NTNU
Norwegian University of
Science and Technology

Topological Detection in Spatially and Directionally Tuned Neural Network Activity

Erik Hermansen

Master of Science in Physics and Mathematics
Submission date: June 2017
Supervisor: Nils A. Baas, IMF

Norwegian University of Science and Technology
Department of Mathematical Sciences

Problem Description

The primary aim of this thesis is to apply persistent homology to neural data obtained from the simulated activity of head direction, place and grid cell networks to detect significant topological features which may describe the dynamics of the networks. This is done to study whether this method obtains additional information of how animals encode a representation of space and may serve as a measure of the correctness of neural network models.

Preface

This Master's thesis is written by Erik Hermansen from January 2017 to June 2017 at the Norwegian University of Science and Technology (NTNU). The thesis completes my Master of Science (MSc) degree in Physics and Mathematics with specialization in Industrial Mathematics.

I would like to thank my supervisor, Nils Baas, for constructive feedback and follow-up of this work. I am also grateful for the advice on neural models provided by Benjamin Adric Dunn (who, additionally, contributed with basis scripts for the head direction and grid cell CANN models). The method presented in this thesis was first proposed by Erik Rybakken, who has given me valuable information on its implementation. Lastly, I would like to thank my family and friends for their encouragement and support throughout my five years at NTNU.

Erik Hermansen
Oslo, June 2017

Summary

Persistent homology is the main tool in topological data analysis and uses methods from algebraic topology to describe the underlying space of data sets. In this thesis, persistent homology is used to detect topological characteristics of the dynamics of head direction, grid and place cell network activity. We simulate the neural activity (neuron firing rate) of the networks (chiefly) based on simple continuous attractor network models. This activity is used to generate a Poisson spike train, from which a continuous time series obtained by means of Gaussian smoothing. We construct Vietoris-Rips and flag complexes based on the n-dimensional points sampled at discrete times from the time series, where n denotes the number of neurons whose activity we study, and apply persistent homology, resulting in what is known as persistence diagrams. These may reveal the topology of the underlying space of the point cloud, in this case describe the activity of the neural networks (and thus the networks themselves).

The method is shown to produce results corresponding to that expected, supporting its efficacy of providing a way to assess the feasibility a neural network model and to understand its properties.

Sammendrag

Persistent homologi har blitt det viktigste verktøyet i topologisk dataanalyse, og benytter metoder fra algebraisk topologi for å beskrive det underliggende rommet til datasett. I denne masteroppgaven anvender vi persistent homologi for å oppdage topologiske karakteristikker ved dynamikken til hoderetnings-, gitter- og stedcellenettverksaktivitet. Vi simulerer den nevrale aktiviteten (avfyringsraten til nevroner) til nettverkene basert på enkle kontinuerlige attraktornettverksmodeller. Denne aktiviteten brukes så til å generere binære, poissonfordelte nervesignaler. En kontinuerlig tidsserie blir dernest dannet ut ifra denne dataen ved hjelp av gaussisk glatting. Etter å ha lagd Vietoris-Rips- og flaggkomplekser basert på n-dimensjonale punkter lest av ved diskrete tidspunkter fra tidsserien, hvor n tilsvarer antall nevroner vi studerer ved den gitte anledning, anvender vi så persistent homologi, som resulterer i såkalte persistensdiagrammer. Disse avdekker topologien til det underliggende rommet til punktskyen - i dette tilfellet, så beskriver det aktivitet til de nevrale nettverkene (og således nettverkene selv).

Denne metoden gir resultater som samsvarer med det forventede, noe som støtter dens kapasitet til å anslå anvendbarheten til en nevral nettverksmodell og forstå dets egenskaper.

Table of Contents

Chapter 1

Introduction

Persistent homology has in the past couple of decades emerged as a robust method to assess the topology of the underlying space of point clouds and thus enable a simple analysis of the structure of large, high-dimensional data. It serves as the most prominent method in topological data analysis (TDA), and has proven successful in various fields of study, one being that of neuroscience.

The vast amount of both external and internal (somatic) input is somehow perceived and interpreted by the neurons in our brains. Already, TDA has shown surprising results in the effort to understand how certain neural networks behave, and one of the first applications was done by Singh et al. (2008), who studied population activity in primary visual cortex (V1). By constructing weak witness complexes and looking at the resulting Betti number distributions and barcode lengths, they showed that the topological structure of activity patterns when the cortex is spontaneously active is like those evoked by natural image stimulation and consistent with the topology of a 2-sphere. However, it was the work done by Curto and Itskov (2008) who ignited the application of persistent homology to spatially tuned cells - namely, the hippocampal place cells. From this cell activity of a rat, they reconstructed the topology of the physical environment it was exploring.

This (in part) motivates our thesis, and we will look at how persistent homology may be applied to neural activity to detect the topological structures of neural networks whose activity is inherently disposed to a spatial and directional tendency.

The representation of space in the brain

We seldom reflect on our ability to navigate and localize position. However, it remains unclear exactly how the notions of space and direction are encoded in our brains, and what inner structures are involved (Langston et al. (2010)). Several different neurons in mammalian brains thought to play a part in this process have been revealed and multiple models of the intricate dynamics have been suggested. Tolman et al proposed in 1948 a connection between cognitive representation of physical space and of abstract conceptual spaces. Subsequently the *head direction*, *place* and *grid cells* were discovered, and these are thought to procure a basis for encoding this representation (see example behaviour

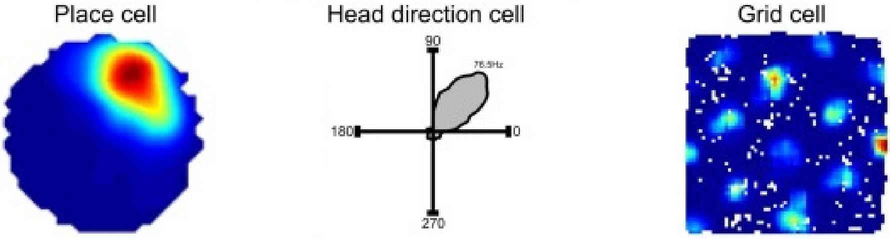


Figure 1.1: Left, the firing rate map of a place cell (the warmer the colour, the more frequent rate) recorded as a rat explored a disklike environment. A clear place field can be observed. Middle, an example head direction cell firing rate polar plot. The rate of a cell is plotted according to the animal's head direction at the time. The cell's 'preferred firing direction' seems to be about 45° in the given environment. Right, the firing rate map of a grid cell. Multiple place fields are observed which form a triangular or hexagonal grid that spans the environment (credit: Grieves and Jeffery (2017)).

for each cell in figure 1.1). Each head direction cell fires in relation to a direction which the animal's head is facing with respect to the environment. The preferred directions are uniformly distributed over the possible angles, and thus maintain a 'cognitive compass' (Taube (2009)). The place cells are preferentially active when the head of the animal is in a specific location in an environment. These were first found in the CA1 field of the hippocampus of freely moving rats by O'keefe and Nadel (1978), who proposed that the place cells encode a (hippocampal) 'cognitive map'. For instance, the representation of space for a rodent placed in a 2D box, would be planar and each place cell then fires within some distance of their individually 'preferred' xy-coordinate and silent otherwise. This region of activity is then called the place cell's *place field*. Grid cells are also related to spatial positioning, but each cell has multiple peaked firing fields forming a hexagonal pattern in an environment. In contrast to the place cells, whose locations seem to be random when put in different environments (Muller and Kubie (1987)), the relationships between the grid cells with similar coding are maintained in different environments and under manipulation (Moser and Moser (2007)).

A novel method detecting the topology of neural network activity

In this paper, we simulate the neural activity of head direction, place and grid cell networks of virtual animals in different environments by using a firing rate neuron model and simple continuous attractor network models (and, in addition, we test a self-organizing grid cell network model). We use synthetically generated data to be in control of all parameters, quickly perform different "experiments" and to study the computational model at hand. The continuous attractor models explain the network properties through the intrinsic cell connections and the neural responses of external input, and smoothly hold and translate an *activity packet* - a collective neuronal activity distribution centred at any point of the network manifold (Samansonovich (2017)).

Experimental neural activity is given in *spike trains* - binary-valued time series indicating whether the neuron(s) have fired or not. Thus, we convert the simulated activity to spike trains using a Poisson probability for firing. To generate more realistic neuronal dynamics, we transform the spike trains into continuous time-series by Gaussian smoothing, i.e. we consider each spike as normally distributed activity (over time), in which the peak is at the moment of the spike firing. This allows us to sample a firing rate value of all neurons in a network at any given time. The sampled data may then be viewed as a point cloud on which we may apply persistent homology to interpret the topology of the neural activity.

Our goal is to understand the topological structure of these networks through the *persistence diagrams* thus created and test to see if the models actually exhibit the features expected, and also look at if the neural structures remain detectable subjected to noise. Thus, the two questions we wish to answer are the following:

- What (persistent) homology does the neural network activity exhibit and is this as expected?
- How robust is the method studied?

Thesis Outline

The thesis is split into six chapters. The introduction just given is followed by a literature survey (Ch. 2) and a theoretical exposition of the neural models and persistent homology (Ch. 3). Further, the methods are described (Ch. 4), before the results are given (Ch. 5). The conclusion shortly discusses the results and indicates further work (Ch. 6).

Chapter 2

Literature Survey

We review selected papers to shed light on the current development of the use of persistent homology to extract information from neural data. Note that some terms are not explained until later in this thesis (or not at all). The literature review recently done by Giusti et al. (2016) serves as a more coherent, extensive survey of the various algebraic-topological tools used. However, the current chapter is an effort to support their statement that these tools have ”the potential to eclipse graph theory in unravelling the fundamental mysteries of cognition (p.1)”.

Hippocampal map and covariate detection

By constructing a concurrence complex based on the relative firing of place cell groups from hippocampal place cell activity (connecting coactive neurons - see figure 2.1) and applying persistent homology, Curto and Itskov (2008) were able to extract the global topological features of a 2D environment in which a rat was moving. Additionally, they reconstruct an accurate geometric representation of the space and track the animal’s position within it, only assuming the existence and stereotypical form of the place fields. Arguing that the brain can compare neural responses, they propose that this might be the way in which the brain constructs internal representations of space.

Constructing clique complexes of a graph representing the functional connectivity of place cells, Giusti et al. (2015) detect geometric organization without reference to external data such as animal behaviour. The geometric features may also be extracted from the intrinsic pattern of neural correlations of a rat undergoing non-spatial behaviour such as wheel running and REM sleep, suggesting the structure is shaped by the underlying hippocampal circuits, and not merely a consequence of position coding.

Despite these geometric findings, Dabaghian et al. (2014) propose that the hippocampal cognitive map is topological rather than geometric. They show this by changing the geometry, but not topology, of a maze and record the neural activity from place cells in a rat as it runs through the maze. The activity then remains largely unchanged consistent with their proposal.

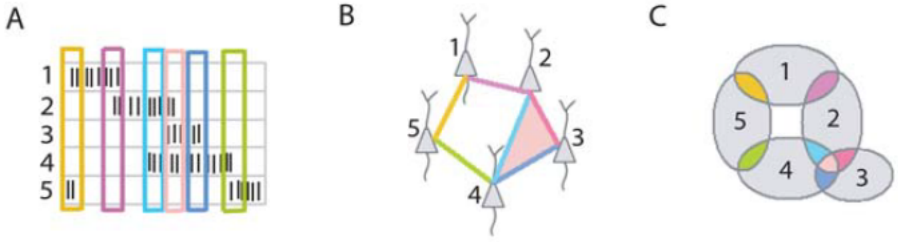


Figure 2.1: (A) Sample raster plots for the population activity of five place cells. Cell groups are obtained by identifying subsets of cells that co-fire within a binned time window (coloured rectangles). (B) The simplicial complex configuration obtained from the data in (A). An edge represents a cell group with two cells and a shaded triangle indicates a cell group with three cells. (C) Cells that co-fire have overlapping place fields. Each cell group in (A) corresponds to an intersection of place fields. The intersection pattern uniquely determines the topology of the environment. In this instance, the arrangement of place fields (re)creates a hole in the middle. (credit: Curto and Itskov (2008)).

Spreemann et al. (2015) define an order complex based on cofiring relations of simulated neural data of place and head direction cells and show that persistent homology may reveal the topological properties of a priori unknown covariates of neuron activity. Hence, they can infer away (and thus analyse) the topological contribution of each covariate. For instance, they analyse the neural activity when the only two stimuli are spatial position and head direction and observe that the persistent homology may detect both stimuli as 'hidden' covariates.

Spatial learning

Using a similar method as Curto and Itskov, Dabaghian et al. (2012) show that the neuronal cofiring patterns can convey topological information about the environment in a biologically realistic length of time. They introduce a 'learning region' which describes the parameter constraints for forming a reliable topological map of the environment, the parameters being the firing rate of the place cells, the size of the place field and the number of cells. The constraints correspond well with that experimentally observed, and thus provides insight into the process of spatial learning in novel environments and further supports the hypothesis of a topological cognitive map.

Arai et al. (2014) advance this spatial learning model by including the effects of theta phase precession (the phase shifting of place cell firing relative to the background theta oscillation in the hippocampus, corresponding to the spatial distance to a landmark (Skaggs and McNaughton (1996))), believed to influence learning and memory. They show that theta precession improves the spatial learning mechanisms, increasing both speed and the size of the learning region.

Babichev et al. (2016) construct coactivity complexes based on the functional architecture of the place cell group network, i.e. the maximal simplices are viewed as representations of physiological place cell assemblies rather than any largest combinations of the coactive place cell groups. By both selecting connectedness based on frequency of the place group co-appearance and by optimally selecting connections based on a given realistic number (e.g. a cell's number of axon terminals), the topological structure of the environment is extracted as fast and as reliably as the complete set of the place cell co-activities, again showing that the place cell population encodes the topology of a given environment in biologically plausible time.

Hoffman et al. (2016) extend the study to the three-dimensional case, looking at the formation of place field maps for virtual bats by constructing coactivity complexes. This correctly captures the topology of the three-dimensional physical environment, revealing the neurophysiological mechanisms of hippocampal spatial learning. For instance, by suppressing theta precession (consistent with experiments), they show that bats improve the place cells capacity to encode spatial maps.

Summary and relation to thesis

The aim of this chapter was to provide an understanding of the fact that persistent homology has been successfully applied in neuroscience and of what results it has procured. Persistent homology seems to be vital in furthering the understanding of the encoding of a representation of space in the brain. It enables a way to interpret the place cell firing, creating a topological map in both 2D and 3D environments, as well as detecting known covariates and possibly revealing information about unknown covariates. We have also discussed how it has provided new insights into spatial learning and the curious phenomenon of theta phase precession. As these discoveries has proved biologically plausible, we are further supported in our notion that persistent homology is a unique method well-suited for the analysis of neural data.

Although the focus of the papers reviewed in this chapter is on place cells, it suggests the applicability of the method to neural data from head direction cells and grid cells (as studied in this thesis). We will give a few examples of the place cell encoding of a topological map later in this thesis, but do not study the spatial learning process connected to this. The network activity studied is extended to 3D in all (three) cases as motivated by Hoffman et al. The possibility of constructing complexes based on functional architecture was appealing, as was the method of covariate detection, but time did not permit these indulgences (although, we do study neural activity collectively accounting for direction and spatial stimuli to detect the separate neural networks encoding the different input).

Chapter 3

Theory

In this chapter, we describe the theoretical background for the methods used later in this thesis. Section 3.1 introduces the biological theory on neurons (head direction cells, place cells and grid cells) and the firing-rate model, before the neural network models are described in section 3.2. Persistent homology (including complex construction) is described in section 3.3, and, in addition, a few examples of applications on known topological spaces are given.

3.1 Neurons and the representation of space

Neurons are often classified by how they respond during a behavioural task. This is true for grid, place, head direction, boundary vector, speed and conjunctive (grid and head) cells found in the hippocampal and/or entorhinal system (see figure 3.2), which are thought to form the basis for the representation of physical space and the animal's location within it (Sanders et al. (2015)). Although it is still unknown how these cells work together in an integrated way, it is theorized that place cells compute position, grid cells distances and boundary cells indicate the edges of the environment, these forming the basis for a 'map', while the 'compass' information is provided by head direction cells. These are modulated by the speed derived from speed cells, and thus might enable navigation.

We will first look at the simple dynamics of a neuron before we discuss the properties of the types of neurons studied in this thesis.

3.1.1 Firing-rate model

Neurons are nerve cells and constitute for a large part of the brain, thought to collectively give animals the ability to think. They function through electrical signalling, sending short electrical pulses known as action potentials or *spikes*. This dynamic can be described by an integration process and a threshold mechanism effecting the 'firing' of a spike. This may be modelled by "integrate-and-fire" models (Gerstner et al. (2014)).

Figure 3.1: Navigation requires knowledge of direction and position. The neural substrate for this is thought to be encoded by place, grid and boundary vector cells and head direction cells respectively (credit: *pickywallpapers*).

Letting u_i be the voltage difference across the cell membrane of neuron i, the simplest model, known as the ”leaky integrate-and-fire” model, is similar to that of an RC-circuit (see figure 3.3) and takes the form of

$$\tau \frac{du_i(t)}{dt} = -(u_i(t) - u_{\text{rest}}) + RI_i(t), \tag{3.1}$$

where $\tau = RC$ is the membrane time constant, C is the capacitance maintained by the difference of charged ions on both sides of the cell membrane, R the resistance and $I_i(t)$ the time-dependent current from within the brain (e.g. synaptic input from other neurons connected to neuron i). After enough positive input current, the voltage passes a threshold (produces a spike) and resets to its resting potential u_{rest}.

With a few simple assumptions (e.g. constant current) it is possible to derive the following firing-rate model from equation (3.1) (the derivation omitted here, see Dunn (2016)):

$$\tau \frac{dS_i(t)}{dt} + S_i(t) = g \left[\sum_j W_{ij} S_j(t - \tau_{\text{delay}}) + I_{\text{ext}} \right]_+ . \tag{3.2}$$

Here, $[\ldots]_+$ is a threshold linear function enforcing non-negativity of the firing rates by setting the input to zero (if negative). $S_i(t)$ is the firing rate of neuron i, W_{ij} is the strength of influence of neuron j to neuron i and thus $W_{ij}S_j(t)$ is the increase in membrane potential of i due to the activity from cell j at time t. I_{ext} represents the (external) input, τ a time constant, τ_{delay} the synaptic time delay and g the gain controlling the input value.

