processing neural recordings

For each neuron, we first divide its activity at every timepoint by its mean activity.

To improve the computational efficiency, we reduce the number of input points while preserving their topological structure by applying a geometric subsampling strategy. We pick the first point at random, and then iteratively add a point to our subsample that is the furthest away from the already chosen points. Specifically, if P is the input point set and Q_i is the subsample after i iterations, we form Q_{i+1} by adding the point q_{i+1} chosen as

$$\arg\max_{p \in P} \min_{q \in Q_i} \|p - q\|. \tag{9}$$

Fig. 1B illustrates a result of this strategy. By construction, the subsample Q_i forms an ε_{i+1}-net of the input point sample, which means the largest distance from any input point to the nearest point of the subsample does not exceed $\varepsilon_{i+1} = \min_{q \in Q_i} \|q_{i+1} - q\|$. Because persistent cohomology is stable, this guarantees that the persistence diagram we compute for the subsample Q_i is at most ε_{i+1} away, in bottleneck distance [47], from the persistence diagram of the full point set P.

4.4 Applying persistent cohomology

We refer the reader to extensive literature on persistent (co)homology [13, 14] for the full details, and only sketch a few of the involved constructions.

For technical reasons—both to recover the circular coordinates and for computational speed—we work with persistent cohomology, which is dual to persistent homology, which a reader might be more familiar

with.

To recover the topology of the space sampled by a point set P, we construct a Vietoris–Rips simplicial complex. A simplex σ is a subset of the point set P. A simplicial complex is a collection of simplices closed under the subset relation, i.e., if K is a simplicial complex and $\sigma \in K$, then for every $\tau \subset \sigma$, $\tau \in K$. Given a parameter r, Vietoris–Rips complex consists of all subsets of the point set P, in which every pair of points is at most r away from each other,

$$\mathrm{VR}(P, r) = \{\sigma \subseteq P \mid \|p - q\| \leq r \ \forall p, q \in \sigma\}. \tag{10}$$

The dimension of a simplex is one less than its cardinality, $\dim \sigma = \operatorname{card} \sigma - 1$. 0-simplices are called vertices; 1-simplices, edges; 2-simplices, triangles; etc. A coboundary of a k-simplex is the alternating sum of its supersets of dimension $k+1$, denoted $\delta\sigma$. A k-cochain is a collection of k-dimensional simplices; its coboundary is the sum of the coboundaries of its simplices. A k-cochain is called a k-cocycle if its coboundary is 0. A k-cocycle is a k-coboundary if it is a coboundary of some $(k-1)$-cochain. Two k-cocycles are said to be cohomologous if their difference is a k-coboundary. A cohomology equivalence class consists of all cohomologous k-cocycles. The cohomology group, $\mathsf{H}^k(\mathrm{VR}(P, r))$, consists of all cohomology classes. The rank of this group is called the k-th Betti number; intuitively, it measures the number of "holes" in the space. When we compute the coboundaries with coefficient in a field, the cohomology group is a vector space. (In all our experiments we use $\mathbb{Z} \mod 3$ as the field.)

As we vary the radius r in the definition of the Vietoris–Rips complex, the simplicial complexes nest: $\mathrm{VR}(P, r_1) \subseteq \mathrm{VR}(P, r_2)$, for $r_1 \leq r_2$. The restriction of the larger complex to the smaller induces a linear map on cohomology groups, and all such maps form a sequence:

$$\mathsf{H}^k(\mathrm{VR}(P, r_1)) \leftarrow \mathsf{H}^k(\mathrm{VR}(P, r_2) \leftarrow \mathsf{H}^k(\mathrm{VR}(P, r_3)) \leftarrow \ldots \tag{11}$$

It is possible to track when cohomology classes appear and disappear in this sequence. Specifically, it is possible to decompose this sequence into a collection of birth–death pairs (r_i, r_j) that specify that a class born in $\mathsf{H}^k(\mathrm{VR}(P, r_i))$ dies in $\mathsf{H}^k(\mathrm{VR}(P, r_j))$. This collection of pairs, called a persistence diagram, completely describes the changes in the sequence of cohomology groups.

Acknowledgments

We would like to thank Francesco Fumarola for helpful suggestions. L.K. has been supported by a Miller Research Fellowship from the Miller Institute for Basic Research in Science at the University of California, Berkeley. B.X. is supported by a U.S. Department of Energy Computational Science Graduate Fellowship. This work was partially supported by the U.S. Department of Energy under Contract Number DE-AC02-05CH11231 at Lawrence Berkeley National Laboratory.

References

[1] Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current Opinion in Neurobiology*, 32:148–155, 2015.

