Article

Dynamic Graph Learning: A Structure-Driven Approach

Bo Jiang [1],*, Yuming Huang [2] , Ashkan Panahi [3], Yiyi Yu [4], Hamid Krim [1] and Spencer L. Smith [4]

[1] Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC 27695, USA; ahk@ncsu.edu

[2] Department of Physics, North Carolina State University, Raleigh, NC 27695, USA; yhuang26@ncsu.edu

[3] Department of Computer Science and Engineering, Chalmers University of Technology, SE-412 96 Göteborg, Sweden; panahi1986@gmail.com

[4] Department of Electrical and Computer Engineering, University of California Santa Barbara, Santa Barbara, CA 93106, USA; yuyiyi7@gmail.com (Y.Y.); sls@ucsb.edu (S.L.S.)

* Correspondence: bjiang8@ncsu.edu; Tel.: +1-(919)-522-1733

Abstract: The purpose of this paper is to infer a dynamic graph as a global (collective) model of time-varying measurements at a set of network nodes. This model captures both pairwise as well as higher order interactions (i.e., more than two nodes) among the nodes. The motivation of this work lies in the search for a connectome model which properly captures brain functionality across all regions of the brain, and possibly at individual neurons. We formulate it as an optimization problem, a quadratic objective functional and tensor information of observed node signals over short time intervals. The proper regularization constraints reflect the graph smoothness and other dynamics involving the underlying graph's Laplacian, as well as the time evolution smoothness of the underlying graph. The resulting joint optimization is solved by a continuous relaxation of the weight parameters and an introduced novel gradient-projection scheme. While the work may be applicable to any time-evolving data set (e.g., fMRI), we apply our algorithm to a real-world dataset comprising recorded activities of individual brain cells. The resulting model is shown to be not only viable but also efficiently computable.

Keywords: dynamic graph learning; graph signal processing; sparse signal; convex optimization

1. Introduction

The increased and ubiquitous emergence of graphs provides a great approach for quantifying interaction between different elements in a great variety of network-based applications. Analyzing and discovering the underlying structure of a graph for a given dataset has become central to various signal processing research problems with such a potential interpretation. For example, in social media [1] (e.g., Facebook and LinkedIn), the basic interaction/relation between two individuals being represented by a link, yields the notion of a graph known as the social network, which is used for inferring further characteristics among all involved individuals [2]. The power of the graph representation paradigm is in fact so fundamental that it captures the very structure of life [3,4], namely the interactions among atoms/molecules to reflect the behavior of different materials or bacteria. The rather recent connectome paradigm [5] in neuroscience is based on the hypothesis that the brain is a massively connected network due to its activity and connectivity. The network connectivity, hence behavioral variation, is in response to an external stimulus, may be used to investigate a brain's structure and its functionality [6,7]. In our validation of this framework, we consider sparse images of the brain's visual cortex where non-zero components vary in time. Our proposed approach is just as applicable to dense images as well as medium-dense images such as Functional Magnetic Resonance Imaging data. Our ready access to and experience in the data at hand thus influenced the choice.

Establishing connectivity among neurons, currently follows two main approaches, (i) Noise correlation analysis [8,9], often used in neuroscience to uncover the connectivity among neurons, (ii) Static graph learning [10,11] used to extract a fixed graph over time. When noise correlation is significant, it is commonly used in neuroscience as a short-time connectivity metric between every pair of neurons. Requiring abundant observations, this method cannot, however, yield an acceptable and reliable connectivity estimate over an observation interval. Additionally, the acquired connectivity does not yield simple rationalization following an experiment with a specific stimulus. To that end, and for additional interpretational potential, graph structures that are learned from data have been of great research interest. Research in this direction has pervasively been based on graphs' topology and signals' smoothness, as well as the application of the graph Laplacian. Other recent work includes deep neural network modeling, for which training/testing was performed on graph datasets to generate a graph, representing patterns under given signals ultimately. These studies have primarily focused on static graphs with non-sparse signals, as well as on the consistency assumption of the graphs over time. These models require sufficiently adequate samples for training and testing, once again making it difficult to use neuronal signals whose sample size is typically small for adequately assessing variations over rather short time intervals. Graphs' dynamics with a potential impact on temporal data characterization have also been of much interest to [12,13]. In this theme, the models are used to predict the links given previous graphs and associated signals. All these approaches require much prior information on known structures and sufficient data for training and predicting future graphs. Moreover, game-theoretic techniques, which have been proposed to address multi-objective optimization problems [14], may be seen as potentially useful in our use case by interpreting our interacting neurons as a multi-player game. It is, however, difficult to achieve a single comprehensive model with a capacity to capture the neurons' large variability and time-varying characteristics.

To analyze the characteristics and representation of a graph, topological data analysis has been shown to measure the shape or connectome of a graph [15]. In addition to investigating pairwise connections between every pair of nodes, higher-order information, using simplices and homology, was proposed to determine the prevailing homology, thus affording an interpretation of a topological hole/cycle as the inefficiency of (or slow down) information diffusion [16]. While this approach is practical when analyzing a given graph, it is more challenging to account for differential constraints to optimize the construction of a graph associated with distributed data.

In this work, we build on the wealth of prior research work in neuroscience as well as in graph learning [11], to propose a new model, with a target dynamic structure to track neurons' temporal connectivity, as well as higher-order connectivity (three neurons connected sets), referred to as 2-simplex information. To that end, our proposed dynamic graph learning will include vertices/nodes with their connectivity reflected by graph edges. An edge attribute is defined as the probability of connection between a pair of neurons, and that of a 2-simplex to reflect the connectivity degree of pairwise connection for all three neurons in a 2-simplex.

To proceed, the outline of our paper follows, in order, our contributions in the sequel. Firstly, exploiting the insight from prior research on graph learning with graph Laplacian [11,17], we propose a formulation to estimate an optimal graph short time intervals, which in turn reflects the evolving transformation of the connectivity. Secondly, we modify our model to fit sparse signals to verify our optimized solution on a neuronal signal dataset. Thirdly, we call on a tensor to model connectivity tracking among three neurons to account for higher-order information. Fourthly, we propose three alternative methods to simplify the optimization procedures so that the optimization problems' solution procedures are simplified and help reach the optimal points. We finally proceed to test our proposed model using a neuronal dataset, to improve our understanding of the neuronal interaction and their process of signal/information propagation.

2. Problem Setup and Background

Throughout the paper, we will adopt an upper and lower case bold letter to respectively denote a matrix and a vector, and the superscripts $T, -1$ to respectively denote a matrix transpose and inverse. The operator $tr(\cdot)$ denotes a matrix trace. The identity, zero and "1" matrices are respectively denoted by $\mathbf{I}, \mathbf{0}$ and $\mathbf{1}$, while x_{ij} represents the i-th row, j-th column element of a matrix $\mathbf{X}$.