3.1.2 Navigation and spatially and directionally tuned neurons

By what means do mammals navigate? Even in the absence of external sensory cues, rodents maintain an estimate of their position, allowing them to return home in a straight line without ever having taken this path before (as illustrated in figure 3.4b). This ability is called *path integration* and suggests an accurate internal representation of their local environment. This requires updating of position and head direction by integration of internally available self-motion information such as proprioceptive and vestibular input - what is known as *idiothetic* cues. A second navigational strategy is *landmark navigation* (figure

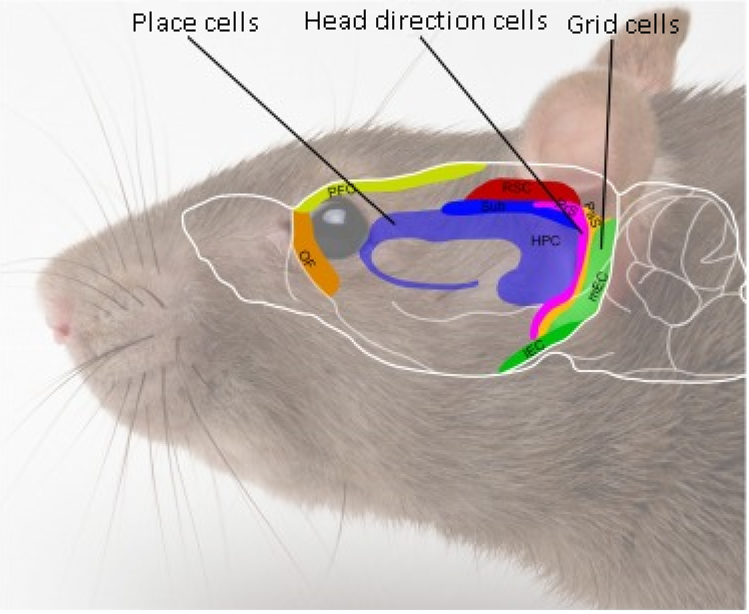
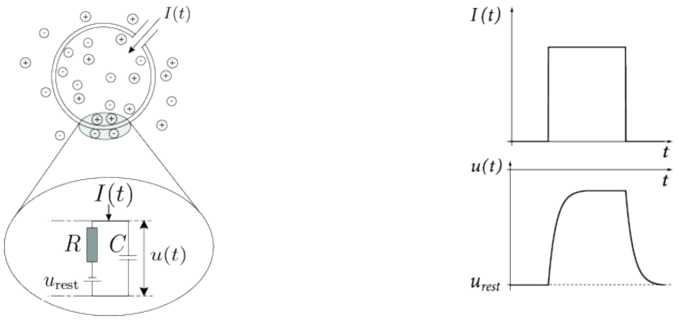


Figure 3.2: Illustration of the rat brain (white outline) with the brain regions involved in encoding spatial representation highlighted. The black lines point to where the cells were discovered, but they have been found in multiple regions. The brain regions are denoted by abbreviations, these are: HPC = hippocampus (pale blue); Sub = subiculum (dark blue), RSC = retrosplenial cortex (red); PrS = presubiculum (pink); PaS = parasubiculum (dark yellow); mEC = medial entorhinal cortex (light green); lEC = lateral entorhinal cortex (dark green); PFC = prefrontal cortex (pale yellow); OFC = orbitofrontal cortex (orange) (credit: Grieves and Jeffery (2017)).



(a) The neuron is enclosed by a cell membrane (big circle) and receives an input current $I(t)$ increasing the electrical charge inside the cell. The cell membrane acts like a capacitor in parallel with a resistor which is in line with a battery of potential u_{rest}
.

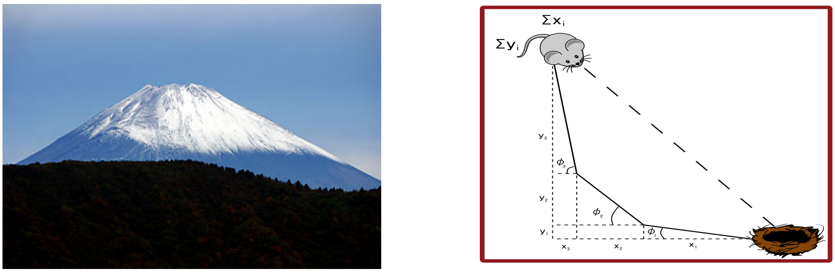
(b) The cell membrane reacts to a step current (top) with a smooth voltage trace (bottom).

Figure 3.3: An illustration of the 'leaky integrate-and-fire' model as an RC-circuit. (credit: Gerstner et al. (2014))

3.4a) - estimating the current position and orientation in the environment relative to surrounding landmarks. This sensory information is usually obtained via visual, auditory or olfactory input, referred to as *allothetic* cues (Taube (2009)).

Both strategies of animal navigation are hypothesized to be conjointly used in the formation of the cognitive map, thought to be encoded by the place cells. This was first proposed by O'Keefe (1976), who professed that place cells relied on two inputs: environmental stimuli and an inner navigational system calculating location through integration of linear and angular self-motion. He argued that when the animal located itself in the environment due to allothetic sources, the internal system would subsequently calculate the position based on how far and in what direction it has moved based on idiothetic sources. Thus, the navigational system would only allow the external stimuli to excite a cell when the animal is in a certain place, i.e., as Poucet et al. (2015) assert, that a landmark-based mechanism resets the animal's position in the place cell representation to compensate for the self-motion navigation not being accurate by itself.

Due to most cortical inputs to the hippocampus stemming from the entorhinal cortex, the information about external cues and self-motion is thought to be conveyed by entorhinal neurons (Poucet and Sargolini (2013)). The discoveries of head direction cells, grid cells and border cells in the mEC has further supported the idea of a brain system working as a landmark-independent navigational system upstream of the hippocampal place cells. However, other spatial and non-spatial cells in the mEC have been found and Zhang et al. (2013) show that the place fields may be generated by a convergence of signals from multiple entorhinal functional cell types. However, we will in the following describe the properties of the three types of cells believed to be crucial in encoding a representation of space.



(a) Landmark navigation is based on estimating position and orientation based on surrounding landmarks, here represented by Mount Fuji (Credit: Frank Gualtieri).

(b) The process of path integration. A mouse moves from its nest roaming around, and can return to its nest in an optimal route as it additively updates its position and head direction.

Figure 3.4: Navigational strategies.

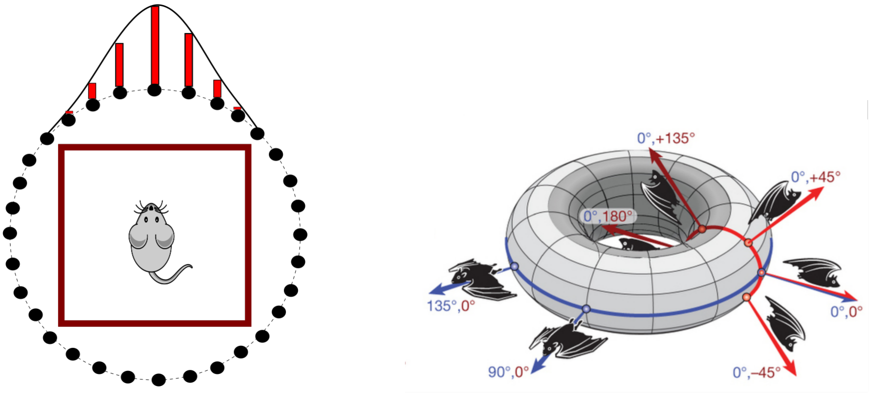
Head direction cells

Head direction cells were originally found in the rat dorsal presubiculum, having peak firing rates at specific head directions in the horizontal plane, and have since been found in several other brain regions largely associated with the hippocampal formation and have also been found in other mammals, including monkeys and humans. (Taube (2009))

The directions within a population of head direction cells are distributed uniformly on the 360°-interval, each cell having one direction in which it fires maximally. The relative directions are maintained under manipulation: when causing one cell's preferred direction to shift, the others' shift reciprocally in a continuous manner. Thus, the head direction system may be viewed as abstractly arranged on a circle, each cell placed at its preferred direction, and the activity represented as a single activity packet with a maximal firing rate in the direction which the animal's head is pointing. This is illustrated in figure 3.5a.

The head direction cell activity is independent of the geomagnetic field and unaffected by the animal's local location and ongoing behaviour. Moreover, the directional tuning is also maintained for some time when the lights are turned off, even if the animal is passively rotated in the dark. Thus, the direction may be maintained by idiothetic sources - for instance by vestibular signals, which have been found to be critically involved in the generation of the head direction signal. However, allothetic sources such as visual and odour cues are also important for setting the preferred directions. Skaggs et al. (1995) summarise that if the animal is not disoriented, the preferred direction of a head direction cell is well-defined by the relative orientation of the animal with respect to stable landmarks and visual and vestibular cues.

Contrary to rodents (in 2D boxes), many animals move in a highly three-dimensional space, but it is unclear whether a 3D compass exists in the brain and how it works. Finkelstein et al. (2015) found head-direction cells tuned to azimuth or pitch (and a few to roll) angles or to conjunctive combinations in both crawling and flying bats. In inverted bats the azimuth-tuning of they found the neurons to shift by 180°, and therefore suggested that 3D head directions is represented in azimuth×pitch toroidal coordinates (see figure 3.5b).



(a) A picture of a mouse inside a box, looking straight forward (upwards). The small circles on the outside of the box are nodes indicating head direction cells uniformly distributed on a larger circle (stippled line). The red columns signify the firing rate of the cells and the black curve is an interpolation of these making out the activity packet.

(b) The bat may have a 3D compass in which the head direction cells are tuned to azimuth × pitch toroidal coordinates. (credit: Finkelstein et al. (2015)/*Nature*)

Figure 3.5: Head direction cell network illustrations.

Place cells

Place cells were first observed by O'keefe & Dostrovsky in 1971, who found neurons in the rat hippocampus firing whenever the rat was in certain places in a local environment, the entire environment being represented by the activity of the local cell population. O'keefe and Nadel (1978) went on to propose that place cells form the basis of an internal map-like representation of space, modelling the animal motion. For rodents, we may view this as an abstract plane in which each place cell is represented by a node located at its preferred location. The neuronal activity behaves as a localized activity packet - a "bump" centred at the location of the chart representing the current location of the animal's head in the environment (see figure 3.6a). (Moser et al. (2008))

The place cell activity has been shown to display multiple properties. The speed and direction of the rat when moving, does not influence the shape or the behaviour of the packet, nor does the stability of immediate availability of sensory cues - for instance turning of the lights. The entire (cognitive) map may be remapped without distorting the planar representation, and this new configuration will persist equivalently to the one preceding. As the animal enters a novel environment, a new map immediately appears and stays consistent after exploration or changes of environmental stimuli. However, the relationship between the firing fields do not persist between common place cells, even when re-entering the same environment under different behavioural conditions (Samsonovich and McNaughton (1997)).

Yartsev and Ulanovsky (2013) studied how hippocampal place cells encode 3D space for freely flying bats. They found that place fields readily extend to a '3D map' - each place cell was active in single, regular 3D volumes, collectively filling the available space in the room.

Grid cells

Grid cells are spatially tuned neurons first found in the rat medial entorhinal cortex (mEC) (and since, in mice, macaques and humans) that behave much like place cells, but with multiple firing fields per cell. The fields are located at the vertices of a periodic hexagonal lattice, forming a triangular grid throughout the environment explored by the animal (see figure 3.6b). Grid firing is not affected by changes in the animal's speed and direction and retains its distinct structure under manipulations of the available sensory inputs, e.g. in the dark, across environment with varying topology and without regard to landmarks. (Dunn (2016))

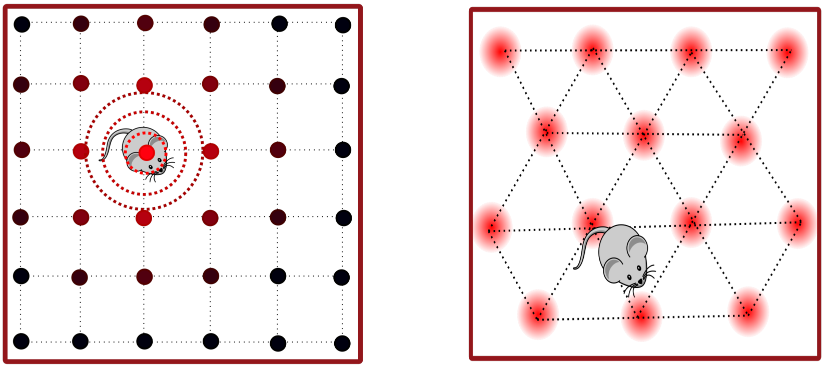
Although cells in the same part of the mEC have similar grid spacing and grid orientation, the phase of the grid is nontopographic. This means that the firing vertices of neighbouring grid cells appear to be shifted randomly, just like the fields of neighbouring place cells in the hippocampus. Cells in different parts of the mEC may also have different grid orientations (Hafting et al. (2005)), but the underlying topography (if there is one) has not been established. (Moser et al. (2008))

The form that grid cells exhibit in higher dimensionality is not clear, but grid cells should generate a common periodicity among neighbouring units while keeping them distinct in terms of spatial phase. Stella and Treves (2015) shows that, by extending a 2D self-organizing model of grid cell activity (Kropff and Treves (2008)), 3D grid cells may switch from firing largely at random to firing in some semblance of a three-dimensional pattern relatively quickly. However, they do not observe an equally clear ordered pattern as seen in two dimensions, but suggest that simpler firing patterns may form over shorter periods of time, providing a looser mapping of three-dimensional space.

3.2 Network model

McNaughton et al. (2006) argue that in path integration the information to be maintained and updated (position or head direction) is a continuous variable and that a continuum of cell assemblies is therefore needed to encode position or head direction. This is made possible by the *continuous attractor neural network* (CANN) model.

In general, an attractor network is a network of nodes whose time dynamics settle to a stable pattern or state called an *attractor* (Eliasmith (2007)). The attractors are determined by the internal connectivity between the nodes of the network and depending on the initial conditions, the network will end up in one of the stable states (McNaughton et al. (2006)). A continuous attractor network is then an attractor network possessing one or more quasi-continuous sets of attractors that in the limit of an infinite number of neuronal units N merge into continuous attractor(s). These will form a continuous manifold on which the system is neutrally stable, and the network state can translate easily when the external stimulus changes (Samansonovich (2017)). Thus, the CANN retains a continuum of stable solutions, and further reflects the neural networks by using recurrent collateral, *a priori*,



(a) Illustration of the planar representation of the place cell network of mouse. Redness signifies firing rate, and the dotted circles represent the activity bump, again the colour represents value.

(b) Illustration of the triangular structure of a single grid cell activity, illustrated as a mouse locomotes a 2D box. Redness signifies firing rate.

Figure 3.6: Place cell and grid cell illustrations.

local connections (local excitation and lateral inhibition) between the neurons. This asserts that a network functions as a coherent population with the connection strengths usually calculated as a function of the distance between the neurons in the state space.

Another model which we will also shortly describe is that of an *adaptation* (self-organization) model, which models 2D and 3D grid cells and explains the network activity by single cell behaviour.

3.2.1 Head direction cells

To model a head direction cell network, Skaggs et al. (1995) proposed using the principles of a continuous ring attractor network. The cells are figuratively allocated on a circle in agreement with their directional preferences and recurrent excitation is set onto head direction cells of similar preferred direction and inhibition of cells with different preferred firing directions. The continuous attractor properties now allow the formation of a collective activity distribution (the activity packet) from a random initial state (as seen in figure 3.7). The packet may be stationarily centred at any point on the circle, with the peak location representing the direction the animal is facing, and the packet is easily translated around the circle accordingly (see figure 3.8b).

The translation is a result of the external angular velocity calculated from vestibular input and optic flow, corresponding to the changes in the head direction of the rat. This drives the packet in either clockwise or anticlockwise direction as the velocity input is enforced through an intermediate ”double hidden layer” of additional neurons. These are placed on two separate circles, one receiving information about clockwise signal motion and projecting to the right of the outer circle, while the other receives information of anticlockwise motion and projects to the left (McNaughton et al. (2006)). In this way, the

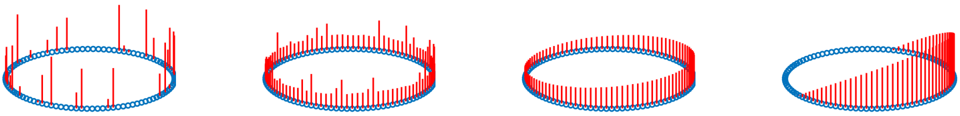
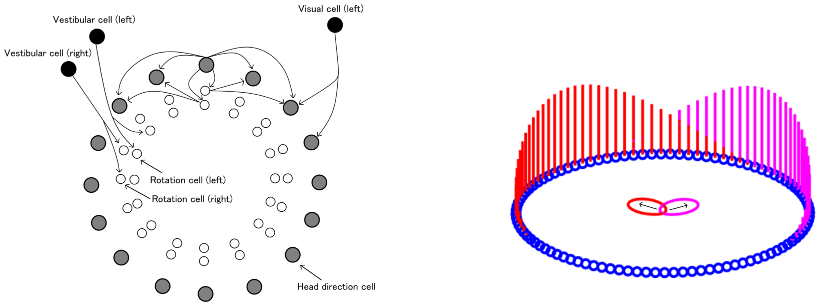


Figure 3.7: The evolution (from left to right) of a regular pattern for the one-dimensional head direction cell CANN, given a random initial state.

network performs angular velocity integration. The connections of the network are illustrated in figure 3.8a, including the connections providing information of (global) visual input and excitatory input from the head direction cell to the hidden layer(s) conveying the current network state.



(a) The architecture of a head direction population modelled as a continuous attractor network. The figure is a replicate of figure 3 in the paper by Skaggs et al. (1995). The rotation cells are the so-called hidden layers enabling rotation of the activity packet, furthering the vestibular input.

(b) The activity packet may be smoothly translated across the circle. The blue circles represent head direction cells, while the red and magenta bars show the activity packet at two different times, the peak firing rates corresponding to the two head directions (arrows), and the feedforward input shown as a function of the preferred head directions in the middle of the circle (red and magenta ovals).

Figure 3.8: Head direction cell CANN.

3.2.2 Place cells

A two-dimensional extension of the one-dimensional head direction cell network model above, may be constructed for the place cell network. In this model, the cells are arranged on a 2D sheet corresponding with the centres of their place fields and the connectivity is given as a function of the relative place field distance in physical space (inhibition of the cells far apart and excitation of those close).

As a consequence of the continuous attractor properties, the formation of a bump given a random initial state arises (see figure 3.9). Again, the resulting activity packet may be moved around the map. This is effected through an analogous intermediate 2D ”hidden layer”, summing the current position encoded in the continuous attractor layer and the speed and directional information given by a displacement vector. The attractor layer connects to the cells of similar positional preference through this hidden layer. The ”hidden” cells are positively modulated by running speed and encoded a given head direction, the return connection then projecting asymmetrically to the corresponding side of the cells in the attractor layer from which they receive input. Thus, when the animal moves, the relevant hidden cells provide an input that shifts the activity bump in the direction moved. (McNaughton et al. (2006))

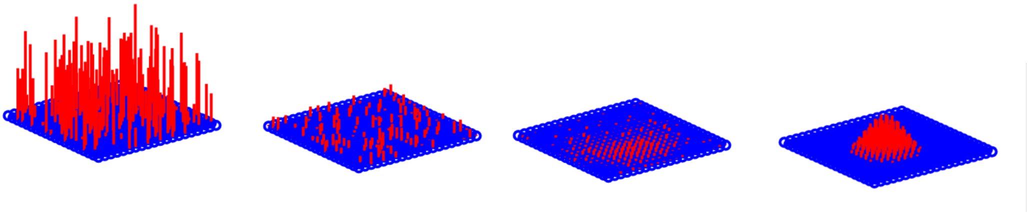


Figure 3.9: The evolution of a regular pattern (right) for the two-dimensional place cell CANN, given a random initial state (left).