[2] Juan A. Gallego, Matthew G. Perich, Lee E. Miller, and Sara A. Solla. Neural Manifolds for the Control of Movement. *Neuron*, 94(5):978–984, 2017.

[3] Mehrdad Jazayeri and Arash Afraz. Navigating the Neural Space in Search of the Neural Code. *Neuron*, 93(5):1003–1014, 2017.

[4] Shreya Saxena and John P Cunningham. Towards the neural population doctrine. *Current Opinion in Neurobiology*, 55:103–111, 2019.

[5] Carina Curto. What can topology tell us about the neural code? *Bulletin of the American Mathematical Society*, 54(1):63–78, 2016.

[6] I R Fiete, Y Burak, and T Brookings. What Grid Cells Convey about Rat Location. *Journal of Neuroscience*, 28(27):6858–6871, 2008.

[7] Sameet Sreenivasan and Ila Fiete. Grid cells generate an analog error-correcting code for singularly precise neural computation. *Nature Neuroscience*, 14(10):1330–1337, 2011.

[8] Alexander Mathis, Andreas V M Herz, and Martin Stemmler. Optimal Population Codes for Space: Grid Cells Outperform Place Cells. *Neural Computation*, 24(9):2280–2317, 2012.

[9] Martin Stemmler, Alexander Mathis, and Andreas V M Herz. Connecting multiple spatial scales to decode the population activity of grid cells. *Science Advances*, 1(11):e1500816–e1500816, 2015.

[10] X-X Wei, J Prentice, and V Balasubramanian. A principle of economy predicts the functional architecture of grid cells. *eLife*, 4:e08362, 2015.

[11] A. Sanzeni, V. Balasubramanian, G. Tiana, and M. Vergassola. Complete coverage of space favors modularity of the grid system in the brain. *Physical Review E*, 94(6):062409, 2016.

[12] Noga Mosheiff, Haggai Agmon, Avraham Moriel, and Yoram Burak. An efficient coding theory for a dynamic trajectory predicts non-uniform allocation of entorhinal grid cells to modules. *PLOS Computational Biology*, 13(6):e1005597, 2017.

[13] Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Soc., 2010.

[14] Herbert Edelsbrunner and Dmitriy Morozov. Persistent homology. In Jacob E Goodman, Joseph O'Rourke, and Csaba D Tóth, editors, *Handbook of Discrete and Computational Geometry*. CRC Press, 2017.

[15] Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. Persistent cohomology and circular coordinates. *Discrete & computational geometry*, 45(4):737–759, June 2011.

[16] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, 2005.

[17] Hanne Stensola, Tor Stensola, Trygve Solstad, Kristian Frøland, May-Britt Moser, and Edvard I Moser. The entorhinal grid map is discretized. *Nature*, 492(7427):72–78, 2012.

[18] Julija Krupic, Marius Bauza, Stephen Burton, Caswell Barry, and John O'Keefe. Grid cell symmetry is shaped by environmental geometry. *Nature*, 518(7538):232–235, 2015.

[19] J S Taube, R U Muller, and J B Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. *Journal of Neuroscience*, 10(2):420–435, 1990.

[20] Francesca Sargolini, Marianne Fyhn, Torkel Hafting, Bruce L McNaughton, Menno P Witter, May-Britt Moser, and Edvard I Moser. Conjunctive representation of position, direction, and velocity in entorhinal cortex. *Science*, 312(5774):758–762, 2006.

[21] J O'Keefe and J Dostrovsky. The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. *Brain Research*, 34(1):171–175, 1971.

[22] Geoffrey W Diehl, Olivia J Hon, Stefan Leutgeb, and Jill K Leutgeb. Grid and Nongrid Cells in Medial Entorhinal Cortex Represent Spatial Location and Environmental Features with Complementary Coding Schemes. *Neuron*, 94(1):83–92.e6, 2017.

[23] Kiah Hardcastle, Niru Maheswaranathan, Surya Ganguli, and Lisa M Giocomo. A Multiplexed, Heterogeneous, and Adaptive Code for Navigation in Medial Entorhinal Cortex. *Neuron*, 94(2):375–387.e7, 2017.

[24] James J Jun, Nicholas A Steinmetz, Joshua H Siegle, Daniel J Denman, Marius Bauza, Brian Barbarits, Albert K Lee, Costas A Anastassiou, Alexandru Andrei, Çağatay Aydın, Mladen Barbic, Timothy J Blanche, Vincent Bonin, João Couto, Barundeb Dutta, Sergey L Gratiy, Diego A Gutnisky, Michael Häusser, Bill Karsh, Peter Ledochowitsch, Carolina Mora Lopez, Catalin Mitelut, Silke Musa, Michael Okun, Marius Pachitariu, Jan Putzeys, P Dylan Rich, Cyrille Rossant, Wei-lung Sun, Karel Svoboda, Matteo Carandini, Kenneth D Harris, Christof Koch, John O'Keefe, and Timothy D Harris. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232–236, 2017.