2.1. Definition of Graph and Graph Laplacian

Our neuronal-activity dataset will be N-dimensional, and will be characterized by a connectivity graph $G = \{V, E, W\}$, where V denotes the vertex set $V = \{v_1, v_2, \ldots, v_N\}$, E is the edge set with attributes reflecting the connectivity between each pair of vertices quantified by $0 \leq w \leq 1, \forall w \in W$ as a connectivity strength. A time series $y_n(t)$ of observations with $t = 1, 2, \ldots, T_{all}$, is associated with each node v_n. For simplicity, we will aggregate the nodes' finite-length time series into a $N \times T_{all}$ matrix $\mathbf{Y}$, where $(\mathbf{Y})_{t,n} = y_n(t)$. Our problem formulation will seek for each observed $\mathbf{Y}$, either a static graph G or a time dependent graph series of graphs $G_1, G_2, \ldots$.

The well-known graph Laplacian of an undirected graph can help describe its topology, and can serve as the second derivative operator of the underlying graph. The corresponding Laplacian matrix $\mathcal{L}$ is commonly defined as [18], with $\mathcal{L}(i, j) = -w_{ij}$, for i, j adjacent nodes, and $\mathcal{L}(i, j) = 0$ otherwise, and $\mathcal{L}(i, i) = d_i$, where $d_i = \sum_j w_{ij}$ denotes the degree of node i. Its simple matrix expression is $\mathcal{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is a diagonal matrix of degrees.

The Laplacian matrix may also usefully adopt, in some context, a second derivative interpretation of graphs: Given an assignment $\mathbf{x} = (x_1, x_2, \ldots, x_N)^T$ of real numbers to the nodes, the Laplacian matrix may be found as the second derivative of $\mathbf{x}$ as $\mathcal{L}(\mathbf{W}) = \sum_i \sum_{j>i} w_{ij} \mathbf{a}_{ij} \mathbf{a}_{ij}^T$, where $\mathbf{a}_{ij}$ denotes a N-dimensional vector whose elements are all 0s, except the i-th element being respectively 1 and j-th element -1. As may be seen, a_{ij} represents the first derivative of the edge between the i-th and j-th node. The notion of a Laplacian will be exploited in this sequel for a given data set as a structural regularizer when an optimal graph is sought.

2.2. Topological Basics

Euler characteristic originally proposed for polyhedron is also invoked in the homology theory, with a motivation of comparing two shapes by taking into account any dimensional holes proper to their topology. For instance, the difference between a disk and a circle in homology is that a one-dimensional hole exists in a circle. In this sense, homology theory provides an alternative definition of shapes in topological spaces by decomposing a shape S into a family of abelian groups $H_0(S), \ldots, H_n(S)$ [19].

In geometry, a simplex is a general concept of a fully connected shape in any dimensions, where a k-simplex is a k-dimensional polyhedron of $k + 1$ vertices. As an example, a disk can be simplified and assimilated to a 0-simplex, while a circle is distinguished as a chain complex with a one-dimensional hole. Two different strategies are used to construct simplex-based structures for computational purposes in homological data analysis. In a so-called Vietoris-Rips complex, two nodes are said to be connected to each other if their neighborhoods (balls around the node) overlap, thus making the polytope a simplex. In a Cech complex, a simplex is, on the other hand, defined if and only if any given node is connected to any other nodes with overlapping of all neighbors. Therefore, a simplex in a Cech complex is a stricter connection in any dimension. These simplices are illustrated in Figure 1.

In our neuronal-activity dataset, we translate the connectivity among three neurons into 2-simplex with the criteria in Cech complex with a mathematical equation, and the result gives higher order connection information beyond pairwise connectivity.
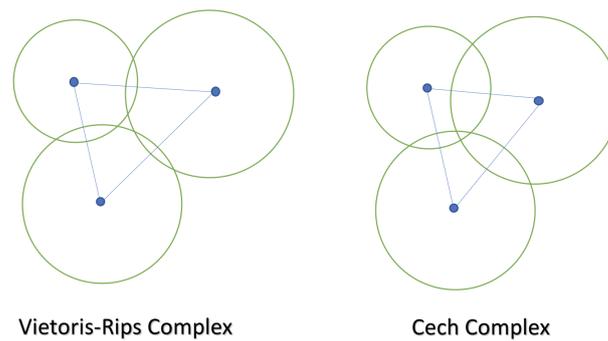
Vietoris-Rips Complex   Cech Complex

Figure 1. Vietoris-Rips Complex and Cech Complex.

3. Dynamic Graph Learning

3.1. Static Graph Learning

Prior to proposing the dynamic structure learning of a graph, we briefly revisit the basic notions of static graph learning [11]. Using the Laplacian quadratic form $tr(\mathbf{X}^T \mathcal{L}(\mathbf{W})\mathbf{X})$ as a smoothness regularizer of the data $\mathbf{x}_n$, and the degree of connectivity K as a tuning parameter, ref. [11] discovers a K-sparse graph from noisy signals $\mathbf{Y}$. This is the result of solving the following,

$$\underset{\mathbf{X},\mathbf{W}}{\text{argmin}} \, \|\mathbf{Y} - \mathbf{X}\|^2 + tr(\gamma \mathbf{X}^T \mathcal{L}(\mathbf{W})\mathbf{X})$$
$$s.t. \quad 0 \leq w_{ij} \leq 1, \quad \forall i, j, \quad \sum_{i,j>i} w_{ij} = K, \tag{1}$$

where γ and K are tuning parameters, $\mathbf{X}$ is the noiseless signals, and $\mathbf{Y}$ its noisy observation. $\mathbf{W}$ is the adjacency weight matrix for the undirected graph, with the additional relaxation of the individual weights to the interval $[0, 1]$.

3.2. Dynamic Graph Learning

Please note that in [11], a single connectivity graph is inferred over the entire observation time interval, thus overlooking the practically varying connections between every pair of nodes over time. To account for these events evolving over short term intervals and hence capture the true underlying structure of the graph, we propose to learn the graph's dynamics. Learning these dynamics is consistent with our goal of modeling the brain signals to elicit timely information. These would particularly aim to pick up the responses to corresponding stimuli in that time interval and account for their dependence on previous graph instances and time intervals. This also introduces a practical constraint which needs to be accounted for by way of the similarity of temporally adjacent graphs in the overall functionality of the sequence of graphs in congruence with the observed data by selecting a 1-norm distance of connectivity weight matrices between consecutive time intervals, we can proceed with the graph sequence discovery, and hence the dynamics by seeking the solution to the following,

$$\underset{\mathbf{X}_t,\mathbf{W}_t}{\text{argmin}} \sum_{t=1}^{T} \left[\|\mathbf{Y}_t - \mathbf{X}_t\|^2 + tr(\gamma \mathbf{X}_t^T \mathcal{L}(\mathbf{W}_t)\mathbf{X}_t) \right]$$
$$+ \alpha \sum_{t=1}^{T-1} \|\mathbf{W}_t - \mathbf{W}_{t+1}\|_1 \tag{2}$$
$$s.t. \quad 0 \leq w_{t,ij} \leq 1, \quad \forall i, j, \quad \sum_{i,j>i} w_{t,ij} = K,$$

where α is a penalty coefficient, $\mathbf{Y} = [\mathbf{Y}_1, \ldots, \mathbf{Y}_T]$, $\mathbf{Y}_t$ is the observed data in the t-th time interval, $\mathbf{X}_t$ is the corresponding noise-free data, $\mathbf{W}_t$ is the weight matrix in the t-th time interval, and K is a tuning parameter.