3.2.3 Grid cells

Burak and Fiete (2009) show that continuous attractor models can accurately integrate velocity and head direction inputs to generate the regular triangular grid responses of grid cells, using a similar 2D sheet CANN as the place cell network. The connectivity can be modelled by global feedforward excitation and local, uniform inhibition. Due to the continuous attractor properties, this produces (from a random initial state) a regular pattern of discrete bumps of neural activity, arranged on the vertices of a regular triangular lattice (as seen in figure 3.10). The continuous attractor network then smoothly translates the pattern throughout the network with each point representing a stable attractor state.

The moving hexagonal pattern of the grid cell population activity is caused by integration of the rat’s velocity and head direction. Each neuron in any patch of the network is given a set of distributed directions and the connectivity is then given an asymmetry in this direction. A slight velocity tuning in the cell’s preferred direction then drives the movement of the network activity, the single cell periodic activity of the grid cells thus generated through the translation of the pattern by the speed-modulated directional selectivity. The cells preferring the current direction of motion will shift the network in that direction, while a neuron with the opposite directional preference will drive the flow in the opposite direction. If all neurons have equal input, the activity pattern will remain static, but as the rat moves in a specific direction in space, the cells with corresponding directional preference will receive larger input than the others, and will succeed in driving a flow of the network pattern along their preferred direction. (Dunn (2016))

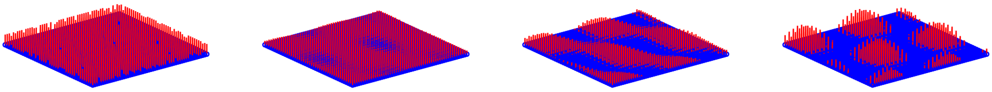


Figure 3.10: The evolution of a regular pattern for the two-dimensional grid cell CANN, given a random initial state.

Self-organizing grid cells

Kropff and Treves (2008) proposed a model in which the organization of grid fields into a triangular grid pattern is a result of single-cell process. The model suggests that during an exploration session, a (future) grid cell will, at first, respond at random locations based on random initial values for the feed-forward connections. These convey spatial input given by place cells. The spatial responses of the grid cells and the periodic grid pattern then gradually forms as the connection weights adapt through a (Hebbian) learning process. Stella and Treves (2015) succeed in generalizing this model to describe 3D grid cells. We will elaborate on both the two- and three-dimensional variant of this in the next chapter as we relate the model implementation.

3.3 Topological data analysis

Topology is the study of spaces equivalent under continuous deformations and has been found to provide desirable properties to data analysis. Edelsbrunner (2002) introduced the notion of persistent homology for a space filtration and gave an effective algorithm for computing this. Persistent homology is insensitive to the coordinate representation, robust to noise and is a tool for dimension-reduction and compressed representation of shapes. The following section gives a short introduction to complexes and persistent homology and is largely apprehended from my previous work (Hermansen (2017)), supported by the paper by Zomorodian and Carlsson (2005) and the survey by Botnan (2011). For a more thorough, yet easily available exposition, the book by Ghrist (2014) gives an excellent introduction to applied topology.

3.3.1 Complexes

We begin the discussion of topological data analysis by describing the notion of complexes - commonly-used objects in topology with both combinatorial and geometric descriptions. Complexes are used to approximate continuous spaces with more manageable constructions whilst inheriting topological properties of the original space. First introducing the basic simplicial complex, we go on to define a nerve, before describing the Čech, Vietoris-Rips, flag, clique and order complex.

Abstract simplicial complexes

Definition 3.3.1. A *k-simplex* is a k-polytope defined as the convex hull of its $k+1$ *affinely independent* vertices $u_0, \ldots, u_k \in \mathbb{R}^k$ (that is, $u_1 - u_0, \ldots, u_k - u_0$ are linearly independent (with an arbitrary choice of u_0)). The k-simplex may then be written explicitly as the set of points $C = \{\sum_i^k \lambda_i u_i | \sum_{i=0}^k \lambda_i = 1, \lambda_i \geq 0, 0 \leq i \leq k\}$.

Given a k-simplex, σ we call the convex hull of any non-empty subset of its $k+1$ vertices the *faces* of σ, and these define simplexes themselves. A k-simplex is of dimension k, and, for instance, a 0-simplex is a point, a 1-simplex an edge, a 2-simplex a triangle, a 3-simplex a tetrahedron and so on in higher dimensions.

Definition 3.3.2. A *simplicial complex κ* is a finite set of simplexes such that any face of a simplex in κ is also in κ, and the intersection of two simplexes is either empty or a face of each simplex.

The dimension of κ is defined to be the dimension of its highest-dimensional simplex. Note that the empty set is a face of all simplexes.

Definition 3.3.3. An *abstract simplicial complex* is a collection K of non-empty finite sets such that for every set $\tau \in K$ and every non-empty subset $\sigma \subseteq \tau$, σ also belongs to K.

Definition 3.3.4. If κ is a simplicial complex, and $V(\kappa)$ the vertex set of κ, then we call the collection of all subsets of $V(\kappa)$ that span a simplex the *vertex scheme* of κ.

Theorem 3.3.1. *Any abstract simplicial complex K is isomorphic to the vertex scheme of some simplicial complex κ.*

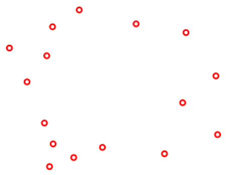
Given the induced topology, κ is a topological space and we call it the *geometric realization* of K, and hence this defines a functorial operation. This means that any simplicial complex is homeomorphic to the geometric realization of its associated abstract simplicial complex.

Nerve

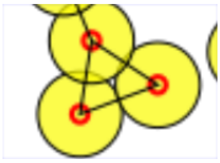
Given a topological space X, an *open cover* is a collection, $\mathbb{U}$, of open subsets of X, U_i, such that $X = \cup_{i \in I} U_i$, where I is some index set. If Y is a subset of X, then a cover of Y is a collection of subsets of X such that Y is contained in its union.

Definition 3.3.5. The nerve, $N(\mathbb{U})$, of the open cover $\mathbb{U} = \{U_i\}_{i \in I}$ is the set of finite subsets of I such that $J \subseteq I$ belongs to N if and only if $\bigcap\limits_{j \in J} U_j \neq \emptyset$.

We note that if J belongs to N then any of its subsets is automatically also in N, making N an abstract simplicial complex where the k-simplexes correspond to the intersection of $k+1$ distinct sets of the cover.



(a) A given point cloud in $\mathbb{R}^2$. Note the circular shape.



(b) Excerpt of lower left corner of the first square in figure 3.12, now computed for the Čech complex which no longer defines a 2-simplex. Note that this is homotopy equivalent to the union of balls.

We call $\mathbb{U}$ a *good cover* if all its sets and non-empty finite intersections are contractible. We may then state the following theorem which underpins the importance of the nerve construction as it gives a method of relating an abstract simplicial complex to a space which inherits many of the topological characteristics of the space.

Theorem 3.3.2. *If* $\mathbb{U} = \{U_\alpha\}$ *is a good cover of* X*, then* $N(\mathbb{U})$ *is* homotopic *to the union* $\bigcup\limits_{i \in I} U_i$.

The Čech and the Vietoris-Rips complex

Given a point set X in some metric space (M, d) and a real number $\epsilon > 0$, the *Čech complex*, $\check{C}_\epsilon(X)$, is the simplicial complex whose simplexes are formed as follows. For each subset σ of points in X, we form a closed ball of radius ϵ, $B(x, \epsilon) = \{x' \in M | d(x, x') \leq \epsilon\}$, around all $x \in \sigma$. If all of the balls contain a common point, then σ defines a simplex of dimension $|\sigma|$. Any subset $\tau \subseteq \sigma$ then contains the same common point, making this a natural complex construction. This defines a good cover and we note the following equivalent definition.

Definition 3.3.6. $\check{C}_\epsilon(X) = N(\bigcup\limits_{x \in X} B(x, \epsilon))$.

Čech complexes are expensive to compute as the dimension of the abstract simplicial complex may grow very large. Thus, one needs to check for a large number of intersections and it also requires storage of all its simplexes. To overcome the latter problem, we introduce the *Vietoris-Rips complex*.

Definition 3.3.7. Given a finite set of points X in some metric space (M, d), the Vietoris-Rips complex is given by: $VR_\epsilon = \{\sigma \subseteq X; d(x, y) \leq 2\epsilon, \forall x, y \in \sigma\}$.

This is a relaxation of the Čech complex condition as it only requires pairwise intersections, and is not necessarily homotopy equivalent to the union of balls around each point (compare figure 3.11b and 3.12). However, the Vietoris-Rips complex approximates the Čech complex due to the following.

Theorem 3.3.3. *Let* $\epsilon > 0$ *and* X *a subset of Euclidean space, then the following inclusions hold:* $VR_\epsilon(X) \subseteq \check{C}_{\epsilon\sqrt{2}}(X) \subseteq VR_{\epsilon\sqrt{2}}(X)$.

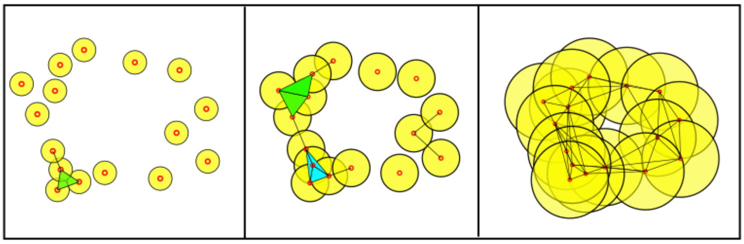


Figure 3.12: The Vietoris-Rips complex for the points in 3.11a for three (increasing) parameter values with the balls, around each of the points, in yellow. Green signifies triangles (2-simplexes) and light blue tetrahedrons (3-simplexes). Only the underlying graph and the balls are shown in the last box, and the complex is given by taking the clique complex of this graph.

This implies that any topological feature which persists under the inclusion $VR_\epsilon \hookrightarrow VR_{\epsilon'}$, where $\epsilon'/\epsilon \geq \sqrt{2}$, is also a topological feature of the Čech complex $\check{C}_{\epsilon'}$

A Vietoris-Rips complex may be preferred to the Čech complex, requiring less storage as it can be represented as by a graph, regenerated by including edges between two points if their corresponding balls overlap and computing simplexes for every clique in the graph. However, it has same worst-case complexity as the Čech complex with max $2^{|S|} - 1$ simplexes, max dimension $S - 1$, so common practice in applications is to compute simplexes up to a certain dimension k for some $k << S - 1$.

Clique and flag complex

A graph is a simplicial complex only consisting of simplexes of dimensions 0 (the vertices) and 1 (the edges). However, using a graph as the basic structure, we may construct higher dimensional complexes.

Definition 3.3.8. Given an undirected graph G, the *clique complex*, $K(G)$, is the abstract simplicial complex formed by the sets of vertices in the cliques of G. In other words, a k-simplex is in $K(G)$ if its $k + 1$ vertices in the graph forms a clique (see figure 3.13).

A similar concept, but not necessarily constructed from a graph, is the *flag complex*.

Definition 3.3.9. A flag complex is an abstract simplicial complex with the property that a finite set of vertices is a simplex if and only if each pair of those vertices is a 1-simplex. Equivalently, a flag complex is an abstract simplicial complex with no empty simplexes. Any flag complex is the clique complex of its 1-skeleton.

We shortly note that the flag complex is natural to use even when the data does not present an obvious metric, but is defined by some pointwise dissimilarity measure. Given this measure between each point in some set, we may draw a weighted edge between (all) points and define the flag complex based on this graph. We may then *filter* the complex by the correlation value, adding edges (and thus cliques and simplexes) correspondingly as we increase this value. This is what is done when we construct the *order complex* as described by Spreemann et al. (2015):

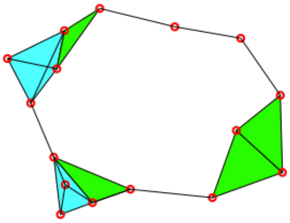


Figure 3.13: Clique complex from a given graph. Light blue signifies tetrahedrons (3-simplexes) and green triangles (2-simplexes).

Definition 3.3.10. Let $G = (E, V)$ be the complete graph on the vertices $V = 1, \ldots, N$ with edge weights W. Let $\phi : \{1, \ldots, \binom{N}{2}\} \to E$ be a bijection that sorts the weights

$$W(\phi(1)) < \ldots < W(\phi(\tbinom{N}{2})),$$

and let $\tilde{G}$ be the complete graph on V with the edge weights

$$\tilde{W}(i, j) = \frac{\phi^{-1}(i, j)}{\binom{N}{2}}.$$

The order complex of G is the flag complex $OC(G) = fl(\tilde{G})$.

3.3.2 Persistent homology

To distinguish and describe the topological properties of a space, one looks for topological invariants for the space, and finding the *homology* of a space is one of the most commonly used methods for obtaining such an invariant. Homology is a functorial, homotopy-invariant algebraic structure motivated by the observation that two shapes can be distinguished by examining the number of n-dimensional holes of each. The homology groups, $H_n(X)$, associate a sequence of abelian groups (or modules) to a space, and the rank of the homology groups are called the *Betti numbers*, β_n. These are the discrete invariants that intuitively gives a count of the number of p-dimensional holes in K: β_0 gives the number of path-connected components, β_1 the number of ”regular” holes, β_2 the number of voids and so on.

One may find the homology of a complex constructed at a given parameter value, but this is easily sensitive to noise which might produce homology cycles not representative of the underlying space, as one does not have *a priori* knowledge of what parameter value to choose. For instance, the homology of the two leftmost complexes in figure 3.12, will not capture the circular feature of the distribution of points. Persistent homology overcomes this obstacle by calculating the homology of a parametrised complex over *all* possible parameter values.

Given an associated nested sequence of complexes K^ϵ, persistent homology gives a way to study how the changes of the parameter value ϵ affects the homology of the sequence, thus capturing the topology of the filtration and studying which features *persist* over the filtration. If a geometric feature at $H_n(K^{\epsilon_0})$ is also present at $H_n(K^{\epsilon_1})$ for

$\epsilon_0 \leq \epsilon_1$, the feature is said to persist from ϵ_0 to ϵ_1. Features which persist through many parameter choices are considered more significant, while features which quickly disappear are considered as noise. This serves as the motivation for persistent homology, and we will in this section further elaborate on the theoretical background for this method.

Filtration

Definition 3.3.11. A *filtration* $\mathbb{F}$ is an indexed set S^i of subobjects of a given algebraic structure S with the index i running over a totally ordered set I subject to the condition that if $i \leq j$ in I then $S^i \subseteq S^j$.

Given a finite simplicial complex K we obtain a filtration by associating a finite sequence of nested subcomplexes of K: $\emptyset = K^0 \subseteq K^1 \subseteq \cdots \subseteq K^l = K$ for some l. We thus call K a *filtered complex*. To construct this, we build simplicial complexes using the methods described in section 3.3.1, for increasing values of the ϵ-parameter.

Simplicial Homology

Generally, computing the homology of a topological space may be difficult, but approximating the space by simplicial complexes and calculating the homology of these, reduces the computation to linear algebra. In the following, we define the homology of the geometric realization of a given abstract simplicial complex.

Definition 3.3.12. Given a simplicial complex K, a *simplicial p-chain* is $\sum_{i=1}^{N} c_i\sigma_i$, where c_i is an integer coefficient and σ_i is an oriented p-simplex in K. Each oriented simplex is declared equal to the negative of the simplex with the opposite direction.

We define $C_p(K)$ as the free abelian group of p-chains, with basis the p-simplexes. The *boundary operator* $\delta_p : C_p(K) \to C_{p-1}(K)$ is then defined as the homomorphism $\delta_p(\sigma) = \sum_{i=0}^{k}(-1)^i(v_0, \dots, \widehat{v_i}, \dots, v_k)$. It is easy to check that $\delta_p\delta_{p+1} = 0, \forall p \in \mathbb{Z}$. This connects the groups $C_*(K)$, forming a *chain complex*:

$$\begin{tikzcd} \dots \arrow[r] & C_{p+1}(K) \arrow[r, "\delta_{p+1}"] & C_p(K) \arrow[r, "\delta_p"] & C_{p-1}(K) \arrow[r] & \dots \end{tikzcd}$$

The subgroups of C_p: $Z_p = \text{Ker}(\delta_p)$ and $B_p = \text{Im}(\delta_{p+1})$ are, respectively, called the *cycle* and *boundary* groups. The boundary operator is then a linear map taking a p-simplex to its *boundary* (the sum of its faces of codimension 1). Since $\delta_p\delta_{p+1} = 0$, the boundary of a boundary is empty, and hence it follows that $B_p \subseteq Z_p \subseteq C_p$. We may now define the simplicial homology groups.

Definition 3.3.13. The homology groups of K, is defined as the quotient groups $H_p(K) = Z_p/B_p, \forall p \in \mathbb{Z}$, whose elements are classes of *homologous* cycles.

The universal coefficient theorem enables us to define homology in other coefficients than the integers. Denoting $H_p(K, G)$ as the p-th homology group of a simplicial complex K with coefficients in an abelian group G, the theorem gives the natural short exact sequence

$$\begin{tikzcd} 0 \arrow[r] & H_p(K, \mathbb{Z}) \otimes G \arrow[r] & H_p(K, G) \arrow[r] & \text{Tor}(H_{p-1}(K, \mathbb{Z}), G) \arrow[r] & 0, \end{tikzcd}$$

where $\otimes$ is the tensor product and Tor is the derived functor of the tensor product. This sequence splits, though not naturally. Thus we may write $H_p(K, G) \cong (H_p(K) \otimes G) \oplus (\mathrm{Tor}(H_{p-1}(K), G))$. By the fundamental theorem for finitely generated abelian groups, we have that

$$H_p(K) \cong \mathbb{Z}^n \oplus \mathbb{Z}_{k_1^{\alpha_1}} \oplus \ldots \oplus \mathbb{Z}_{k_m^{\alpha_m}}$$

where n is the rank of the group, k_i is a prime and α_i a positive integer. We then see that for $G = \mathbb{Z}_q$,

$$H_p(K, \mathbb{Z}_q) \cong (\mathbb{Z}_q^n \oplus \mathbb{Z}_{c_1} \oplus \ldots \oplus \mathbb{Z}_{c_m}) \bigoplus (\mathbb{Z}_{c_1'} \oplus \ldots \oplus \mathbb{Z}_{c_m'})$$

where $c_i = \gcd(p_i, k)$ and $c_i' = \gcd(p_i', k)$, where p_i' are the primes in H_{p-1}.