[25] Jose A Perea. Sparse circular coordinates via principal $\mathbb{Z}$-Bundles. *arXiv*, September 2018.

[26] Michael N. Shadlen and William T. Newsome. The Variable Discharge of Cortical Neurons: Implications for Connectivity, Computation, and Information Coding. *Journal of Neuroscience*, 18(10):3870–3896, 1998.

[27] Louis Kang and Vijay Balasubramanian. A geometric attractor mechanism for self-organization of entorhinal grid modules. *eLife*, 8:e46687, 2019.

[28] John E Lisman and Ole Jensen. The Theta-Gamma Neural Code. *Neuron*, 77(6):1002–1016, 2013.

[29] Erik Rybakken, Nils Baas, and Benjamin Dunn. Decoding of neural data using cohomological learning. *bioRxiv*, 2018.

[30] Rishidev Chaudhuri, Berk Gerçek, Biraj Pandey, Adrien Peyrache, and Ila Fiete. The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature Neuroscience*, 22(9):1512–1520, 2019.

[31] Y Dabaghian, F Mémoli, L Frank, and G Carlsson. A Topological Paradigm for Hippocampal Spatial Map Formation Using Persistent Homology. *PLOS Computational Biology*, 8(8):e1002581, 2012.

[32] Gard Spreemann, Benjamin Dunn, Magnus Bakke Botnan, and Nils A Baas. Using persistent homology to reveal hidden information in neural data. *arXiv*, 2015.

[33] Samir Chowdhury, Bowen Dai, and Facundo Mémoli. The importance of forgetting: Limiting memory improves recovery of topological characteristics from neural data. *PLOS ONE*, 13(9):1–20, September 2018.

[34] Andrey Babichev, Dmitriy Morozov, and Yuri Dabaghian. Robust spatial memory maps encoded by networks with transient connections. *PLOS Computational Biology*, 14(9):e1006433, 2018.

[35] Andrey Babichev, Dmitriy Morozov, and Yuri Dabaghian. Replays of spatial memories suppress topological fluctuations in cognitive map. *Network neuroscience (Cambridge, Mass.)*, 3(3):707–724, 2019.

[36] Marco Piangerelli, Matteo Rucco, Luca Tesei, and Emanuela Merelli. Topological classifier for detecting the emergence of epileptic seizures. *BMC research notes*, 11(1):392, June 2018.

[37] Yuan Wang, Hernando Ombao, and Moo K Chung. Topological data analysis of Single-Trial electroencephalographic signals. *The annals of applied statistics*, 12(3):1506–1534, September 2018.

[38] Arshi Khalid, Byung Sun Kim, Moo K Chung, Jong Chul Ye, and Daejong Jeon. Tracing the evolution of multi-scale functional networks in a mouse model of depression using persistent brain network homology. *NeuroImage*, 101:351–363, 2014.

[39] Chad Giusti, Eva Pastalkova, Carina Curto, and Vladimir Itskov. Clique topology reveals intrinsic geometric structure in neural correlations. *Proceedings of the National Academy of Sciences of the United States of America*, 112(44):13455–13460, November 2015.

[40] Mengsen Zhang, William D Kalies, J A Scott Kelso, and Emmanuelle Tognoli. Topological portraits of multiscale coordination dynamics. *Journal of Neuroscience Methods*, 339:108672, 2020.

[41] Frédéric Chazal, David Cohen-Steiner, and Quentin Mérigot. Geometric inference for measures based on distance functions. *Foundations of computational mathematics*, 11(6):733–751, 2011.

[42] Leonidas Guibas, Dmitriy Morozov, and Quentin Mérigot. Witnessed k-distance. *Discrete & computational geometry*, 49(1):22–45, January 2013.

[43] Floris Takens. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*, pages 366–381. Springer Berlin Heidelberg, 1981.

[44] Yoram Burak and Ila R Fiete. Accurate Path Integration in Continuous Attractor Network Models of Grid Cells. *PLOS Computational Biology*, 5(2):e1000291, 2009.

[45] Emilio Kropff, James E Carmichael, May-Britt Moser, and Edvard I Moser. Speed cells in the medial entorhinal cortex. *Nature*, 523(7561):419–424, 2015.

[46] James R Hinman, Mark P Brandon, Jason R Climer, G William Chapman, and Michael E Hasselmo. Multiple Running Speed Signals in Medial Entorhinal Cortex. *Neuron*, 91(3):666–679, 2016.

[47] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. *Discrete & computational geometry*, 37:103–120, 2007.