3.3. Dynamic Graph Learning from Sparse Signals

The solution given by [11] provides the static graph learning, but the observed signals $\mathbf{Y}$ may often be sparse, which poses a problem: noting that $tr(\mathbf{X}^T \mathcal{L}(\mathbf{W})\mathbf{X}) = tr\left(\sum_i \sum_{j>i} w_{ij} \mathbf{X}^T \mathbf{a}_{ij} \mathbf{a}_{ij}^T \mathbf{X}\right)$ is the Laplacian quadratic form, we use $w_{ij}\|\mathbf{x}_i - \mathbf{x}_j\|^2$ to reflect the distance between two signals, whose minimization will unveil some problematic nodes with very similar signals over a small time interval.

Technical and Practical Constraints

While the above formulation (2) captures the principles of the dynamics of interest, additional practical difficulties arise in the case when $\mathbf{x}_i$ and $\mathbf{x}_j$ are close to 0, making their distance close to 0. This, as a result, introduces unexpected false edges when sparse signals are present. Such an instance may arise for, say, given sparse signals written as $\mathbf{Y} = [\tilde{\mathbf{Y}}, \mathbf{0}]^T$, where the dimension of $\mathbf{Y}$ is $N \times t$, $\tilde{\mathbf{Y}}$ is an $n \times t$ matrix and $\mathbf{0}$ is $(n - N) \times t$. Given that 2-norm is non-negative and the Laplacian matrix is positive semi-definite, we can find a trivial optimal solution of $(\mathbf{X}, \mathbf{W})$, where $\mathbf{W}$ is sparse, such that $\mathbf{X} = \mathbf{Y}$, and the weight matrix can be represented by some block matrix, $\mathbf{W} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{W}} \end{bmatrix}$.

Given that our given graph learning problem is a convex and non-negative problem, one can show that the optimal loss value is 0 by inserting the solution $\mathbf{Y} = \mathbf{X}$ and $\mathbf{W}$. This indicates that the presence of sparse signals (which happens to be the case for brain firing patterns) results in a solution of the formulated optimization that may not be unique. Additionally, this leads to optimal points that establish connectivity between zeros-signal nodes (i.e., erroneous and perhaps meaningless connections).

Proposition 1.

$$L_{DGL}(\mathbf{X}_t, \mathbf{W}_t) = \sum_{t=1}^{T} \left[\|\mathbf{Y}_t - \mathbf{X}_t\|^2 + tr\left(\gamma \mathbf{X}_t^T \mathcal{L}(\mathbf{W}_t)\mathbf{X}_t\right) \right.$$

$$\left. - 2\eta \sum_{i,j>i} w_{ij}(\|\mathbf{x}_{t,i}\|^2 + \|\mathbf{x}_{t,j}\|^2) \right] + \alpha \sum_{t=1}^{T-1} \|\mathbf{W}_t - \mathbf{W}_{t+1}\|_1 \qquad (3)$$

$$s.t. \quad 0 \le w_{ij} \le 1, \quad \forall i,j, \quad \sum_{i,j>i} w_{ij} = K,$$

where η is a penalty coefficient, $\mathbf{x}_{t,i} = (x_{1,i}, \ldots, x_{t,i})^T$ is the i-th node's signal in the t-th time interval, and $\mathbf{X}_t = [\mathbf{x}_{t,1}, \mathbf{x}_{t,2}, \ldots, \mathbf{x}_{t,N}]^T$.

Proving this proposition is tantamount to lifting the fore-noted difficulty of sparse signals. To that end, we introduce a constraint term to help focus on the nodes with significant values, specifically we constrain edge nodes energy to be of relevance.

Since the weight matrix for an undirected graph is symmetric, therefore this additional part of the new optimization can be simplified as follows:

$$\mathcal{P} = -2\eta \sum_{i,j>i} w_{t,ij}(\|\mathbf{x}_{t,i}\|^2 + \|\mathbf{x}_{t,j}\|^2)$$

$$= -tr(\mathbf{X}_t^T \eta \mathcal{D}(\mathbf{W}_t)\mathbf{X}_t) \qquad (4)$$

where $\mathcal{D}(\mathbf{W}_t)$ is a diagonal matrix defined above from weight matrix $\mathbf{W}_t$. Combining the two $tr(\cdot)$ expressions from Equations (3) and (4) will yield the simpler form. With a little more attention, one could note that this procedure naturally prefers nodes of higher energy by associating a higher weight.

3.4. Dynamic Graph Learning with Higher-Order Information

The quadratic form of $\mathbf{X}_t$ over $\mathbf{W}_t$ in optimization model (3) accounts for pairwise connections between nodes. To exploit higher order than pairwise interaction among nodes requires a topology-driven information structure across nodes in a graph. To that end, and to bypass the computational complexity of homology assessment of the topological complex resulting from a set of nodes, we propose a practically computable alternative in the optimization model which quantifiably reflects the connectivity among three neurons, thus capturing the inherent information within a 2-simplex.

Proposition 2.

$$
\begin{aligned}
L_{DGLH}(\mathbf{X}_t, \mathbf{W}_t, \mathbf{W}_t^{(3d)}) = & \sum_{t=1}^{T} \Big[\|\mathbf{Y}_t - \mathbf{X}_t\|^2 \\
& + tr(\gamma \mathbf{X}_t^T \mathcal{L}(\mathbf{W}_t)\mathbf{X}_t) - 2\eta_1 \sum_{i,j>i} w_{ij}(\|\mathbf{x}_{t,i}\|^2 + \|\mathbf{x}_{t,j}\|^2) \Big] \\
& + \alpha \sum_{t=1}^{T-1} \|\mathbf{W}_t - \mathbf{W}_{t+1}\|_1 + \tau \sum_{i,j,k} w_{t,ijk}^{(3d)} \Big(\|\mathbf{x}_{t,i} - \mathbf{x}_{t,j}\|^2 \\
& + \|\mathbf{x}_{t,i} - \mathbf{x}_{t,k}\|^2 + \|\mathbf{x}_{t,j} - \mathbf{x}_{t,k}\|^2 \Big) \\
& - \eta_2 \sum_{i,j,k} w_{t,ijk}^{(3d)}(\|\mathbf{x}_{t,i}\|^2 + \|\mathbf{x}_{t,j}\|^2 + \|\mathbf{x}_{t,k}\|^2), \\
s.t. \quad & 0 \le w_{t,ij} \le 1, \quad \forall i,j, \quad \sum_{i,j>i} w_{t,ij} = K_1, \\
& 0 \le w_{t,ijk}^{(3d)} \le 1, \quad \forall i,j, \quad \sum_{i,j>i} w_{t,ijk}^{(3d)} = K_2.
\end{aligned}
\tag{5}
$$