When $H_p(K)$ is torsion-free, i.e. of the form $\mathbb{Z}^n$, we call the rank of the homology groups the *Betti numbers*, $\beta_p = \mathrm{rank}(H_p) = \mathrm{rank}(\mathrm{Ker}(\delta_p)) - \mathrm{rank}(\mathrm{Im}(\delta_{p+1}))$. These are always integers regardless of the coefficient ring, but do not account for the torsion of the homology groups. The Betti number for homology groups with coefficients over a field will only differ when the homology contains some q-torsion and the field is of characteristic q.

In computation, we will only work with coefficients in the field $\mathbb{Z}_2$, for which we do not need to worry about orientation or storing coefficients.

Simplicial Persistent Homology

Given a filtration of the simplicial complex K and a sequence of homology groups,

$$0 \longrightarrow H_p(K^0) \xrightarrow{i_*^0} H_p(K^1) \xrightarrow{i_*^1} \ldots \xrightarrow{i_*^{n-1}} H_p(K^{N-1}) \xrightarrow{i_*^{n-1}} H_p(K^N) \longrightarrow 0,$$

where the homomorphisms $\{i_*^j\}$ are the induced inclusions (compositions and identities have been omitted from the diagram). We may then define the persistent homology groups.

Definition 3.3.14. For $i < j$, the (i, j)-persistent p-th homology group of K, $H_p^{i \to j}$, is defined as the image of the induced homomorphism $\phi_p : H_p(K^i) \to H_p(K^j)$, and the p^{th} persistent Betti number $\beta_p^{i,j}$ is the rank of this group.

Persistence module

We go on to define the *persistence module* which will enable us to uniquely represent the persistent homology of a simplicial complex with a *barcode* or a *persistence diagram*. First, we note that the filtered complex K with inclusion maps for the simplexes becomes a *persistence complex*.

Definition 3.3.15. A persistence complex $\mathbf{C}$ is a family of chain complexes $\{C_*^i\}$ over a ring R together with chain maps $f^i : \mathbf{C}_*^i \to \mathbf{C}_*^{i+1}$.

Definition 3.3.16. A persistence module, $\mathscr{M}$, is a set of R-modules M_i with linear maps $\phi_t^s : M_s \to M_t$ whenever $s \leq t$, with $\phi_t^t = Id$ and $\phi_t^s \circ \phi_s^r = \phi_t^r$ for $r \leq s \leq t$.

We call a persistence module of *finite type* if each component module is a finitely generated R-module, and if the maps $\phi_t^s, \forall s \in P$ are isomorphisms for $t \geq v$ for some integer $v \in P$.

The homology of the persistence complex $\mathbf{C}(K)$ of finite type generates a persistence module of finite type.

Correspondence

Given a persistence module $\mathscr{M} = \{M^i, \phi^i\}_{i \geq 0}$ over R, we may define a correspondence with a graded module over $R[x]$, defined by $\alpha(\mathscr{M}) = \bigoplus_{i=0}^{\infty} M_i$, where the R-module structure is simply the sum of the structures on the individual components, and where the action of x is given by $x \cdot (m^0, m^1, \ldots) = (0, \phi^0(m^0), \phi^1(m^1), \ldots)$, i.e. multiplication by x corresponds to shifting one step up in the grading, $x : M_i \to M_{i+1}$.

Theorem 3.3.4. *The category of persistence modules of finite type over R is equivalent to the category of finitely generated non-negatively graded modules over $R[x]$.*

Intuitively, we are building a single structure that contains all the complexes in the filtration. We begin by computing a direct sum of the complexes, arriving at a much larger space that is graded by the filtration ordering. Then we remember the time each simplex enters using a polynomial coefficient. For instance, if a simplex σ enters the filtration at time 0, then we must multiply the simplex using x to shift it along the grading. Then σ exists at time 0, $x \cdot \sigma$ at time 1, $x^2 \cdot \sigma$ at time 2, etc. The key idea is that the filtration ordering is encoded in the coefficient polynomial ring.

Decomposition

Theorem 3.3.4 gives rise to the belief that a simple classification of $\mathbb{Z}[x]$-modules does not exist as the classification of modules over $\mathbb{Z}[x]$ is extremely complicated. However, it gives a simple decomposition if we take R to be a field $\Bbbk$. The graded ring $\Bbbk[x]$ is a PID and its graded ideals are homogenous of the form $(x^n) = x^n \cdot \Bbbk[x], n \geq 0$. By the standard structure theorem, we get a classification of the $\Bbbk$-modules,

$$\mathscr{M} \cong \bigoplus_i x^{t_i} \cdot \Bbbk[x] \oplus (\bigoplus_j x^{r_j} \cdot (\Bbbk[x]/(x^{s_j} \cdot \Bbbk[x])))$$

.

Intuitively, for the persistence module in study, the free part on the right side corresponds to the homology generators that appear at filtration level t_i and never disappear, while the torsion part corresponds to those that appear at filtration level r_j and disappear at filtration level $s_j + r_j$.

Persistence Intervals, Diagrams and Barcodes

We may now easily grasp the persistence module structure by introducing the notion of *persistence intervals* and representing these using persistence diagrams, which visualize the intervals (and thus the modules), enabling an easily accessible visual representation of the topological structure of the space studied.

Definition 3.3.17. A persistence interval is an ordered pair (i, j) with $0 \leq i < j \in \mathbb{Z} \cup \{+\infty\}$.

We define a bijection Q which associates a graded $\Bbbk[x]$-module to a set of persistence intervals. Let $Q(i, j) = x^i \cdot \Bbbk[x]/x^{j-i}$ and $Q(i, +\infty) = x^i \cdot \Bbbk[x]$. Then for a set $(i_1, j_1), (i_2, j_2), \ldots, (i_n, j_n)$, we define a finitely graded $\Bbbk[x]$-module M_Q by $\bigoplus_{i=1}^{n} Q(i_l, j_l)$, and the following corollary is immediate.

Corollary 3.3.4.1. *The finite sets of persistence intervals are in bijective correspondence with the finitely generated graded modules over* $\Bbbk[x]$.

We may then represent the persistence intervals (and thus the persistent homology classes) in the form of a persistence diagram, defined as the set of points $(r_j, r_j + s_j)$ related to each interval. r_j is called the birth and $r_j + s_j$ the death of the corresponding class. In other words, the persistence diagram is the union of a finite multiset of points in $\overline{\mathbb{R}}^2$. We also include the multiset of points on the diagonal ($\Delta = \{(x, y)|x = y\}$), in which all points are defined to have infinite multiplicity.

3.3.3 Examples - sampled spaces

In this section, we illustrate the use of persistent homology and the resulting persistence diagrams applied on point clouds sampled from known topological spaces. That is, we draw a random sample from a parameterization of familiar topological spaces - the sphere, the torus and the Klein bottle, and calculate persistent homology of the filtered Vietoris-Rips complexes of these point clouds. These are computed using the Ripser software.[1]

Sphere

We first test for a random sample of 500 points on the 2-sphere, $\mathbb{S}^2$, (see figure 3.14a). The homology groups of an n-sphere is $H_i(\mathbb{S}^n, \mathbb{Z}_2) \cong \mathbb{Z}_2$ for $i = 0$ and n, and 0 otherwise. Thus the theoretically calculated Betti numbers for the 2-sphere are $\beta_0 = 1, \beta_1 = 0, \beta_2 = 1$ and $\beta_i = 0, i > 2$, complying with the persistence diagram in figure 3.14d.

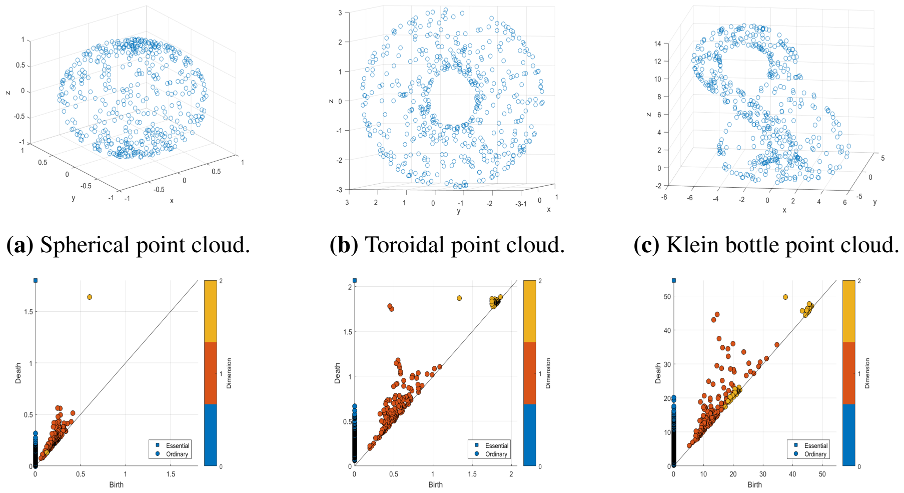
Torus

500 points were then randomly sampled from the 2-torus as shown in figure 3.14b. The nontrivial homology groups of the torus, $T \cong \mathbb{S}^1 \times \mathbb{S}^1$, are $H_0(K, \mathbb{Z}_2) \cong \mathbb{Z}_2$, $H_1(K, \mathbb{Z}_2) \cong \mathbb{Z}_2 \oplus \mathbb{Z}_2$ and $H_2(K, \mathbb{Z}_2) \cong \mathbb{Z}_2$, and hence $\beta_0 = 1, \beta_1 = 2, \beta_2 = 1$ and $\beta_i = 0, i > 2$. The persistence diagram in figure 3.14e show an accordance with this.

Klein Bottle

The third example consists of 500 points randomly sampled from the klein bottle, as shown in figure 3.14c. The nontrivial homology groups of the Klein bottle K are similar to that of the torus when computing with $\mathbb{Z}_2$-coefficients: $H_0(K, \mathbb{Z}_2) \cong \mathbb{Z}_2$, $H_1(K, \mathbb{Z}_2) \cong \mathbb{Z}_2 \oplus \mathbb{Z}_2$

[1] `https://github.com/Ripser/ripser`



(a) Spherical point cloud.

(b) Toroidal point cloud.

(c) Klein bottle point cloud.

(d) Persistence diagram for the point cloud in figure 3.14a. We clearly see two distinct homology classes, one 0-dimensional, and one 2-dimensional.

(e) Persistence diagram for the the point cloud in figure 3.14b. Four significant persistent homology classes are seen, one 0-dimensional, two 1-dimensional and one 2-dimensional.

(f) Persistence diagram for the point cloud in 3.14c. Four significant persistent homology classes are seen, one 0-dimensional, two 1-dimensional and one 2-dimensional.

Figure 3.14: In (a)-(c), the point clouds for the three sampled space are shown. Each cloud consists of 500 points randomly sampled from a larger uniform sample computed by parametrisation. (d)-(e) show the persistence diagram resulting from applying persistent homology to the sampled spaces. The x-axes signify the time of birth for the persistent homology classes and the y-axes the time of death. The colouring corresponds to the dimension of the homology classes - blue represents the 0-dimensional classes, red 1-dimensional classes and yellow the 2-dimensional classes.

and $H_0(K, \mathbb{Z}_2) \cong \mathbb{Z}_2$, i.e. $\beta_0 = 1, \beta_1 = 2, \beta_2 = 1$ and $\beta_i = 0, i > 1$. The persistence diagram show long-lived persistent homology classes correspondingly, see figure 3.14f. However, the significant persistence classes are not as clear as in the two other cases, perhaps due to the oversampling of the bottle neck as may be seen in figure 3.14c.

Chapter 4

Method

In this chapter we depict the methods used. We first discuss the implementation of the neural network models and the neural data management in section 4.1, before relating the main scheme in detecting the topology in this data in section 4.2.

4.1 Neural model implementation

The dynamics of the neural networks are all based on the firing rate model of equation (3.2). Thus, we compute the activity level (firing rate), S_i, of neuron i using the difference scheme:

$$S_i^{t+1} = -S_i^t + \frac{\Delta t}{\tau}\left(S_i^t + g\left[\sum_j W_{ij}S_j^t + I_{\text{ext}} + \xi_i^t\right]_+\right), \tag{4.1}$$

where the time delay is disregarded, $[\ldots]_+$ is once again the threshold-linear function (enforcing non-negativity), $g = 1$ the gain (keeping the response of the nervous system within certain bounds), $I_{\text{ext}} = 1$ the constant external input, $\tau = 10$ms the time constant, ξ the feed-forward input and W_{ij} the strength of the connection between neuron i and j.

4.1.1 Head direction cell network

To simulate $N = 100$ head direction cells, we first assign an angle, θ, to each neuron i: $\theta_i = \frac{2\pi i}{N} + \epsilon_i$, where ϵ_i is a small, normally distributed noise variable (~ 0.02). For simplicity, we do not explicitly model inhibitory interneurons but account for their influence in the effective connectivity between neurons. The connectivity pattern follows that of an all-or-none inhibitory connectivity, with the connection strengths defined by parameters W_0 and R as follows:

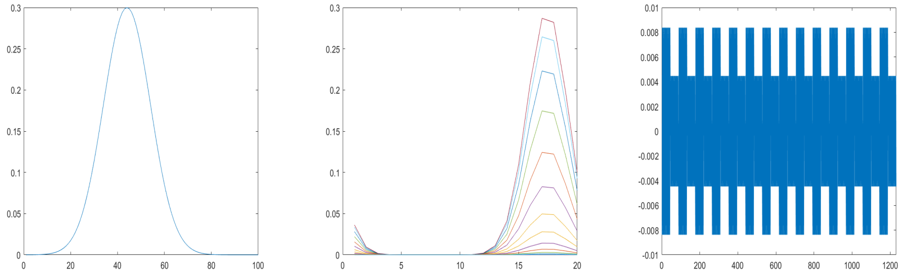
$$W_{ij} = W_0 H\left(\max(|x_i - x_j|, N - |x_i - x_j|) - R\right). \tag{4.2}$$

Here, R is the radial extent of the connectivity, and $W_0 = -0.05$ represents the strength of the inhibitory interaction between connected neurons. $x_i = 1, \ldots, N$ and $y_i = 1, \ldots, N$ represent the position of neuron i in a circle. $H(\ldots)$ is the Heaviside function (producing a 0 or a 1 depending on whether its argument is greater than zero), giving a bimodal distribution: the strength of the inhibitory interaction equals 0 or W_0 depending on whether the relative abstract distance between units i and j is larger than $R = 25$. We disregard excitatory functional connectivity. In fact, anatomical studies show no evidence of recurrent excitatory collaterals, but motivate exclusively inhibitory connections (Boucheny et al. (2005)). However, all neurons receive a constant excitatory input ($I_{\text{ext}} = 1$) representing possible idiothetic and/or allothetic sources.

For head direction networks the feed-forward input is given by:

$$\xi_i^t = \alpha v \text{exp}\left(\frac{(\theta_t - \theta_i)^2}{2\sigma^2}\right), \tag{4.3}$$

where $\alpha = 0.3$ is the velocity modulation, v the speed, θ_t the head direction at time t, θ_i the preferred direction of neuron i and $\sigma = 0.4$ the standard deviation. The input is plotted for a random time in figure 4.1a.



(a) Head direction cell network input.

(b) Place cell network input.

(c) Grid cell network input.

Figure 4.1: The feedforward input of the different networks (spatial and directional noise set to 0 for the cell's preferred positions and directions).

The activity pattern is initialized by randomly setting $\frac{1}{3}$ of the neuron population at a random firing rate between 0 and 1 and the rest silent ($S_i^0 = 0$), and a regular pattern is stabilized by numerically integrating equation (4.1) with $\Delta t = 1$ms over 200ms and the feed-forward input set to 0. This was illustrated in figure 3.7.

To perform the simulation, we integrate equation (4.1) with a 1ms time step. We use both a 40min interval as the rat moves in a square box of size 1m×1m from recorded rat positions and generate a 17min random walk in an equally sized box. The random walk is performed with a constant estimate of speed and at each step generating a random direction from a normal distribution with the previous direction as mean and a small variance. The random walk is then smoothed using the *cscvn* MATLAB-function, creating a periodic interpolating cubic spline curve through the random walk positions generated. We then evaluate the obtained piecewise polynomial at linearly spaced query points (time steps) and thus get xy-coordinate positions which we assume to be the location of the head of an animal during locomotion. The temporal head directions are then calculated by linearly interpolating the direction vector at 10ms time bins while the speed is assumed constant ($v = 1$) throughout the simulations.

We also simulate a head direction cell network in which each cell has two separate preferred head directions and the network thus receives an added feed-forward input calculated separately using two preferred angles respectively, and inhibitory connections are enforced based on the difference in both preferred head directions, between all cells. In addition, we simulate head direction cells for variously computed angles as explained in the next chapter.

3D head direction cell network

To simulate a 3D head direction cell network, we first need to simulate a 3D animal movement. We do this by generating a 3D random walk ('flight') restricted to a $1\text{m}\times1\text{m}\times1\text{m}$ cube, extending the random walk of the 2D simulation in an obvious manner. We wish to find the azimuth (ϕ), pitch (θ) and roll (γ) angles of the animal in the created movement and follow the method of Finkelstein et al. (2015) to generate azimuth and pitch angle (described in the following), and find the roll angle in a random fashion (as we only produce a one-dimensional path, which does not allow us to recreate a "natural" roll angle).

Finkelstein et al. propose (empirically) that a bat's 3D head direction network is represented in toroidal coordinates and relate the two axes of the torus to azimuth and pitch angles. They define the azimuth angle as the angle of the inter-aural (ear-to-ear) axis projection onto the xy-plane, which, in toroidal coordinates, results in the angle being insensitive to rotations in pitch. The pitch is defined as the angle of the rostro-caudal axis of the head in the plane, i.e. defined by the z-axis and the projection of the rostro-caudal axis onto the xy-plane. In the toroidal coordinates the pitch becomes continuous with a cyclical range of $\pm\pi$. Thus, when the animal is (assumed) inverted, the angles are recalculated as $\phi \rightarrow \phi + 180°$ and $\theta \rightarrow 180° - \theta$. The angles are calculated as follows:

$$\phi(t) = \text{atan2}(dy(t), dx(t)), \tag{4.4}$$

$$\theta(t) = \text{atan2}(dz(t), \sqrt{dx(t)^2 + dy(t)^2}), \tag{4.5}$$

$$\gamma(t) = \gamma(t) + \frac{\pi}{8}\zeta. \tag{4.6}$$

atan2 is the four-quadrant inverse tangent and, given the xyz-coordinates of the flight trajectory we calculate the differentials $dx(t) = x(t + dt) - x(t - dt), dy(t) = y(t + dt) - y(t - dt), dz(t) = z(t + dt) - z(t - dt)$, with $dt = 10\text{ms}$. To allow the pitch angle to assume all values, we assume the animal is inverted (flying upside-down) when $|(\theta(t))| > \pi/2 - 2(\theta(t-1) - \theta(t-2)$ and $||\phi(t+1)| - |\phi(t)|| > \pi/4$, and assuming again flying the right way up by checking similar conditions. ζ is a random number drawn from the normal distribution with mean 0 and standard deviation 0.1. All angles are initialized at a random angle and the roll angle γ is also kept within $[-\pi, \pi]$.