In the basic Vietoris-Rips complex, if three given neurons are connected to each other, we will consider these three neurons as a 2-simplex. We hence propose the following strategy for considering all connections among three neurons in the t-th time interval: $w_{t,ijk}^{(3d)}[\|\mathbf{x}_{t,i} - \mathbf{x}_{t,j}\|^2 + \|\mathbf{x}_{t,i} - \mathbf{x}_{t,k}\|^2 + \|\mathbf{x}_{t,j} - \mathbf{x}_{t,k}\|^2]$. The formulation is in fact a direct result of the simplex definition in a VR-complex, making the proximity between every pair of signals in a triplet of nodes, a 2-simplex is generated. In other words, we regard this 2-simplex information as a tight connection among these three neurons.

As with the development for pair-wise node interaction, we proceed to tackle the 3-way node interaction by introducing a weight attribute tensor $\mathbf{W}_t^{(3d)} = [w_{t,ijk}^{(3d)}]$, when recalling the non-negative characteristic of $\|\mathbf{x}_{t,i} - \mathbf{x}_{t,j}\|^2 + \|\mathbf{x}_{t,i} - \mathbf{x}_{t,k}\|^2 + \|\mathbf{x}_{t,j} - \mathbf{x}_{t,k}\|^2$. Our inclusion of a constraint $\sum_{i,j,k} w_{t,ijk}^{(3d)} = K_2$ to force to construct the tightest higher order connections may hence be achieved. As with the afore-discussed issue about sparse signals yielding non-realistic connections among three weak/zero node signals, we proceed to constrain such a solution out by adding a energy penalty term at each node: $-w_{t,ijk}^{(3d)}(\|\mathbf{x}_{t,i}\|^2 + \|\mathbf{x}_{t,j}\|^2 + \|\mathbf{x}_{t,k}\|^2)$. By accounting for all these factors, we obtain Proposition 2 which considers both the pairwise connection and a 2-simplex connection/information.

4. Algorithm Numerical Solution

Solving Propositions 1 and 2 by the conventional Lagrangian duality is computationally complex, and an alternative is in order. To that end, we propose a coupled pair of computational steps to solve Equation's optimization (5). The first facilitates solving the optimization by the periodic updates of the proximal operator as detailed, and the other seeks to relax graph adjacency weight terms to be continuous on $[0, 1]$ below.

4.1. Proximal Operator

$\mathbf{W}_t$ are updated with gradient descent method. On account of the non-smoothness of the l_1 norm in Equation (5), we call on the proximal operator to first rewrite model (3) and proceed to solve it [20,21]. Firstly, the l_1 term $\|\mathbf{W}_t - \mathbf{W}_{t+1}\|_1$ in optimization (3) may be affected by the order of updating $\mathbf{W}_t$s. To reduce the impact of the order of updating variables, we introduce an auxiliary variable $\mathbf{Z}_t$, with the constraint that of the order of updating variables, we introduce $\mathbf{Z}_t$ to replace the this term, and add a new constraint that $\mathbf{Z}_t = \mathbf{W}_t - \mathbf{W}_{t+1}$. This substitution allows an update of $\mathbf{W}_t$ with no influence by the others weight matrices, and $\mathbf{Z}_t$ provides the relaxation between each pair of adjacency weight matrices. This results in an equivalence to the previous optimization problem, with the advantage a reduced impact on the optimization caused by the variables updating order. We introduce β_t as the Lagrange multiplier of the equality constraints $\mathbf{Z}_t - \mathbf{W}_t + \mathbf{W}_{t+1} = 0$.

Claim: As a result, the Lagrangian duality form of the optimization can hence be re-expressed as,

$$
\begin{aligned}
L(\mathbf{W}_t, \mathbf{X}_t, \mathbf{Z}_t, \gamma, \eta, \alpha, \beta_t) =& \sum_{t=1}^{T} \|\mathbf{Y}_t - \mathbf{X}_t\|^2 \\
&+ tr(\mathbf{X}_t^T(\gamma \mathcal{L}(\mathbf{W}_t) - \eta \mathcal{D}(\mathbf{W}_t))\mathbf{X}_t) + \alpha \sum_{t=1}^{T-1} \|\mathbf{Z}_t\|_1 \\
&+ \langle \beta_t, \mathbf{Z}_t - \mathbf{W}_t + \mathbf{W}_{t+1} \rangle.
\end{aligned}
\tag{6}
$$

We now have a function of $\mathbf{Z}_t$, denoted as $f(\mathbf{Z}_t) = \alpha\|\mathbf{Z}_t\|_1 + \langle \beta_t, \mathbf{Z}_t \rangle$, which is a convex and non-smooth function over $\mathbf{Z}_t$. to alleviate the numerical difficulty of the non-differentiable contribution to Equation (6), we adopt the proximal operator approach to search for an optimal Z_t. The function is defined as $\mathbf{prox}_{\lambda f}(\mathbf{V}_t) = \mathrm{argmin}_{\mathbf{Z}_t} f(\mathbf{Z}_t) + \frac{1}{2\lambda}\|\mathbf{Z}_t - \mathbf{V}_t\|_2^2$, where λ is a tuning parameter, which can be interpreted as the gradient step in the proximal algorithm. It is clear that we achieve the optimal point $\mathbf{Z}_t^*$, if and only if we have $\mathbf{Z}_t^* = \mathbf{prox}_{\lambda f}(\mathbf{Z}_t^*)$, therefore for the *k-th* iteration, we update the variable $\mathbf{Z}_t$ as $\mathbf{Z}_t^k = \mathbf{prox}_{\lambda f}(\mathbf{Z}_t^{(K-1)})$.