4.1.2 Place Cell Network

We simulate $N = 20 \times 20$ place cells arranged on a two-dimensional square neuronal sheet. Though periodic place fields never have been reported (McNaughton et al. (2006)), Samsonovich and McNaughton (1997) propose a place cell CANN based on a periodic sheet, and we simulate both an aperiodic and a periodic neuronal sheet.

The neuronal dynamics are again given by equation (4.1), and the connection strengths are defined as follows:

$$W_{ij} = W_0 H\left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} - R\right), \tag{4.7}$$

with $R = 5$ for the aperiodic network and 2.5 for the periodic, and $W_0 = -0.05$ and the Heaviside function once again ensuring non-existent local inhibition.

The feed-forward input is now defined by:

$$\xi_i^t = \alpha v^t \text{exp}\left(\frac{(x^t - x_i)^2 + (y^t - y_i)^2}{2\sigma^2}\right), \tag{4.8}$$

where we set $\sigma = 0.015$ and $\alpha = 0.3$, and where v^t, x^t and y^t are the speed and coordinates of the virtual animal (rat) at time t, while x_i and y_i represent the "preferred" location of neuron i, the locations given uniformly throughout the box (and added a small error value ~ 0.005). The input is plotted for a random time in figure 4.1b.

The activity pattern is initialized by randomly setting $\frac{1}{4}$ of the neuron population at a random firing rate between 0 and 1 and the rest silent ($s_i^0 = 0$), and then forming a stabilized pattern by numerical integrating equation (4.1) with $\Delta t = 1$ms over 200ms and $\xi = 0$ as seen in figure 3.9. The simulation then consists of integrating equation (4.1) with a 1ms time step, using the same movement samples as calculated the head direction cells.

We also simulate neural activity when changing the environment by inserting k holes with radius 0.15 in the box and letting the rat perform a random walk in this novel environment. The random walk is performed as previously explained, but we deny the rat position falling within the hole, computing a new position until finding one allowed. The place fields are still distributed uniformly on a square grid lattice as before. The speed, v^t, is, in the case of recorded rat positions, linearly interpolated at 1ms time bins, and assumed constant ($v^t = 1$ for all t) in the random walk configuration.

3D place cells

Aperiodic 3D place cells are simulated given a random flight of a virtual bat in a 1m×1m×1m box, both with and without a spherical 'hole' or obstacle in the centre with radius 0.2. The extension to 3D does not signify any major difference, other than assigning a preferred xyz-coordinate to the $N = 10 \times 10 \times 10$ neurons and generalizing the connection strengths in an obvious manner.

4.1.3 Grid Cell Network

Grid Cell CANN

To simulate the grid cell network activity, we use a similar setup as Couey et al. (2013), with an $N = N_x \times N_y = 28 \times 44$-neural network and calculate the dynamics of the network once again using equation (4.1).

Every cell is given a preferred location on a two-dimensional neuronal sheet with periodic boundary conditions and a preferred direction distributed on 2×2 patches of neurons with angles $\theta_k = \frac{\pi}{2}k, k = 0, 1, 2, 3$.

The feed-forward input is then calculated by:

$$\xi_i^t = \alpha v^t \text{cos}\left(\theta^t - \theta_i\right), \tag{4.9}$$

where $\alpha = 0.15$ is the velocity modulation, v^t the speed at time t, θ_i the preferred direction of neuron i. The input is plotted for a random time in figure 4.1c. The strength of connection from neuron j to neuron i is defined by:

$$W_{ij} = W_0 H\left(R - \sqrt{(x_i - x_j - l\text{cos}(\theta_i)))^2 + (y_i - y_j - l\text{sin}(\theta_i))^2}\right) \tag{4.10}$$

where $W_0 = -0.01$, $H(\dots)$ is the Heaviside function (enforcing exclusively local inhibition), $x_i = 0, \dots, N_x - 1$ and $y_i = 0, \dots, N_y - 1$ the location of neuron i in the neuronal sheet, $R = 20$ the radial extent of the connectivity and θ_i the preferred directions and $l = 2$ is a spatial offset parameter providing asymmetry to the connectivity.

The activity pattern is initialized by randomly setting $\frac{9}{10}$ of the neuron population at a random firing rate between 0 and 1 and the rest silent ($s_i^0 = 0$), and then the pattern is stabilized by numerical integration of 4.1 with $\Delta t = 1$ms over 2000ms with $\xi = 0$ (as seen in figure 3.10).

The simulation then consists of integrating equation (4.1) with a 1ms time step over a 100min interval as the rat moves in a square box of size 1m×1m from recorded rat positions. The temporal head directions and the speeds are linearly interpolated at 1ms time bins.

Self-organizing grid cell network

We also simulate grid cell network activity by using an adaption model described by Kropff and Treves (2008) and generalized to 3D by Stella and Treves (2015). The following description is as presented in the latter paper.

A virtual bat is simulated in a volume with sides $L = 2.5$m assuming constant speed v. The path is calculated using a correlated random walk, in which the direction of movement between two time steps ($\Delta t = 10$ms) depends on the previous direction, that is randomly sampled from a normal distribution with mean the previous angle and variance equal to 0.15 in azimuth direction and 0.1 in pitch angle. The total simulated time is approximately 833 minutes.

The network consists of two layers: one input layer (place units) with $N_i = 12^3$ neurons and one output layer (grid units) with $N_o = 125$ neurons. Each grid unit receives spatial inputs from regularly arranged place cells. The input to unit i at time t is given by

$$h_t^i = \sum_j W_{ij}^{t-1} r_j^t, \tag{4.11}$$

where the weight W_{ij}^t is the connection strength from unit j to i. The place fields are modelled for each place unit j as a Gaussian bump centred in the "preferred" position x_j homogeneously distributed in the volume and given by:

$$r_j^t = \exp(-\frac{||x^t - x_j||^2}{2\sigma_p^2}). \tag{4.12}$$

x^t is the current bat position at time t, $\sigma_p = 0.05L$ is the width of firing field and $||\ldots||$ is the Euclidean distance.

The firing rate of unit i is determined by the non-linear transfer function:

$$\Psi_i^t = \frac{2}{\pi}\text{arctan}[g^t(\alpha_i^t - \mu^t)]H(\alpha_i^t - \mu^t), \tag{4.13}$$

which is subsequently normalized to have maximal firing rate equal to 1, and $H(\ldots)$ is the Heaviside function, ensuring the adaption-mediated input to unit i, α_i^t, is larger than the threshold value μ^t. This is updated by:

$$\alpha_i^t = \alpha_i^{t-1} + b_1(h_i^{t-1} - \beta_i^{t-1} - \alpha_i^{t-1}), \tag{4.14}$$

$$\beta_i^t = \beta_i^{t-1} + b_2(h_i^{t-1} - \beta_i^{t-1}), \tag{4.15}$$

where $b_2 = b_1/3, b_1 = 0.1$, providing β_i with slower dynamics than α_i. These adaptive dynamics make it more difficult for a neuron to fire for prolonged periods of time.

The gain, g^t, and the threshold are iteratively adjusted at every time step to fix the mean activity, $a = \sum_i \Psi_i^t/N_o$, and the sparsity, $s = (\sum_i \Psi_i^t)^2/(N_o \sum_i \psi_i^t)$, within a 10% relative error bound from prespecified values $a_0 = 0.1, s_0 = 0.3$:

$$\mu^{t,k+1} = \mu^{t,k} + b_3(a^k - a_0), \tag{4.16}$$

$$g^{t,k+1} = g^{t,k} + b_4 g^{t,k}(s^k - s_0). \tag{4.17}$$

$b_3 = 0.01$ and $b_4 = 0.1$, and a^k and s^k are determined at each iteration by $\mu^{t,k}$ and $g^{t,k}$, the process stopping once these are within the 10% error range and the grid cell activity is determined by equation (4.13) using the final values of the gain and threshold.

A learning process modifies the strength of the feed-forward connections by the following Hebbian rule:

$$\tilde{W}_{ij}^t = W_{ij}^{t-1} + \epsilon(\Psi_i^t r_j^t - \overline{\Psi}_j^{t-1}\overline{r}_j^{t-1}) \tag{4.18}$$

with a rate $\epsilon = 0.002$. $\overline{\Psi}_j^t$ and $\overline{r}_j^t$ are estimated mean firing rates of grid unit i and place unit j, adjusted at each time step:

$$\overline{\Psi}_j^t = \overline{\Psi}_j^{t-1} + \eta(\Psi_i^t - \overline{\Psi}_j^{t-1}), \tag{4.19}$$

$$\overline{r}_j^t = \overline{r}_j^{t-1} + \eta(r_i^t - \overline{r}_j^{t-1}), \tag{4.20}$$

where $\eta = 0.05$ is a time averaging factor. After each learning step, the provisional weights, $\tilde{W}_{ij}^t$, are normalized such that:

$$\sum_j (W_{ij}^t)^2 = 1. \tag{4.21}$$

To produce 3D grid alignment, head direction input is introduced (in contrast to the 2D model), providing two extensions to the model. Firstly, each grid unit is assigned a preferred direction (randomly assigned, the directions uniformly spanning the 4π solid angle) and secondly, additional recurrent connections are introduced in the grid layer. Thus, the overall input to unit i is given by:

$$h_i^t = f_{\theta_i}(\omega_t)(\sum_j W_{ij}^{t-1} r_j^t + \rho \sum_k W_{ik} \Psi_k^{t-\tau}), \tag{4.22}$$

where $\rho = 0.1$ is a factor setting the relative strength of the feed-forward (W_{ij}) and collateral weights W_{ik}. $\tau = 25$ time steps signifies signal transmission delay, and $f_{\theta_i}(\omega_t)$ is a tuning function, maximal when the current direction ω_t is along the preferred direction θ_i:

$$f_\theta(\omega) = c + (1-c)\text{exp}[\nu(\cos(\theta - \omega) - 1)]. \tag{4.23}$$

$c = 0.2$ is the minimum value and $\nu = 0.8$ the width of the cell tuning curve.

The collateral weights are set prior to the simulation. First, each grid cell i is assigned a preferred position (x_i, y_i, z_i), randomly chosen among the place field centres of the input layer. This defines the collateral weight strength between i and k:

$$W_{ik} = \left[f_{\theta_i}(\omega_{ik}) f_{\theta_k}(\omega_{ik}) \text{exp}\left(-\frac{d_{ki}^2}{2\sigma_f^2} \right) - \kappa \right]_+, \tag{4.24}$$

where $[\ldots]_+$ is the threshold-linear function and $\kappa = 0.05$ is an inhibition factor to favor sparse weights. ω_{ik} is the direction of the line connecting the preferred positions of i and k and $\sigma_f = 0.2L$ defines how broad the connectivity is, while

$$d_{ki}^2 = [x_i - (x_k + l\cos(\omega_{ik}))]^2 + [y_i - (y_k + l\cos(\omega_{ik}))]^2 + [z_i - (z_k + l\cos(\omega_{ik}))]^2, \tag{4.25}$$

is the squared distance between the preferred positions with an offset $l = v\tau$. Normalization of the collateral connections are performed, satisfying equation (4.21). The weights provide strong positive interaction between cells with similar preferred head direction and activation fields appropriately shifted along the same head direction.

This summarises the adaption model, and the only difference between the two-dimensional and three-dimensional model is the introduction of a head direction bias for each grid cell. The code for the 2D model was retrieved from the supplementary data of the grid cell model survey by Zilli (2012)[1], and subsequently extended (by myself) to the 3D model using the details just given.

[1] `https://senselab.med.yale.edu/modeldb/showModel.cshtml?model=144006`

4.1.4 Spike train generation and Gaussian smoothing

To elicit the data produced in experiments, we compute spike trains (discrete sets of spike time series) based on the firing rate simulated as above. This consists of converting the network activity to binary-valued data determining whether a neuron has fired (1) or not (0) at distinct time step, following a simple method proposed by Heeger (2000).

We first assume a Poisson distribution of whether a single neuron, $i = 1, \ldots, N$, has fired k times in a time interval, using the temporal average firing rate, S^t, calculated as described in the previous section as mean, i.e.:

$$P_i^t(X = k) = S_i^{tk} e^{-S_i^t}. \tag{4.26}$$

Then, for each neuron we ascertain whether it has fired by determining if the probability of the neuron firing at all in the interval $\Delta t = 1$ms is larger than a threshold, μ_i^t, drawn randomly from a uniform distribution between 0 and 1. Thus, the spike train, T, is given by:

$$T_i^t = H((1 - e^{-S_i^t}) - \mu_i^t), \tag{4.27}$$

where $H(\ldots)$ is still the Heaviside function.

After generating the spike train, we wish to produce a continuous measure of the neural activity - a continuous time series. We thus define a real, positive, time-varying function, $f_i^t : \mathbb{R} \to \mathbb{R}$, representing the firing rate for each neuron i by smoothing the spike trains by Gaussian convolution. This amounts to replacing each spike $T_i^{t_j}$ with a Gaussian distribution with mean the time value of the spike and summing the value derived from each spike at a given time based on the time difference, i.e.:

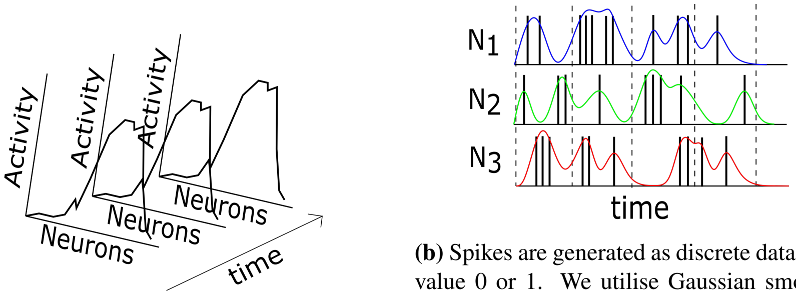
$$f_i^t = \frac{1}{\sqrt{2\pi\sigma^2}} \sum_{j=1}^{T_i} e^{\frac{-(t-t_j)^2}{2\sigma^2}}. \tag{4.28}$$

$\sigma = 15$ (time steps) and for performance reasons, the Gaussian function is truncated at $\pm 5\sigma$. We give an example of this process in figure 4.2b with the prior activity rates illustrated by three (identical, manufactured) time steps as shown in figure 4.2a).

4.2 Extracting and analysing the topological features

We have now characterized the methods used for simulating the different neural networks in study and go on to describe how to extract the topological features from this data. Our goal is to see if there are significant topological structures that could give us information about how the network functions and whether the models we use to simulate the neural activity provides the expected outcome after applying persistent homology. We also look at the robustness of the method subjected to an increasing amount of noise (in the form of adding data from multiple cell types).

Given the obtained continuous time series, we first take a random sample of neurons, n, from the different neural data sets and then sample once more at a given number of time steps, m. We do this as, in experiments, one is seldom able to record the entire population



(a) The neural activity is recorded for each time step as shown.

(b) Spikes are generated as discrete data with value 0 or 1. We utilise Gaussian smoothing to obtain a continuously valued function for each neuron (blue, green and red). We may then sample at given time steps (stippled lines).

Figure 4.2: Neural data processing.

of a network, but for a random sample. The data reduction by sampling at certain time steps is done to provide a discrete set of points, and to lessen the computation time.

The data may now be regarded as a point cloud, PC, in either $\mathbb{R}^{|n|}$ or in $\mathbb{R}^{|m|}$. Thus, in the former, one obtains a point for each time step, where the coordinate values are the neural activity of a single neuron at the given time; i.e. $PC = \{(S^t_{n_1}, \dots, S^t_{n_{|n|}})\}_{t \in m}$. In the latter way each neuron defines a point and the coordinate values corresponds to the activity of the given neuron at that time; i.e. $PC = \{(S_i^{t_1}, \dots, S_i^{t_{|m|}})\}_{i \in n}$. We will solely utilise the former method. Although the latter may prove to be successful in simulated data, it might give more difficulties in experimental data. For instance, when considering the head direction cells, the neurons usually exhibit quite noisy tuning curves, firing (at times) in more than one direction, thus (potentially) corrupting the circular feature of the network.

Before we apply persistent homology, we may also dimension-reduce the data by using principal component analysis (PCA) to lessen the noise and get clearer results. PCA is a method based on orthogonal linear transformation, transforming the data to a new coordinate system such that the greatest variance by some projection of the data comes to lie on the first coordinate (first *principal component*), the second greatest variance on the second coordinate, and so on (Jolliffe (2002)). We also test removing points based on k-nearest neighbour clustering. This implies computing the (Euclidean) distance between the points and finding the k nearest points to each point. We then keep the points which has the closest k-th neighbour, thus removing potential outliers. The two steps just described are illustrated in figure 4.3.

Given the resulting (reduced) point cloud, we then compute either the Euclidean distance or the cross correlation between the points and thus obtain a dissimilarity matrix. From this, we construct either the Vietoris-Rips or flag complex (or order complex) on which we apply persistent homology. The resulting persistence diagrams are then left to analyse.

We note that other neural models and dimension-reduction methods, as well as complexes and ways to remove outliers could have been tested, but this is not our main goal,

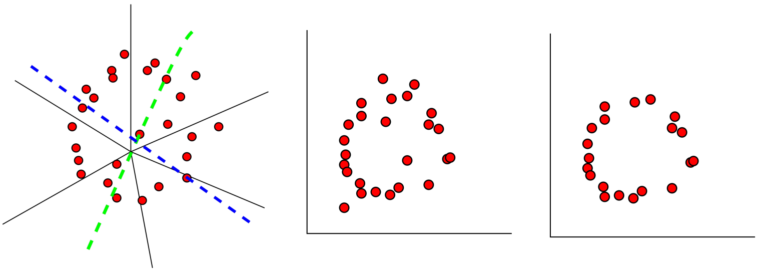


Figure 4.3: The continuous time series obtained from the simulated neural activity is sampled at discrete time values, and viewed as point in $\mathbb{R}^n$ (n equal to the number of neurons studied) (left), each neuron's activity value giving a coordinate value. The data is then reduced using PCA (middle), where the green and blue dotted lines (left) are the principal components, before points are removed based on k-clustering (right).

and we needed to limit the number of parameter choices. However, we test the proposed possibilities of dissimilarity measures, complexes, number of dimensions to which to reduce to (i.e. number of principal components used in our analysis) and number of points removed, and see how the different configurations result in different persistence diagrams. In many cases, however, the choice may seem arbitrary as we keep the results which we deem most interesting - that is, provides the clearest interpretation.

We end this chapter by giving a summary of the algorithm mentioned in this section:

1. Simulate the neural activity of head direction, place and/or grid cell networks.
2. Generate spike trains, and subsequently construct a continuous time series for each neuron.
3. Take a sample of the time series obtaining a matrix of the neural activity of n neurons at m time steps.
4. Consider the data as a point cloud in $\mathbb{R}^n$, and use PCA to dimension-reduce the data.
5. Remove outliers based on the k-nearest neighbour method.
6. Compute the dissimilarity using Euclidean distance or cross correlation between points.
7. Construct the Vietoris-Rips or flag complex.
8. Compute the persistent homology using the Eirene[2] or Ripser software.
9. Plot the persistence diagram.