4.2. Projection Method

To address this issue, we associate a subspace structure W to the set of constraints, where $\mathcal{W}$ is the whole weight space for graphs, with $w_{ij} \geq 0$, and $W \subset \mathcal{W}$, such that $0 \leq w_{ij} \leq 1$. We next construct a projection procedure of $\mathbf{W}_t \in \mathcal{W}$ onto W to ensure a subspace membership of the next iterate. Considering an updated weight matrix as a point in a high dimensional space, we minimize its distance to the subspace within the whole space by enforcing $\min_{\tilde{W}_t} \frac{1}{2}\sum_{i,j>i}(\tilde{w}_{t,ij} - w_{t,ij})^2$, $s.t. \sum_{i,j>i}\tilde{w}_{t,ij} = K$ and $\tilde{\mathbf{W}}_t \in W$. A similar membership-set projection can equally be applied to $W_t^{(3d)}$ by vectorizing the associated tensor. Applying the Lagrangian Duality on this minimization problem yields,

$$
\begin{aligned}
L_{Proj}(\tilde{\mathbf{W}}_t, \kappa) =& \frac{1}{2}\sum_{i,j>i}(\tilde{w}_{t,ij} - w_{t,ij})^2 \\
&+ \kappa(\sum_{i,j>i}\tilde{w}_{t,ij} - K_1), \\
& s.t. \; \tilde{\mathbf{W}}_t \subset \mathcal{W}. \\
L_{Proj}(\tilde{\mathbf{W}}_t^{(3d)}, \kappa') =& \frac{1}{2}\sum_{i,j,k}(\tilde{w}_{t,ijk} - w_{t,ijk})^2 \\
&+ \kappa'(\sum_{i,j,k}\tilde{w}_{t,ijk} - K_2), \\
& s.t. \; \tilde{\mathbf{W}}_t^{(3d)} \subset \mathcal{W}^{(3d)}.
\end{aligned}
\tag{7}
$$

4.3. Algorithm

With the $\mathbf{Z}_t$ update in hand, we proceed to unfold the various steps of the algorithmic solution of model (5). The functional dependent on $\mathbf{X}_t$ being convex smooth allows the calculation of the differential of the optimization formulation over $\mathbf{X}_t$ and and yielding the following iteration.

$$\mathbf{X}_t^{(k)} = \left(\mathbf{I} + \gamma\mathcal{L}(\mathbf{W}_t^{(k-1)}) - \eta\mathbf{D}(\mathbf{W}_t^{(k-1)})\right)^{-1}\mathbf{Y}_t. \tag{8}$$

Since the functional depending on $\mathbf{W}_t$s are smooth, we use gradient descent to update each $\mathbf{W}_t$. The whole algorithm is presented in Algorithm 1.

Algorithm 1: algorithm for dynamic graph learning

    Input   : $\mathbf{Y}_t$
    Output: $\mathbf{X}_t, \mathbf{W}_t$

1   $\alpha, \gamma, \eta, \lambda$ and learning rate τ are pre-defined.
2   **while** *not converged* **do**
3   Update $\mathbf{X}_t$ by (8)
4      $\mathbf{W}_t^{(k)} = \mathbf{W}_t^{(k-1)} - \tau_1\frac{\partial}{\partial W_t}L_{DGLH}(\mathbf{W}_t, \mathbf{X}_t, \mathbf{Z}_t, \gamma, \eta, \alpha, \beta_t)\big|_{\mathbf{W}_t^{(k-1)}}$
5   Project $\mathbf{W}_t^{(k)}$ to the defined domain by Projection method.
6   Update $\mathbf{W}_t^{(3d)}$.
7   Project $\mathbf{W}_t^{(3d)}$ to the defined domain by Projection method.
8   Update $\mathcal{L}(\mathbf{W}_t^{(k-1)})$ and $\mathbf{D}(\mathbf{W}_t^{(k-1)})$ by definition.
9   Update $\mathbf{Z}_t$ by Proximal operator.
10     $\beta_t^{(k)} = \beta_t^{(k-1)} - \tau_2\frac{\partial}{\partial\beta_t}L_{DGLH}(\mathbf{W}_t, \mathbf{X}_t, \mathbf{Z}_t, \gamma, \eta, \alpha, \beta_t)\big|_{\beta_t^{(k-1)}}$
11   **end**

---

5. Experiments and Results

The following experiments on neural activity data also provide, as a result, a new analysis and a potential new exploratory tool in neuroscience and machine learning. The neuronal activity data was provided by Dr. S. L. Smith's Laboratory at UCSB.

5.1. Computational Complexity

The computational complexity is defined as the execution time of our program which is, in turn, dependent on the number of nodes N as well as on the number of time intervals T. We fix all other parameters and vary the choice of N and T, respectively, to discuss our model's complexity. The results are shown in Figure 2.

5.2. Synthetic Data Generation

Modeling and analyzing experimental/real data of neural activity with inference objectives were the primary motivation of this work. Our goal of thoroughly evaluating our model of neural activity is unfortunately met with limited amounts of real data available for validation, as noted, which led us to generate a set of synthetic signals from pre-defined connections based on biological criteria. This affords us a characterization and retrieval of all associated structural features to compare our modeled-based results with graph representations obtained using Pearson correlation-based edge connection. This is widely used in neuroscience to evaluate neuronal connectivity of observed signals.
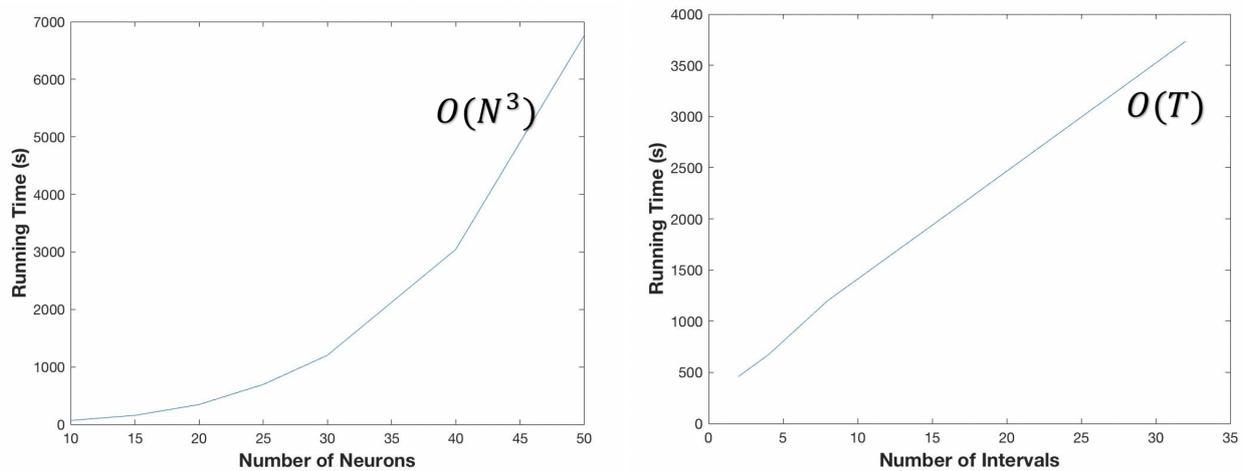
(**a**) Execution time vs. Number of Neurons

(**b**) Execution Time vs. Number of intervals

Figure 2. Computational Complexity Test.