[2] http://gregoryhenselman.org/eirene/index.html

Chapter 5

Results

In this chapter, we present our results, first looking at the data obtained from our neural network simulations in section 5.1, then looking at the persistence diagrams retrieved of various point clouds given by different configurations of neural networks in section 5.2.

5.1 Neural Model

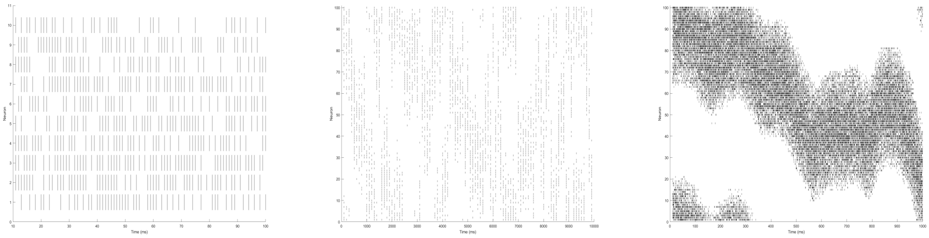
We simulate head direction, place and grid cell networks as described in chapter 4. To validate the methods, we display a sample of plots giving an idea of the data retrieved from the simulations.

Figure 5.1 shows the raster plots of a sample of the spike trains generated from the simulation of a head direction network at different time intervals. It is clear from figure 5.1c that the spikes portray a consistent correlated network behaviour. From the spike trains, we calculate continuous time series using Gaussian smoothing given as a continuous function as plotted in figure 5.2. The value of the function clearly correlates with the frequency of spikes.

The figures in 5.3 show the directional tuning of the activity of two head direction cells, two place cells and a single head direction cell with two preferred directions. The data clearly displays a directional bias for the former and latter cells (the latter clearly having two angles in which the firing rate is more frequent), while the place cell is scattered randomly when plotted against the head directions of the virtual rat. This is, however, reversed when we plot the head direction cell and a place cell against the position of the given times (see figure 5.4), the place cell exhibiting a clear (circular) place field, while the head direction cell fires at random positions.

The two aforementioned figures (5.3 and 5.4) also display the difference in neural activity before and after spike generation and Gaussian smoothing. The obtained data resembles the original, but shows somewhat less clear directional and spatial tuning (respectively), being error prone to random noise due to the methods used.

We also show the firing rate of place cells in different environments in figure 5.5. The leftmost figure shows a place cell in a box with four holes placed uniformly in the box,



(a) Spike trains for $|n| = 10$ neurons from a head direction cell network at times $m = \{1, 2, \dots, 100\}$.

(b) Spike trains for the entire ($|n| = 100$) head direction cell network at times $m = \{100, 200, \dots, 10000\}$.

(c) Spike trains from the entire ($|n| = 100$) head direction cell network at times $m = \{1, 2 \dots, 1000\}$.

Figure 5.1: Spikes from a head direction cell network after Poisson spike train generation derived from a Poisson distribution based on the firing rate simulated as explained in section 4.1.1.

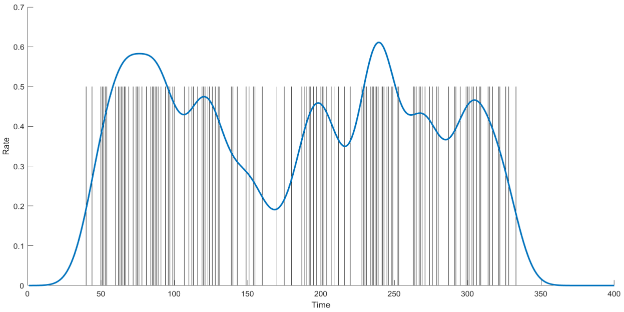
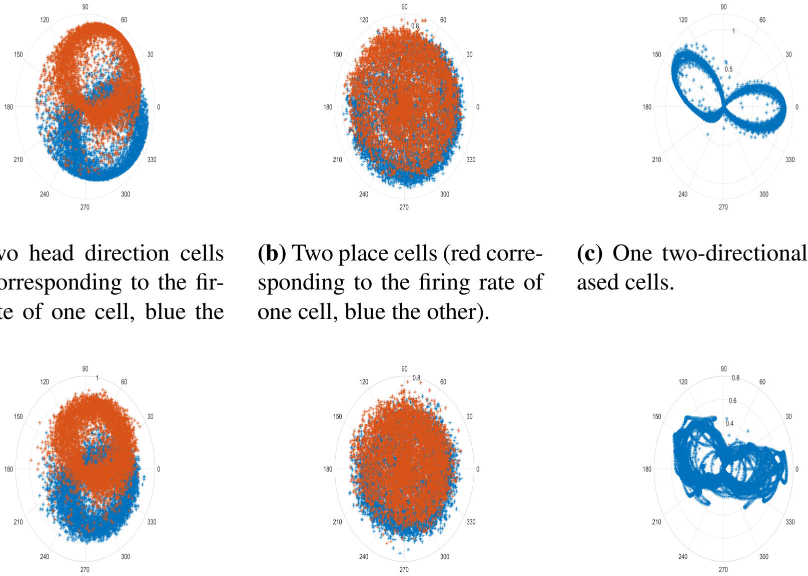


Figure 5.2: The result of Gaussian smoothing on a set of spike trains from a single neuron at times $m = \{1, 2, \dots, 400\}$. The spikes (black) are set to the value 0.5 for illustrative reasons.

the middle figure that of a place cell from a periodic place cell network in a box with one hole placed in the box centre, and the rightmost figure shows the place field for a three-dimensional (aperiodic) place cell in a $1\text{m}\times 1\text{m}\times 1\text{m}$ box.

The activity of single grid cells is plotted as a function of position in figure 5.6, using the different models described in section 4.1.3. The grid-like pattern is quite clear in the CANN model, less so in the 2D adaptive model. The latter might provide clearer results if we had let the simulation run for a longer period (the given plotted data ran for $t \sim 833$ (simulated) minutes).

The neural activity of the head direction, place and grid cell networks at a single time step is shown in figure 5.7. Both the head direction cell network and the place cell network exhibits a single activity packet - ”bump” (although the latter is split into several smaller ”bumps” due its two-dimensional nature), while the grid cell shows two activity packets (again displaying the same behaviour as the place cell network). Note that the grid cell network consists of 1232 neurons, while the other two of 100 neurons.



(a) Two head direction cells (red corresponding to the firing rate of one cell, blue the other).

(b) Two place cells (red corresponding to the firing rate of one cell, blue the other).

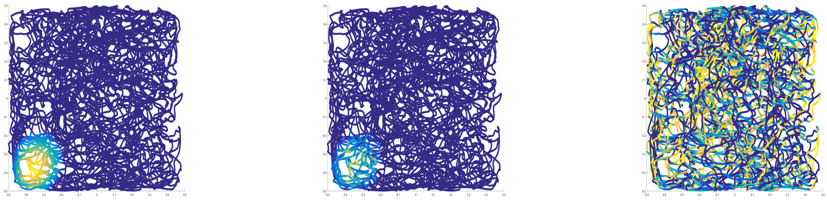
(c) One two-directionally biased cells.

(d) Data as in 5.3a after spike generation and Gaussian smoothing.

(e) Data as in 5.3b after spike generation and Gaussian smoothing.

(f) Data as in 5.3c after spike generation and Gaussian smoothing.

Figure 5.3: Tuning curves (polar plots) for different neuron networks simulated using the CANN models. Each point equals the firing rate at time t as a function of the head direction angle. The same data is shown (below) after Poisson spike train generation and Gaussian smoothing.

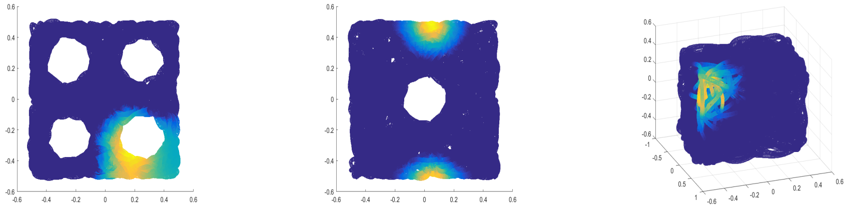


(a) Firing rate of place cell as a function of position.

(b) The data in 5.4a after spike generation and smoothing.

(c) Firing rate of a single head direction cell plotted as a function of position.

Figure 5.4: Plots of the firing rates of place and head direction cell networks shown according to the position in box. The warmer the colour, the larger the value of the neural activity, dark blue equal to zero activity.

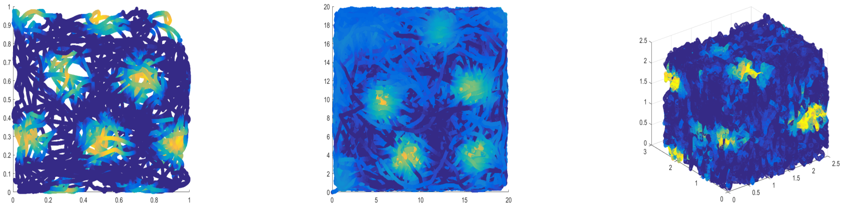


(a) Aperiodic place cell in a box with four holes.

(b) Periodic place cell activity in a box with one hole.

(c) 3D place cell in a cubic volume.

Figure 5.5: Firing rate of a place cell in different environments. The middle figure shows a cell from a place cell network simulated as a periodic two-dimensional neuronal sheet. The warmer the colour, the larger the value of the neural activity, dark blue equal to zero activity.



(a) Firing rate of a single grid cell, in which the network is modelled using the CANN model, as a function of position.

(b) Firing rate of a single grid cell, in which the network is modelled using the adaption model, as a function of position.

(c) Firing rate of a single 3D grid cell, in which the network is modelled using the adaption model, as a function of position.

Figure 5.6: Firing rate of a grid cell using different models to simulate the grid cell network activity plotted against ”animal” position. The warmer the colour, the larger the value of the neural activity, dark blue equal to zero activity.

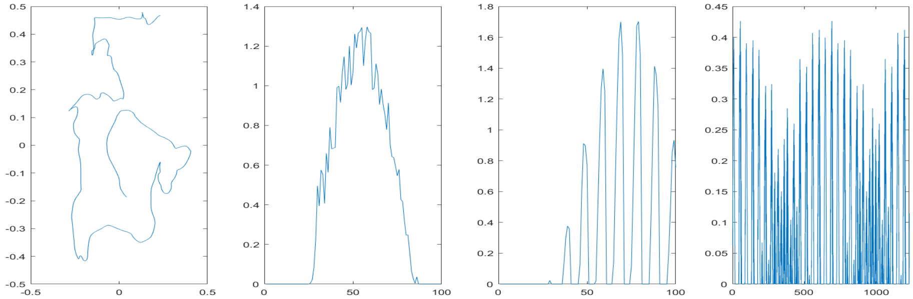


Figure 5.7: Graphs of the firing rate for a random time step for a head direction cell network (second from the left), place cell network (third from the left) and grid cell network (far right). The y-axis corresponds to the firing rate and the x-axis the different neurons numbered from 1 to N. The path of the rat in the 2D box up to the given time value is also plotted (far left).

5.2 Persistent homology applied on neural network data

Persistent homology is applied to a subset of the resulting continuous time series obtained after neural network simulation, spike generation and Gaussian smoothing as described in the previous chapter. To evaluate the results, we look at the dimensions, births and deaths of the persistent homology classes for the given data set. We do not have a defined threshold to claim a class to be 'significant', but (for the most part) look at the persistence diagram in question and see if there are any distinct, long-lived classes that may be deemed more significant than the others. From the significant classes, we may interpret the topological structures to be that of known spaces exhibiting the same homology.

When studying the head direction cell network, we look at the life lengths of only the most and the second most persistent one-dimensional persistent homology class (when these exist). We do this as we expect only a single significant persistent homology class (in addition to the essential class in zero dimensions which denotes the connectedness of the point cloud when all are connected). This gives a faster way to compare persistence diagrams than looking at each diagram, especially when we look at the response of iteratively adding the data of multiple neurons in a linear fashion to the data from the previous iteration. We postulate that if the life length of the most persistent class is large and small for the second, the data has equivalent topology to that of the 1-sphere ($\mathbb{S}^1$). If both are large this may signify either two circles or a torus (if it also has a significant two-dimensional persistence class). If both are small, we are faced with no clear structure in the data, and we deem the classes to be a consequence of noise (randomness).

In addition, we test the different configurations of our scheme, i.e. give examples of the resulting persistent homology using different complexes, dissimilarity measures, number of dimension which we reduce to and number of points removed. The basis configuration leaves the data unreduced and no points removed, and computes the Vietoris-Rips complex, measuring the Euclidean distance between the points, but we state the ways used for each instance if different from this. However, the focus is foremost on the topology of the data suggested by the persistent homology and not on the effect of varying parameters. Note that, in the following, when merely referring to a 'class', this is equivalent to a 'persistent homology class'.

5.2.1 Head Direction Cells

One head

We simulate a head direction cell network as described in section 4.1.1, obtaining an 100×241440-matrix. To analyse this and to exhibit the possibilities of the method described in 4.2, we test: i) reducing the data to different dimension by using PCA; ii) computing the distance by Euclidean distance or using correlation measures; iii) computing Vietoris-Rips complex versus order complex; iv) removing points based on the mean value of the k nearest neighbours; v) using different time steps when sampling from the total data. In all cases, we expect a single significant one-dimensional persistent homology class representing the circular feature of the head direction cell network. The neural activity is at each step randomly sampled at $|m| = 100$ time points (unless otherwise stated) for $|n| = 4, 5, \ldots, 100$ randomly chosen neurons.

The results are shown in figures 5.8, 5.9, 5.10 and 5.11 where we have plotted the life interval (y-axis) (i.e. the time of death minus the time of birth) of the longest and second longest one-dimensional persistent homology class against the number of neurons sampled from the total set (x-axis). It is (albeit with considerable variance, i.e. the life length of the most persistent class is not consistently increasing as we increase the number of neurons studied) clear that the life length of the most persistent homology class increases as we add neurons, exhibiting a true circular topology in all cases.

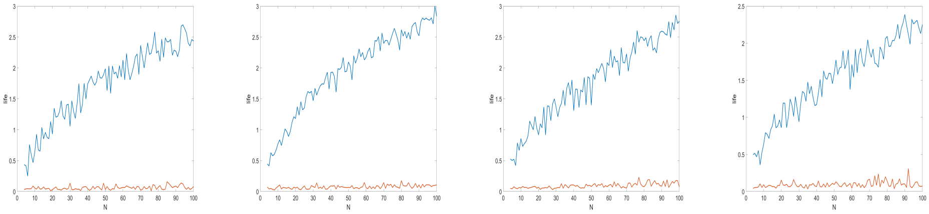
i) In figures 5.8a (3 dimensions), 5.8b (5 dimensions), 5.8c (10 dimensions), 5.8d (unreduced), we see the effect of reducing the data with PCA. All graphs exhibit quite similar behaviour - a linearly increasing trend in the life length of the most persistent class, whilst entertaining a constant life length for the second most significant class. It seems we get the least variance (and a somewhat more significant life length for the most persistent class) when dimension-reducing to 5 dimensions, but the small difference in the results when dimension-reducing the data is somewhat surprising.

ii) Figures 5.11a and 5.11b show the results when using the Spearman and cross correlation as dissimilarity measures. The life length is necessarily smaller in value compared to Euclidean distance (compare with figure 5.8d), as the largest difference between any two points is 2 (as we take it the difference to be $1 - c(i, j)$, in which $c(i, j)$ is the correlation between the times i and j, taking a value between -1 and 1). However, the correlation measures do seem to provide a single significant 1D class, the Spearman correlation showing less variance than the cross correlation, but both behaving quite similarly, the life length of the most persistent class considerably larger than the of the second most persistent class. The life length of both classes stays quite constant, the value obtained at an early stage (few neurons considered).

iii) The results when using the order complex is shown in figures 5.11c and 5.11d. In the latter, we have used the cross correlation as dissimilarity measure to retrieve the given order. The values are once more necessarily smaller as the method used only gives differences between 0 and 1. However the order complex also obtains a single significant class, validating its potential.

iv) From figures 5.10a, 5.10b and 5.10c we see the effect of removing 20 points from 120 randomly chosen time points based on the mean value of the k nearest neighbours, with $k = 3, 5$ and 10, respectively. The first gives, clearly, the best results, and portrays, when compared with figure 5.8d, somewhat less variance and longer lifetime of the most persistent class.

v) From figures 5.9a, 5.9b and 5.9c we see the effect of sampling at different times and difference in number of time steps, uniformly sampling at 100 and 1000 time steps (between times 1 and 100000), randomly at sampling 300 time steps respectively. Increasing the number of time steps considered, results in less variance and longer lifetime of the most persistent homology class, but is computationally expensive - the 97 persistence diagrams computed using 1000 time steps took about 11 hours to compute, compared to 1 minute for 100 time steps. There does not seem to be much difference in results when uniformly sampling compared to randomly sampling.



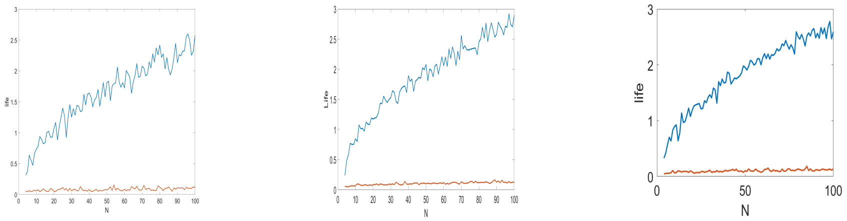
(a) The data is reduced to 3 dimensions.

(b) The data is reduced to 5 dimensions.

(c) The data is reduced to 10 dimension.

(d) The data is left unreduced.

Figure 5.8: Analysis of the effect of PCA dimension-reduction. The lifetime of the most (blue) and second most (red) persistent 1D homology class is plotted as a function of number of head direction cells accounted for.

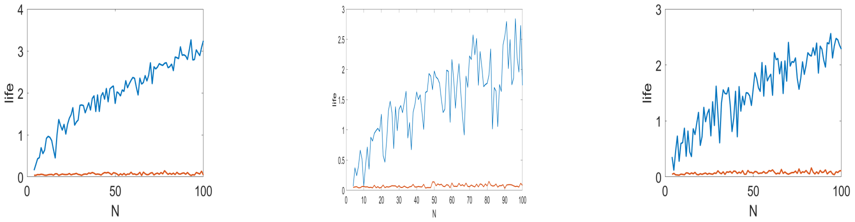


(a) The data is uniformly sampled at 100 time steps between times 1 and 100000.

(b) The data is uniformly sampled at 1000 time steps between 1 and 100000.

(c) The data is sampled randomly at 300 time steps between 1 and 241440.

Figure 5.9: Analysis of the effect of differing number of time steps and sampling method. The lifetime of the most (blue) and second most (red) persistent 1D homology class is plotted as a function of number of head direction cells accounted for.

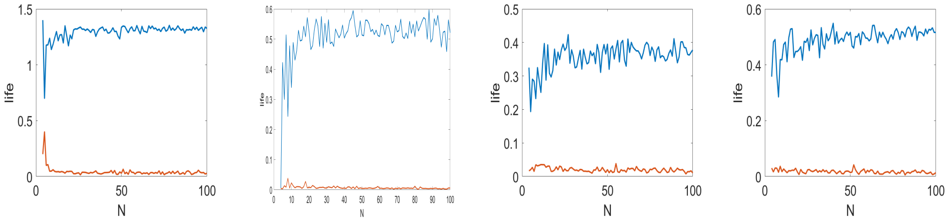


(a) 20 points are removed based on the mean value of the 3 closest neighbours.

(b) 20 points are removed based on the mean value of the 5 closest neighbours.

(c) 20 points are removed based on the mean value of the 10 closest neighbours.

Figure 5.10: Analysis of the effect of removing outliers. The lifetime of the most (blue) and second most (red) persistent 1D homology class is plotted as a function of number of head direction cells accounted for.



(a) Spearman correlation is used to compute the flag complex.

(b) Cross correlation is used to compute the flag complex.

(c) Euclidean distance is used to compute the order complex.

(d) Cross correlation is used to compute the order complex.

Figure 5.11: Analysis of the effect of differing dissimilarity measures and complexes. The lifetime of the most (blue) and second most (red) persistent 1D homology class is plotted as a function of number of head direction cells accounted for.

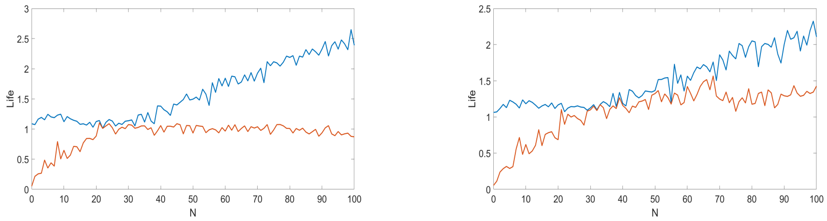
Two independent networks

In the following, we have simulated the neural activity of two head direction cell networks ('head 1' and 'head 2'), and wish to see if we can extract two significant one-dimensional persistent homology classes representing the topology of the arranged cell networks. The two networks receive different head direction input, namely, from two different time sets, the data for head 1 taken from the first half of the simulation, while that of head 2 from the second half of the simulation. In this way we get two completely independent heads. This might describe the neural activity of either two rats or one rat with two uncorrelated heads.

In the figures in 5.12, we have plotted the two most persistent classes as we iteratively add an increasing number of neurons from head 2 to a "basis" random sample of neurons from head 1. This basis consists of the activity $|n_1| = 30, 70$ neurons and is identical throughout, while the neurons accounted for, which belong to head 2, is sampled randomly at each iteration with $|n_2| = 1, 2, \ldots, 100$. For each iteration, we uniformly sample $|m| = 100$ time steps between times 1 and 100000. The data thus consists of 100 points in $\mathbb{R}^{|n_1|+|n_2|}$.

We observe that the lifetime of the second most persistent class increases linearly until it surpasses (and becomes) the most persistent class which stays practically constant. This happens when $|n_1| = |n_2|$, and it seems clear that the former class represents head 2 while the latter represents head 1. The method thus extracts the topology of two circles belonging to the two head direction cell networks.

However, when only looking at the one-dimensional homology classes, we omit to see if there are other topological spaces which may give the same results. In the next section, we will see that we may expect the data to have the topology of a torus, and simultaneously, we try to see the effect of different angle inputs.



(a) Initial data from the neural activity of 30 neurons from head 1.

(b) Initial data from the neural activity of 70 neurons from head 1.

Figure 5.12: Plots of the lifetime of the most (blue) and second most (red) persistent class obtained from applying persistent homology on the neural activity of two head direction cell networks. The x-axis denotes the number of randomly selected neurons accounted for from head 2.

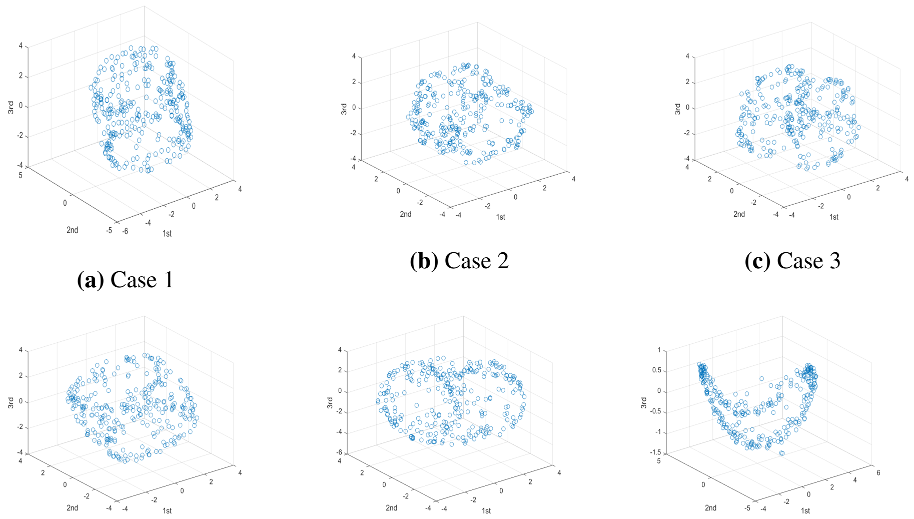
Different angle inputs

In this section, we simulate six instances of neural activity for two separate head direction cell networks receiving varying head direction input and plot the persistence diagram (with 0-, 1- and 2-dimensional classes) of the resulting data, considering all neurons in the networks $|n| = 100$ and $|m| = 300$ randomly sampled time steps. The two networks might serve to represent the activity from two different heads correlated in different manners.

We call the angles used for calculating the feed-forward input to head 1 and head 2 for a_1 and a_2 respectively. The following bullet points list the angle updates at each time step. We keep all angles within the periodic interval $[-\pi, \pi]$, altering the angles ($\pm 2\pi$) after each iteration if outside the boundaries. In the first four cases, a_1 is initiated at a randomly given angle and a_2 is set to $a_1 + \pi$, the two heads thus starting by looking in opposite directions, while in the two last cases, both angles are independently, randomly initiated.

- **Case 1** $a_1^{t+1} = a^t, \quad a_2^{t+1} = a^{t+t_{\text{end}}}.$
- **Case 2** $a_1^{t+1} = a_1^t + 0.1, \quad a_2^{t+1} = a_2^t + 0.1.$
- **Case 3** $a_1^{t+1} = a_1^t + 0.1, \quad a_2^{t+1} = a_2^t - 0.1.$
- **Case 4** $a_1^{t+1} = a_1^t + 0.1\upsilon_1, \quad a_2^{t+1} = a_2^t - 0.05\upsilon_2.$
- **Case 5** $a_1^{t+1} = a_1^t + \frac{\pi}{8}\zeta_1, \quad a_2^{t+1} = a_2^t + \frac{\pi}{8}\zeta_2.$
- **Case 6** $a_1^{t+1} = \alpha_1^t + \zeta_1', \quad a_2^{t+1} = \alpha_2^t + \zeta_2'.$

Case 1 is the same as that described in the previous section, where a is the ’original’ angle data, obtained by interpolating the rat movement recorded as done in the case when looking at only one head direction cell network as in section 5.2.1. t is the current time step, while $t_{\text{end}} = 100000$ is the end of the first time interval. υ_1 and υ_2 are random numbers drawn from a uniform distribution between 0 and 1, while ζ_1 and ζ_2 are drawn from a normal distribution, $\mathscr{N}(0, 0.1)$, α_1^t and α_2^t from $\mathscr{N}(a_1^t, \pi/12)$ and $\mathscr{N}(a_2^t, \pi/12)$, respectively, and ζ_1' and ζ_2' from $\mathscr{N}(\mathrm{d}(a^t, a_1^t), 0.02)$ and $\mathscr{N}(\mathrm{d}(a^t, a_1^t), 0.02)$, where $\mathrm{d}(-,-)$ is the angular difference with values in $[-\pi, \pi]$.



(a) Case 1

(b) Case 2

(c) Case 3

(d) Case 4

(e) Case 5

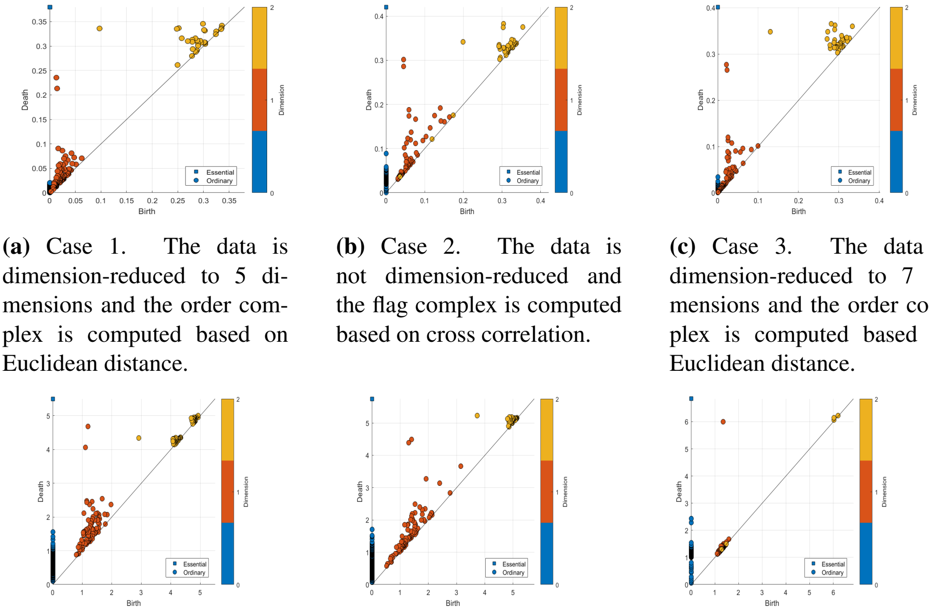
(f) Case 6

Figure 5.13: Scatter plots along the three first principal components of the neural activity of head 1 and 2 receiving different angle input, considering $|n| = 100$ neurons in each network, and $|m| = 300$ randomly drawn time steps.

To visualize the data structure, we show in figure 5.13 the dimension-reduced neural activity of the six cases mentioned above, using the three first principal components as given by PCA. Although somewhat unclear (as it is difficult to portray three-dimensional data in two dimensions), we observe a torus-like structure in all cases, but for case 6. In figure 5.14 (where we selected the parameter configurations of the algorithm which gave the clearest results in each case), we see that we obtain equivalent (persistent) homology to that of a torus as expected - in cases 1-5 we obtain two quite similarly-lived one-dimensional classes, as well as an (albeit less) significant two-dimensional class (in addition to an essential 0D PH class). Thus, we prove that, given different angle inputs, we may find a toroidal topological structure in the neural data based on our method, describing the two degrees of freedom expressed by the two head direction cell networks. In case 6 (as was clear from figure 5.13f) we only obtain a single one-dimensional PH class (see figure 5.14f). The reason for this is that the two head direction inputs correlated almost perfectly.

Two-directional head direction network

We simulate a head direction cell network in which each cell is given two preferred head directions and given input accordingly, the neural activity of one cell shown in figure 5.3f. We reduce the data to 10 dimensions using PCA after randomly sampling $|m| = 300$ time steps for $|n| = 100$ neurons. Removing 10% of the points based on the distance to the 5th nearest neighbour is also tested, and the result of applying persistent homology to these



(a) Case 1. The data is dimension-reduced to 5 dimensions and the order complex is computed based on Euclidean distance.

(b) Case 2. The data is not dimension-reduced and the flag complex is computed based on cross correlation.

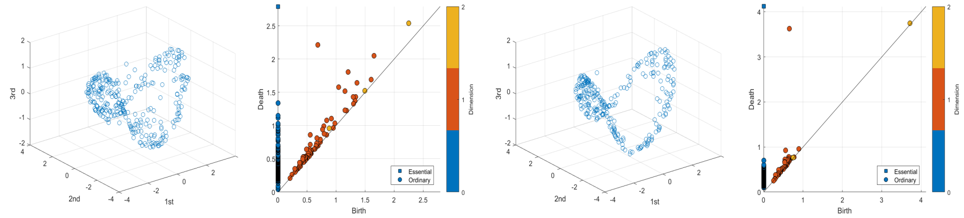
(c) Case 3. The data is dimension-reduced to 7 dimensions and the order complex is computed based on Euclidean distance.

(d) Case 4. The data is dimension reduced to 7 dimensions and the Vietoris-Rips complex is computed.

(e) Case 5. The data is dimension-reduced to 5 dimensions and the Vietoris-Rips complex is computed.

(f) Case 6. The data is not dimension-reduced and the Vietoris-Rips complex is computed.

Figure 5.14: Persistence diagrams of the different cases of angle input. All data sets consist of 300 randomly sampled time points (no points removed) from all neurons in both head cell networks each of population size 100.



(a) Point cloud of two-directional head direction cell network plotted along its three first principal components.

(b) Persistence diagram of the point cloud in (a), considering the first 10 principal components.

(c) The point cloud in (a), in which 10% of the points have been removed based on the distance to the 5th nearest neighbour.

(d) Persistence diagram of the point cloud in (c), considering the first 10 principal components.

Figure 5.15: Analysis of two-directional head direction cell network activity. 300 points of the reduced neural activity of the $|n| = 100$ neurons has been sampled.

data sets is shown in figure 5.15. We get a significant 1D class in both cases, more clearly after removing 10% of the points. One could have expected to find the topology of a torus, but as the input angle is the same, the movement of the two activity packets of the network is correlated and they exhibit a single circular network behaviour.

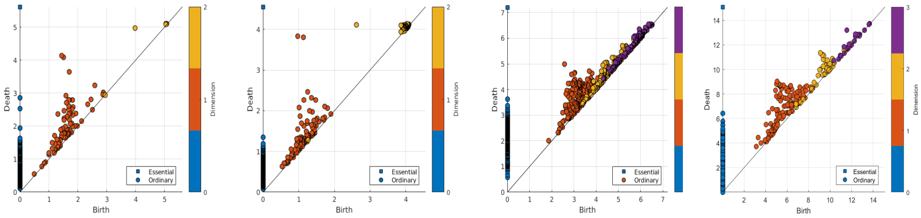
3D head direction cells

We simulate four different (extreme) cases of 3D head direction cell networks:

- **Case i** Two ring head direction CANNs accounting for azimuth and pitch angle separately.
- **Case ii** A toroidal head direction cell network where all cells are conjunctively given both preferred azimuth and pitch head directions, i.e. arranged as a 2D neural sheet with periodic boundaries.
- **Case iii** Three ring head direction CANNs, accounting for azimuth, pitch and roll angle separately.
- **Case iv** A head direction network of conjunctive neurons in which all cells are given preferred azimuth, pitch and roll angles, i.e. arranged as a cube with periodic boundaries. These three degrees of freedom could be described by the real projective space $P^3(\mathbb{R})$.

The neural connections for case 2 and 4 are computed similarly as in the one-dimensional case, only generalized to two and three dimensions, respectively.

We randomly sample $|m| = 300$ time steps of the entire networks ($|n_1| = 200$, $|n_2| = 400$, $|n_3| = 300$ and $|n_4| = 1000$) from the continuous time series obtained after simulation of the four cases. The persistent homology of the four data sets are shown in 5.16. Both case 1 and 2 show the homology of a torus, the latter displaying clearer results than the former.



(a) Case i. **(b)** Case ii. **(c)** Case iii. **(d)** Case iv.

Figure 5.16: Persistence diagrams describing the 3D head direction cell network activity simulated according to the four cases described in section 5.2.1 as a virtual animal explores a 1m^3 box. The point cloud is sampled at 300 points and reduced to 10 dimensions in all cases.

The Betti numbers of $P^3(\mathbb{R})$ is: $\beta_0 = 1, \beta_1 = 1, \beta_2 = 0, \beta_3 = 1$ and $\beta_i = 0$ for $i > 3$, but we do not seem to obtain this in case 3 or 4, where (except for an essential zero-dimensional class) only a single significant 1D class in case 3 and perhaps one in dimension 3 for case 4 is seen. However, we deem these classes as not clear enough to make any interpretation, and despite testing multiple runs, we were not any more successful in retrieving a clear topology, perhaps due to the small number of points considered.

We note that the toroidal representation of the head direction network suggested by Finkelstein et al is debated. According to Andre J. Noest (who comments on the paper by Finkelstein et al. (2015)), "toroidal head direction coding would make bats dizzy", arguing that the structure of physical head-rotations could be described by the $SO(3)$-group.[1] As we do get the topology of a torus (when modelling our network as such) using this method, we propose that this scheme might provide an answer to what type of structure 3D head direction cells actually forms if applied to experimental data.

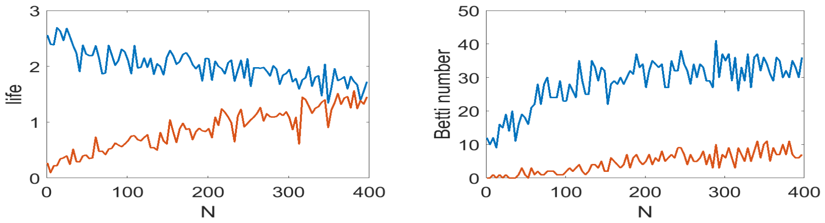
5.2.2 Place Cells

We simulate place cells as described in section 4.1.2. Considering the neural activity of aperiodic place cells in a regular box, we expect only a significant 0D class. However, when introducing circular obstacles (holes) in the box (and keeping the place fields uniformly distributed), we believe the topology of the space will be encoded by the neural activity and thus represented when applying persistent homology.

Aperiodic network

To begin with, we observe how the complexity (represented by the Betti numbers) and the two most persistent classes behave of the neural activity of 100 neurons from a head direction cell network as we iteratively increase the number of neurons added from a place cell network simulated as a rat locomotes throughout a regular 2D box. At each iteration, the point cloud is sampled at 100 random times. We plot the first and second Betti number (y-axis) as a function of the number of place cells added (x-axis) in figure 5.17a. The

[1] `https://www.nature.com/nature/journal/v517/n7533/full/nature14031.html` Accessed: 30.05.2017



(a) Life time of the most (blue) and second most (red) persistent class .

(b) β_1 (blue) and β_2 (red).

Figure 5.17: The analysis of the point clouds sampled from the neural activity of a head direction cell network and an incrementally number of neurons from a place cell network (x-axis) at 100 random time points at each step and reduced to 5 dimensions.