5.2.1. Synthetic Data Model

Following computational neuroscience guidelines [22], we postulate the following model to generate signals mirroring the relevant dynamics,

$$y_i(t) = p(x_i(t)I(x_i)) + n_i(t), \tag{9}$$

where $y_i(t)$ is the measured spiking signal of i-th neuron at time t, $x_i(t)$ represents the number of supposed arrival spikes generated from Poisson Distribution, $I(x_i)$ is an indicator function, which indicates the i-th neuron firing/not, p function is a probabilistic process for determining each spike firing/not, and $n_i(t)$ represents a measurement noise.

The number of arrival spikes is described by a probabilistic model, where the probability of a firing spike in an interval $(t, t + \Delta t]$ is in direct proportion to Δt [22],

$$P(s = 1|(x, x + \Delta t]) = \lambda \Delta t, \tag{10}$$

λ is the firing rate for generating the number of arriving spikes, hence, parameterizing a Poisson distribution λ.

The indicator function $I(\cdot)$ supports the non-uniform firing of neurons due to the same stimulus. The fact that only a subset of the neurons in a given region are active at each trial, indicates that the proportion of event-related neurons should be a tuning parameter.

p function imitates the idea of the perceptron model in the visual cortex, making the output the result of a weighted summation of neurons' contributions in the previous layer, i.e., $y = \sigma(w_1 x_1 + \cdots + w_n x_n + w_0)$, with w_i describing the degree of influence of each neuron, w_0 as some bias, and $\sigma(\cdot)$ representing an activation function. In the proposed model, the probability of each neuron i firing/not is dependent on its neighborhood's neurons signals/contributions and computed as follows: $P(s_i = 1|graph) = P(s_i = 1|N(i)) = P(s_i = 1|x_1, x_2, \ldots) = \sigma(w_0 + w_1 x_1 + w_2 x_2 + \ldots)$, where s_i denotes i^{th} neuron spike, $N(i)$ its ith neighborhood. The combined model accounts for the probability of spike firing and the Poisson nature of the generated spikes, together with the Bernoulli process of a turn on, yields under a connectivity schedule, some generated information.

The measurement noise $n_i(t)$, assumed white and Gaussian (μ, σ^2), was experimentally determined to best fit the generative process.

5.2.2. Synthetic Data

To first evaluate the viability of the proposed model, we qualitatively show the behavior of the model's respective densities and of the real data in Figure 3. We also show the degree of correlation between the neuronal data for various trials and the degree of the

correlation between different neurons in the same trial. The original intensities, X-axis, are all normalized between 0 and 1 using $\frac{sig-min}{max-min}$, Y-axis presents the number of times that the values occurred within the intervals set by the X-axis. The results are shown in Figure 3.
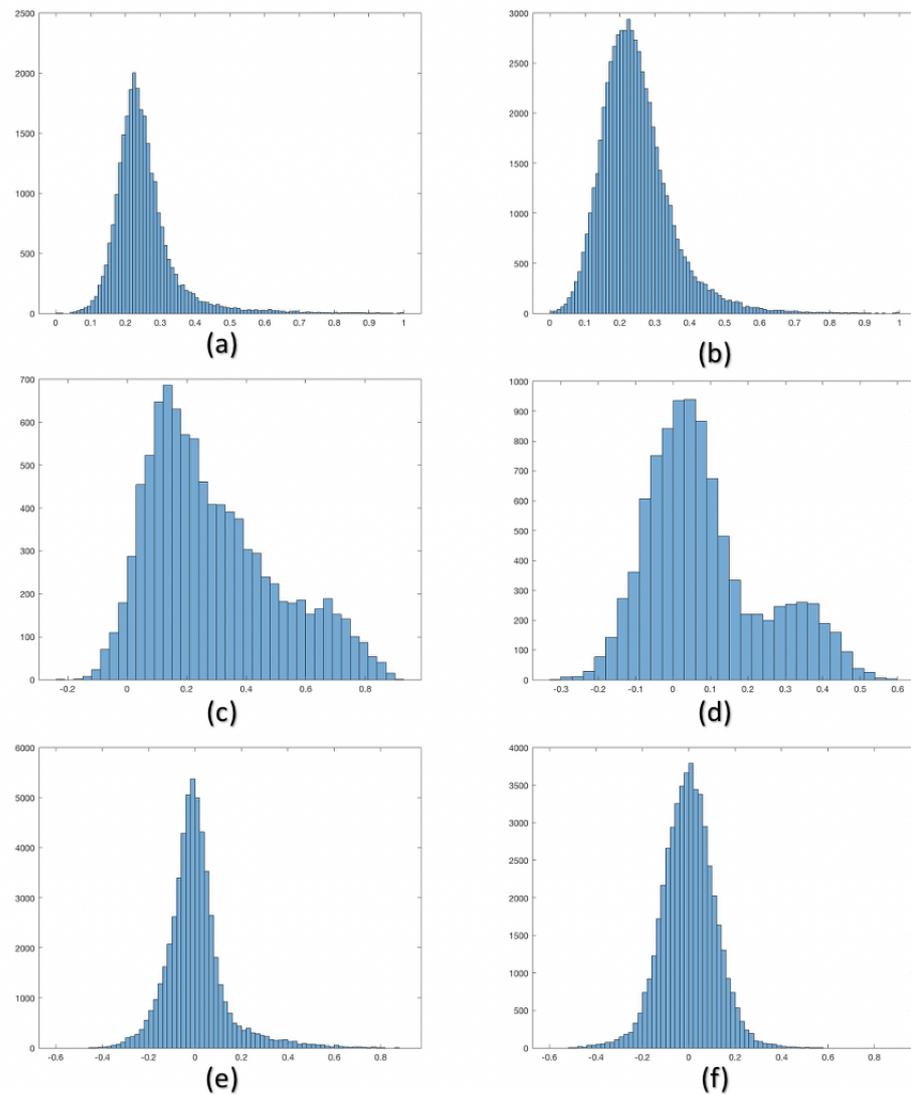

Figure 3. Distribution comparison: (**a**) The distribution of data intensity of neuronal activity data. (**b**) The distribution of data intensity of synthetic data. (**c**) Distribution of trial-to-trial variability of neuronal data. (**d**) Distribution of trial-to-trial variability of synthetic data. (**e**) Distribution of correlations between neurons of neuronal data. (**f**) Distribution of correlations between neurons of synthetic data.

5.2.3. Simulated Connectivity Graph

To evaluate the effectiveness of generating neuronal networks by merely using the Pearson correlation, we manually subdivide the signals into 3 intervals. We use the same constraint parameter K, where $\sum_{i,j} w_{t,ij} = K$ for our model. At the same time, we select K largest correlation values after calculating the Pearson correlation between every two nodes' signals. Upon applying the same process on N synthetic sets of signals, we obtain the graph representation sets $\mathbf{W}_t^i$ for our model, and $\tilde{\mathbf{W}}_t^i$ using Pearson correlation, t representing t-th time interval, and i denoting i-th trial. By calculating $\mathbf{W}_t = \sum_i \mathbf{W}_t^i$ and $\tilde{\mathbf{W}}_t = \sum_i \tilde{\mathbf{W}}_t^i$, we determine the consistency of edge connectivity over trials by defining a threshold th_w to count those edges repeating more than th_w times.

To get an additional assessment of our activity network generation model against that obtained by Pearson correlation, we use the receiver operating characteristic (ROC) curves and further illustrate the two methods' capabilities. We use 100 synthetic trials and record connections in each trial over each time interval. By choosing different threshold th_w, we select N consistent edges and then count the number of correct edges n based on the ground truth, which is set to 30 in our synthetic data experiment. The total number of edges of an undirected graph among 50 nodes is 1225. Therefore, the ROC curve is true positive rate, $\frac{n}{30}$, vs. false negative rate, $\frac{N-n}{1225-30}$.

From Figure 4, the ROC curve for graphs using the Pearson correlation shows that it is close to a random-select model. In other words, the Pearson correlation can hardly recover the correct connectivity among all nodes. Compared with the Pearson correlation method, our model shows a significantly better performance in every state. Even with limited synthetic data trials, our model can also restore the essential connectivity (ground truth) accurately. Our model recovers the essential graphs for each state with 10 generated trials, by choosing those edges repeating more than 5 times. As shown in Table 1, the correct number of edges number in each state is 30, the number of selected edges from our model for each state is shown in the third column, and the number of correct edges for each state is shown in the last column. Those edges with a high repetition rate infer the essential structure of the underlying connectivity.
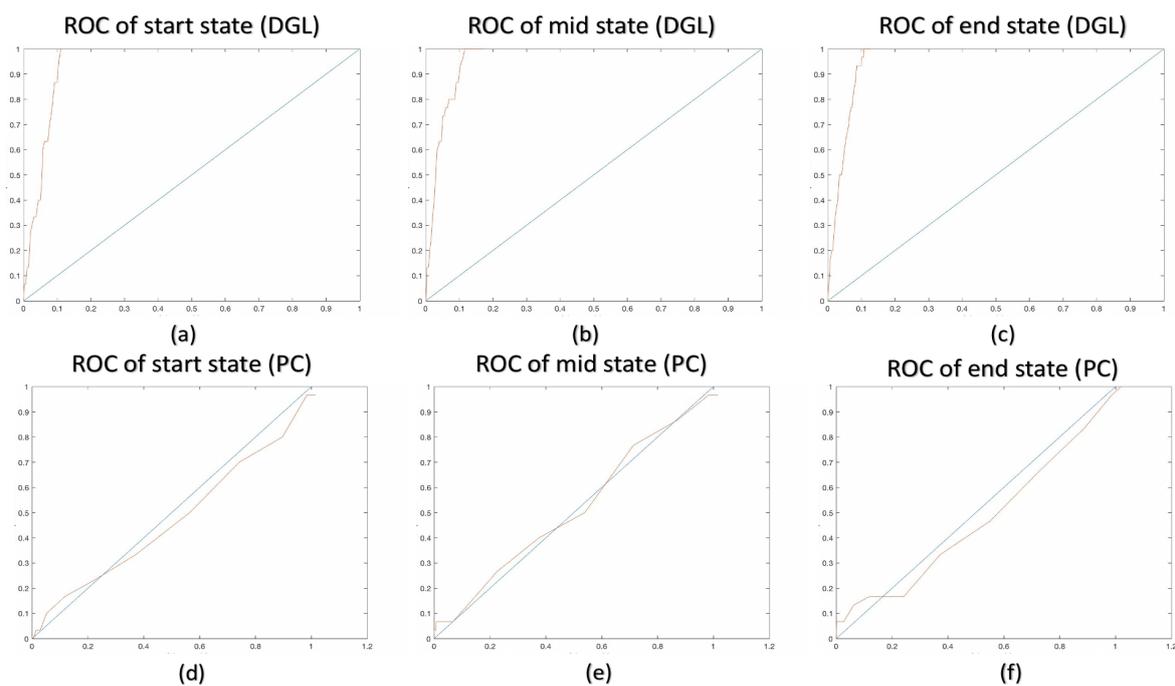

Figure 4. ROC curve: (**a**–**c**): ROC curve for our model (DGL) in different states. (**d**–**f**): ROC curve for using the Pearson correlation (PC) in different states.

Table 1. Synthetic data results.

State	Ground Truth	# Connectivity Selection in Our Model	# Correct Edges
Start state	30	66	20
Mid state	30	59	18
End state	30	57	21

5.3. Neuronal-Activity Data

Neurons, as the brain's basic active elements, receive and transmit information to a center and one another (typically the pre-frontal cortex in the primates). Information collected by sensory organs, such as visual information, olfactory, etc., is encoded into

neural signals, which are membrane potentials, which are, in turn, propagated in the brain for decision making. The widely accepted model distinguishes the brain's visual sensorial part as the visual cortex. The visual cortex [23,24], a widely studied part of the brain for its importance and relevance here, is often used to explore the cause-effect of visual stimuli. The visual cortex study may comprise several layers, the first being the so-called V1 area, known for its directional sensitivity. The two cardinal features of neurons in the V1 area are driven by 2 types, orientation selectivity and direction selectivity [25], where orientation stands for the stimulus angle outline, and direction stands for the motion tendency. Most neurons in the V1 area are orientation-selective, which means that they are highly active under a specific angle of the stimulus for both directions. Some neurons are direction-selective, where they are sensitive to a specific angle and moving direction of a stimulus. As a result, models are constructed to explain motion detection in the primary visual cortex [26,27], which suggest that a few simple cells with quadrature phases and directions jointly define and transfer the information of motion direction to the higher-level visual cortex.