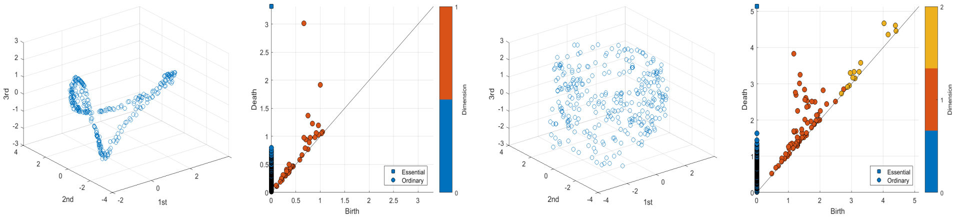
complexity increases somewhat, but not significantly as we add neurons. The 0th Betti number is not plotted as it stays constantly equal to 100, as the number of points does not alter. This also explains the limited difference in the 1st and 2nd Betti number. The clear, significant class representing the head direction cell network gradually becomes less clear and is finally indistinguishable from the other 1D classes (represented by the second most persistent class). This shows that the method is somewhat affected by the presence of noise in the form of place cell activity, i.e. is not able to clearly distinguish the head direction cell network.

Next, we let the virtual rat perform a random walk in a box where we have introduced a single circular obstacle placed at the centre and simulate both the activity of a head direction cell network and a place cell network. The analysis of this is shown in 5.18a-5.18d, and despite two clear circles in the reduced data of the cells when the networks are displayed separately (see the three first principal components of the sampled place cell network activity in figure 5.18a and the corresponding persistence diagram in 5.18b), it seems we retrieve the topology of a cylinder when both networks are analysed simultaneously (see figures 5.18c and 5.18d).

A second simulation is similarly done, but now we introduce four holes placed regularly inside the same box (the activity of a single place cell and the holes clearly shown in 5.5a). In 5.19a we see four significant 1D classes when considering the activity of the entire place cell network sampled uniformly at 300 time steps. However, we obtain merely one significant 1D class once the head direction network is taken into account, this data thus corrupting the topology of the environment recreated in the former.

Periodic network

The neural activity of a periodic place cell network is also simulated, letting the ’rat’ locomote a 2D box both with and without a hole in its centre. In the latter case, we would expect to retrieve the topology of a torus as the place cell sheet identifies top with bottom and left with right. This is indeed (although not particularly clearly) seen to be the case in figure 5.20a. In the case of the ’punctured’ environment, we see that this network retains its (toroidal) structure, and does not reveal the topology of the environment (see figure



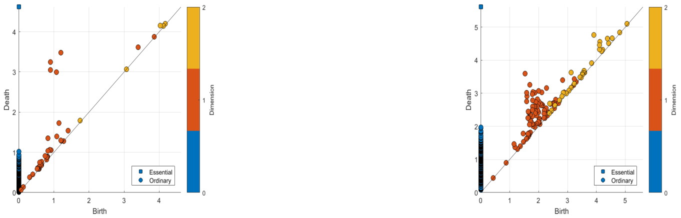
(a) The aperiodic place cell network activity point cloud plotted along its three first principal components.

(b) Persistence diagram of the point cloud in (a), considering the first 10 principal components.

(c) The point cloud from the activity of both networks plotted along its three first principal components.

(d) Persistence diagram of the point cloud in (c), considering the first 5 principal components.

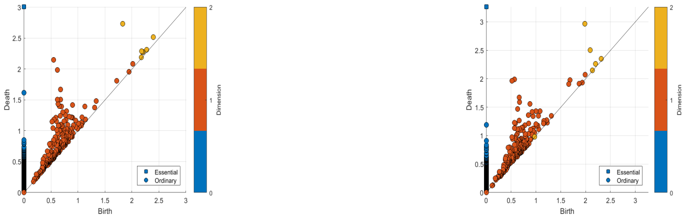
Figure 5.18: The analysis of the simulated neural activity of place cell and head direction cell networks from a simulation in which the explored environment is a box with a hole placed in the centre. The data accounts for 400 place cells and 100 head direction cells and the activity is sampled at 300 random time points.



(a) Persistence diagram of the aperiodic place cell network activity.

(b) Persistence diagram of the point cloud of both place and head cell network activity.

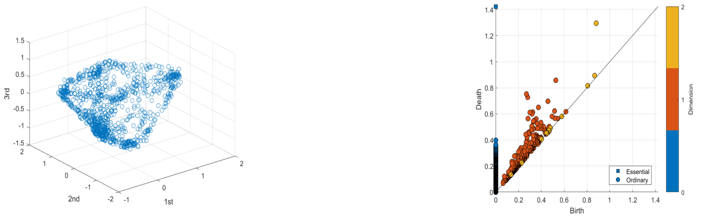
Figure 5.19: The analysis of the simulated neural activity of place cell and head direction cell networks from a simulation in which explored environment is a box with four hole placed regularly within. The data accounts for 400 place cells and 100 head direction cells and the activity is sampled at 300 regularly spaced time points, and is reduced to 7 dimensions. The head cells obscure the four clear 1D classes seen in (a), and we seemingly get only one significant persistent homology class in (b).



(a) Persistence diagram of the periodic place cell activity simulated as a rat explores a 2D box.

(b) Persistence diagram the periodic place cell activity. The environment is that of a 2D box with a circular hole placed in the centre.

Figure 5.20: The analysis of the simulated neural activity of a periodic place cell network. The data accounts for 400 place cells and the activity is sampled at 1000 random time points, before dimension-reduced to 10 dimensions.



(a) The point cloud of the neural activity plotted along its three first principal components.

(b) Persistence diagram derived from the neural activity.

Figure 5.21: The analysis of neural activity of a 3D place cell network simulated as a virtual bat flies inside a cubic volume of $1\text{m}\times1\text{m}\times1\text{m}$, with a spherical hole of radius 0.2 placed in the centre.

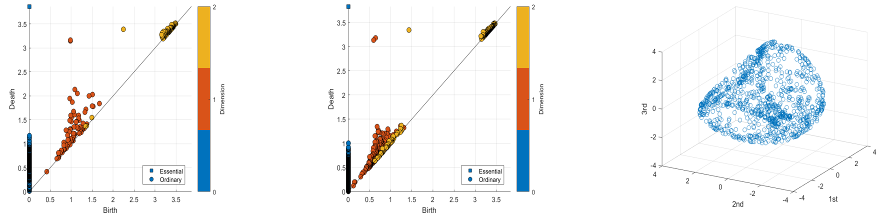
5.20b).

3D place cells

We simulate the neural activity of an aperiodic 3D place cell network with ball-shaped place fields (see figure 5.5c) as a virtual bat flies through a cubic volume with a spherical object (hole) placed in the centre of the cube.

The data is plotted along the three first principal components as calculated using PCA in figure 5.21a, and consists of 1000 points corresponding to the neural activity of $|n| = 1000$ place cells at $|m| = 1000$ randomly sampled time steps, reduced to five dimensions.

The persistence diagram plotted in figure 5.21b shows a significant two-dimensional class which might represent the $\mathbb{S}^2$-topology of the environment the bat traverses.



(a) Persistence diagram for 300 sampled points.

(b) Persistence diagram for 1000 sampled points.

(c) The grid cell activity point cloud (300 points) plotted along its three first principal components.

Figure 5.22: Analysis of the neural activity of a grid cell network simulated using the CANN model. The entire network data is studied ($|n| = 28 \times 44$), and we look at two sets of points sampled, one consisting of 300 point, the other of 1000, both sampled at random time steps. We leave the data unreduced prior to applying persistent homology.

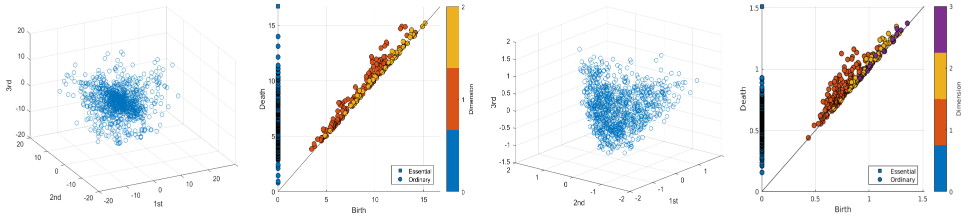
5.2.3 Grid Cells

CANN model

We simulate the activity of a grid cell network using the CANN model as explained in section 4.1.3. Given the periodic boundary conditions, we once more expect to find the topology of a torus. From figure 5.22a and 5.22b, where we have applied persistent homology on 300 and 1000 points based on 300 and 1000 random time points of the activity of the entire network ($n = 1232$), respectively, this indeed seems to be the case. We also plot the former point cloud along the three first principal components as shown in figure 5.22c.

Adaption model

We simulate the activity of a 2D and a 3D grid cell network using the self-organizing model as explained in section 4.1.3. This is done to compare with the method above and see if the self-organizing grid cell network with single unit dynamics may ultimately display a coherent network structure. However, this is not the case as we see from the persistence diagrams in figures 5.23b and 5.23d. Here we have applied persistent homology to the sampled activity point cloud consisting of 300 points sampled from at 300 random time steps including the activity of all the neurons simulated in the respective networks ($|n| =$ 25 grid cells for the 2D case and $|n| = 125$ for the 3D case). From figures 5.23b and 5.23c, where we have plotted 100 points from the two data sets according to the three first principal components, the point clouds seem somewhat randomly scattered, but, especially in the 3D case, both display a sort of structure resembling a compact tetrahedron, probably caused by applying PCA.



(a) 1000 randomly sampled points plotted along its three first principal components.

(b) Persistence diagram of the unreduced, 300 randomly sampled points.

(c) 1000 randomly sampled points plotted along its three first principal components.

(d) Persistence diagram of 300 randomly sampled points reduced to 7 dimensions.

Figure 5.23: Analysis of the activity of grid cell networks simulated using the adaption model. The 2D case is considered in 5.23b and 5.23a, and the 3D case in 5.23d and 5.23c. In both cases the activity of the network (25 and 125 neurons respectively) is included.

Chapter 6

Conclusion

In this thesis, we have tested a novel method based on persistent homology in detecting the topology of the activity of neuron populations of head direction cells, place cells and grid cells. This is done to better understand the dynamics of neurons playing a part in encoding the representation of space, and subsequently, to serve as a test of the feasibility of a neural network model. We will now try to answer the two questions posed in the introduction.

The method has in many cases proven successful in finding the expected topology of the networks. It is especially robust when considering the head direction cell network activity, which exhibits a clear ring topology. We have also revealed the topology of the environment a rat is navigating in, based on place cell network activity, and found the toroidal topology of the two-dimensional grid cell CANN, but could not detect any interesting topology when modelling the grid cell activity using the adaption model. We propose that our findings support persistent homology as a tool to determine how these networks function, and could be used on experimental data. Particularly, it would be compelling to test this method on the experimental data in the cases of 3D head direction cells and grid cells.

However, the results are not insensitive to noise and the definition of a *significant* persistent homology class seems somewhat unclear. We have thus not attained a clear measure of the robustness of the method, but have seen that the method detects the topology expected also subjected to noise, and, for instance, in the case of head direction cells we get a quite clear ring topology even when only considering the neural activity of four randomly chosen neurons at a hundred random time steps. The field of the statistics of persistent homology is a topic of active study, with persistence landscapes and Betti number distribution serving as the two most prominent methods as of now. Both could have been utilised in this thesis to get a clearer mark of the results. We also note that the use of the *witness complex* could have proved interesting as it could have alleviated us of our somewhat awkward way of sampling from the neural activity.

Additionally, there does not seem to be a unique configuration of the parameters involved in the algorithm (the number of points removed, dimensions which we reduce the data to, points sampled, as well as the ways in which to sample the points and what complexes to use) which provides the best results in all cases, but we have seen that varying the parameters may be helpful in revealing the topology. Thus, we propose that if the same significant persistent homology is obtained when altering the configuration, this then serves as a verification of the results.

Bibliography

Arai, M., Brandt, V., Dabaghian, Y., 2014. The effects of theta precession on spatial learning and simplicial complex dynamics in a topological model of the hippocampal spatial map. PLoS Comput Biol 10 (6), e1003651.

Babichev, A., Ji, D., Mémoli, F., Dabaghian, Y. A., 2016. A topological model of the hippocampal cell assembly network. Frontiers in Computational Neuroscience 10.

Botnan, M. B., 2011. Three approaches in computational geometry and topology: Persistent homology, discrete differential geometry and discrete morse theory. Master's thesis, Institutt for matematiske fag.

Boucheny, C., Brunel, N., Arleo, A., 2005. A continuous attractor network model without recurrent excitation: maintenance and integration in the head direction cell system. Journal of computational neuroscience 18 (2), 205–227.

Burak, Y., Fiete, I. R., 2009. Accurate path integration in continuous attractor network models of grid cells. PLoS Comput Biol 5 (2), e1000291.

Couey, J. J., Witoelar, A., Zhang, S.-J., Zheng, K., Ye, J., Dunn, B., Czajkowski, R., Moser, M.-B., Moser, E. I., Roudi, Y., et al., 2013. Recurrent inhibitory circuitry as a mechanism for grid formation. Nature neuroscience 16 (3), 318–324.

Curto, C., Itskov, V., 2008. Cell groups reveal structure of stimulus space. PLoS Comput Biol 4 (10), e1000205.

Dabaghian, Y., Brandt, V. L., Frank, L. M., 2014. Reconceiving the hippocampal map as a topological template. Elife 3, e03476.

Dabaghian, Y., Mémoli, F., Frank, L., Carlsson, G., 2012. A topological paradigm for hippocampal spatial map formation using persistent homology. PLoS computational biology 8 (8), e1002581.

Dunn, B., 2016. Functional reconstruction of a grid cell network. Ph.D. thesis, Norwegian University of Science and Technology & Kavli Institute for Systems Neuroscience Centre for Neural Computation.

Edelsbrunner, L. . Z., 2002. Topological persistence and simplification. Discrete Comput. Geom 28, 511, doi:10.1007/s00454-002-2885-2.

Eliasmith, C., 2007. Attractor network. Scholarpedia 2 (10), 1380, revision #91016.

Finkelstein, A., Derdikman, D., Rubin, A., Foerster, J. N., Las, L., Ulanovsky, N., 2015. Three-dimensional head-direction coding in the bat brain. Nature 517 (7533), 159–164.

Gerstner, W., Kistler, W. M., Naud, R., Paninski, L., 2014. Neuronal dynamics: From single neurons to networks and models of cognition. Cambridge University Press.

Ghrist, R., 2014. Elementary applied topology. Createspace.

Giusti, C., Ghrist, R., Bassett, D. S., 2016. Two's company, three (or more) is a simplex. Journal of computational neuroscience 41 (1), 1–14.

Giusti, C., Pastalkova, E., Curto, C., Itskov, V., 2015. Clique topology reveals intrinsic geometric structure in neural correlations. Proceedings of the National Academy of Sciences 112 (44), 13455–13460.

Grieves, R. M., Jeffery, K. J., 2017. The representation of space in the brain. Behavioural Processes 135, 113–131.

Hafting, T., Fyhn, M., Molden, S., Moser, M.-B., Moser, E. I., 2005. Microstructure of a spatial map in the entorhinal cortex. Nature 436 (7052), 801–806.

Heeger, D., 2000. Poisson model of spike generation. Handout, University of Standford 5, 1–13.

Hermansen, E., 2017. Topological data analysis - a survey of persistent homology and its applications, unpublished article.

Hoffman, K., Babichev, A., Dabaghian, Y., Jan. 2016. Topological mapping of space in bat hippocampus. ArXiv e-prints.

Jolliffe, I., 2002. Principal component analysis. Wiley Online Library.

Kropff, E., Treves, A., 2008. The emergence of grid cells: Intelligent design or just adaptation? Hippocampus 18 (12), 1256–1269.

Langston, R. F., Ainge, J. A., Couey, J. J., Canto, C. B., Bjerknes, T. L., Witter, M. P., Moser, E. I., Moser, M.-B., 2010. Development of the spatial representation system in the rat. Science 328 (5985), 1576–1580.

McNaughton, B. L., Battaglia, F. P., Jensen, O., Moser, E. I., Moser, M.-B., 2006. Path integration and the neural basis of the'cognitive map'. Nature Reviews Neuroscience 7 (8), 663–678.

Moser, E., Moser, M., 2007. Grid cells. Scholarpedia 2 (7), 3394, revision #144456.

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

Muller, R. U., Kubie, J. L., 1987. The effects of changes in the environment on the spatial firing of hippocampal complex-spike cells. Journal of Neuroscience 7 (7), 1951–1968.

O'Keefe, J., 1976. Place units in the hippocampus of the freely moving rat. Experimental neurology 51 (1), 78–109.

O'keefe, J., Nadel, L., 1978. The hippocampus as a cognitive map. Oxford: Clarendon Press.

Poucet, B., Chaillan, F., Truchet, B., Save, E., Sargolini, F., Hok, V., 2015. Is there a pilot in the brain? contribution of the self-positioning system to spatial navigation. Frontiers in behavioral neuroscience 9.

Poucet, B., Sargolini, F., 2013. A trace of your place. Science 340 (6128), 35–36.

Samansonovich, A., 2017. Continuous attractor network, unpublished article.
URL http://www.scholarpedia.org/article/Continuous_attractor_network

Samsonovich, A., McNaughton, B. L., 1997. Path integration and cognitive mapping in a continuous attractor neural network model. Journal of Neuroscience 17 (15), 5900–5920.

Sanders, H., Rennó-Costa, C., Idiart, M., Lisman, J., 2015. Grid cells and place cells: an integrated view of their navigational and memory function. Trends in neurosciences 38 (12), 763–775.

Singh, G., Memoli, F., Ishkhanov, T., Sapiro, G., Carlsson, G., Ringach, D. L., 2008. Topological analysis of population activity in visual cortex. Journal of vision 8 (8), 11–11.

Skaggs, W. E., Knierim, J. J., Kudrimoti, H. S., McNaughton, B. L., 1995. A model of the neural basis of the rat's sense of direction. Advances in neural information processing systems, 173–182.

Skaggs, W. E., McNaughton, B. L., 1996. Theta phase precession in hippocampal. Hippocampus 6, 149–172.

Spreemann, G., Dunn, B., Bakke Botnan, M., Baas, N., Oct. 2015. Using persistent homology to reveal hidden information in neural data. ArXiv e-prints.

Stella, F., Treves, A., 2015. The self-organization of grid cells in 3d. eLife 4, e05913.

Taube, J., 2009. Head direction cells. Scholarpedia 4 (12), 1787, revision #91349.

Yartsev, M. M., Ulanovsky, N., 2013. Representation of three-dimensional space in the hippocampus of flying bats. Science 340 (6130), 367–372.

Zhang, S.-J., Ye, J., Miao, C., Tsao, A., Cerniauskas, I., Ledergerber, D., Moser, M.-B., Moser, E. I., 2013. Optogenetic dissection of entorhinal-hippocampal functional connectivity. Science 340 (6128), 1232627.

Zilli, E. A., 2012. Models of grid cell spatial firing published 2005–2011. Frontiers in neural circuits 6, 16.

Zomorodian, A., Carlsson, G., 2005. Computing persistent homology. Discrete & Computational Geometry 33 (2), 249–274.