5.3.1. Real Experimental Data

The data were collected in S. L. Smith's Laboratory at UCSB [8]. A two-photon electro-microscope [28] was used to collect fluorescent signals. The whole experiment consists of 3 specific scenarios with a 20 trial measurement, with the same stimuli in each trial. The stimuli for each of the scenarios are shown in Figure 5, consisting of a "gray" movie, an artificial movie, and natural movie 1 and movie 2 (which we will not discuss here and defer to future work). The dataset includes 590 neurons in the V1 area, and the sample rate is approximately 13.3 Hz. In the artificial movie data we have used in our experiments, each stimulus is characterized by an orientation and pattern motion direction. This is adapted to be detectable and trackable in the brain V1 area. We qualify two stimuli as similar if they share both orientation and direction of pattern motion. The stimuli are deemed related if they share the same orientation but differ in the pattern motion direction.
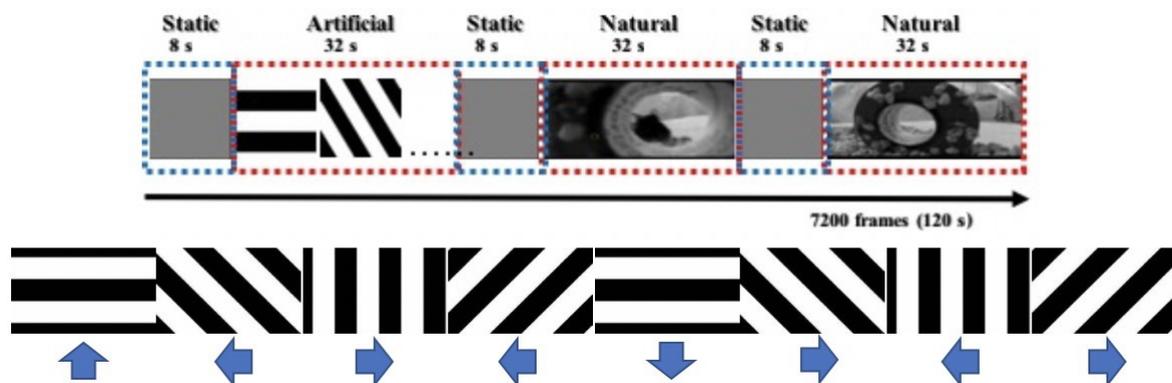

Figure 5. Visual Stimuli: The visual stimuli for the mouse in a single trial.

5.3.2. Pre-Processing

The presumed and accepted neuronal redundancy in the brain is equivalent to observing a diverse neuronal activity pattern for the exact same stimulus. As is also known, additive noise causes unexpected firings of neurons. Moreover, much noise, such as unexpected firing, is contained in signals. To better cope with such variability and improve our analysis quality, we select the 50 most active and consistent neurons from the 590 neurons in the V1 area. Specifically, the selection is based on the highest correlation of each neuron activity across trials. We consider these neurons as the most consistently functioning neurons due to the same stimulus over trials. Neuronal signals stimulated by the artificial movie in two different trials are presented in Figure 6a, where X-axis reflects time, Y-axis indexes the 50 neurons, while the brightness reflects the normalized data intensities. Figure 6b depicts

the 50 selected neurons' relative positions, where numbering reflects the consistency (i.e., #1 is the most consistently active neuron).
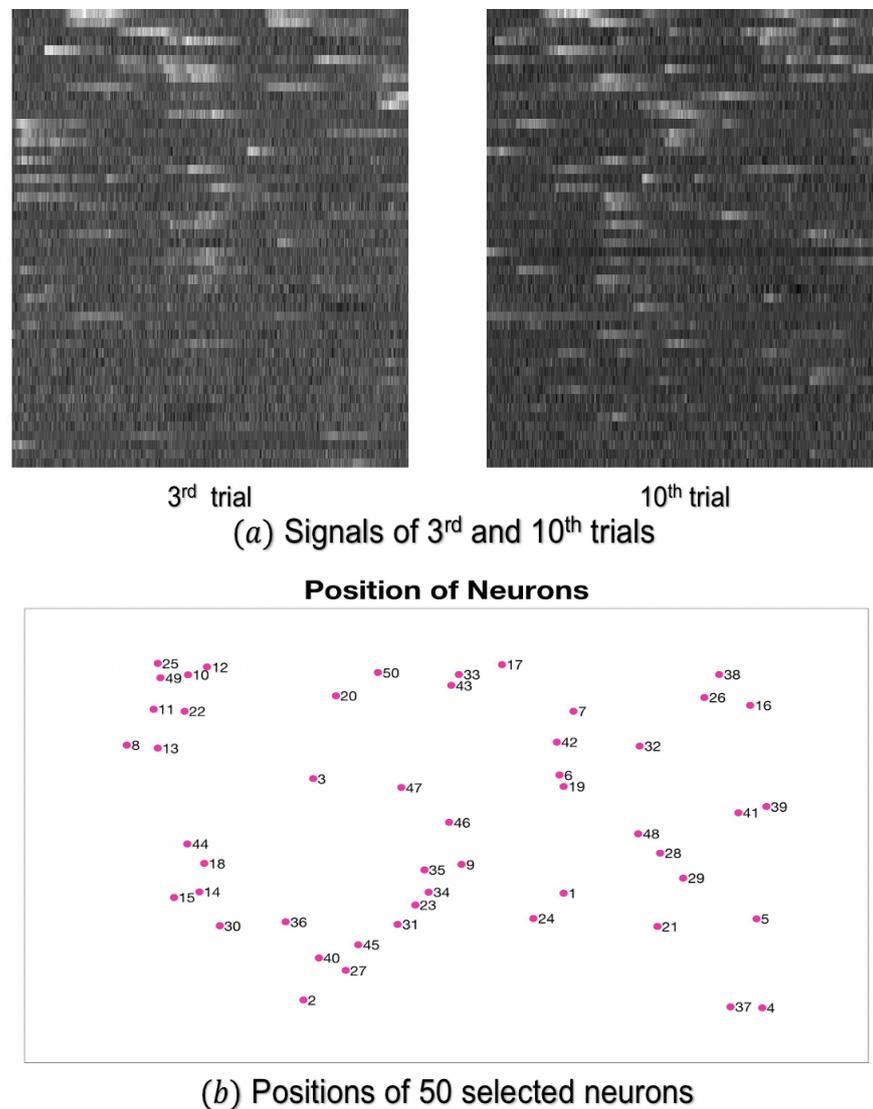

(*a*) Signals of 3rd and 10th trials

(*b*) Positions of 50 selected neurons

Figure 6. Neurons' Signals and Associated Positions.

5.3.3. Interval Partitioning of Data

The brain's reaction time for stimuli is approximately 100 ms, and the delay of the device is around 50 to 100 ms; the time difference between 2-time points is 75 ms; therefore, we choose $T = 213$ in the optimization model to capture the change within 150 ms, and we have 25 to 26 graphs for each stimulus. We choose $K = 30$ (5 percent of the total edge number of a complete graph) to enforce a sparse graph. Uniformly adopting these parameters on the data across 20 trials, we obtain as a result, 213 graphs for each trial $\mathbf{W}_t^1, \ldots, \mathbf{W}_t^{20}$, where $t = 1, \ldots, 213$. Great variations can be observed between graphs across the different trials. We obtain a neuronal connectivity graph/adjacency matrix, by setting the weights (probabilities) less than 0.5, to zero. The sum of the adjacency matrices from different trials in the same time interval $\mathbf{W}_t = \mathbf{W}_t^1 + \cdots + \mathbf{W}_t^{20}$, are also used to determine the connectivity consistency. All consistencies greater than 5 are preserved.

A graph correlation is obtained by way of cross-correlating the duly vectorized weight matrices, e.g., the element value of *i*-th row and *j*-th column of the matrix stands for the correlation between *i*-th and *j*-th graphs' weight vectors, and the matrix is symmetric. The red dash lines divide the plot of Figure 7 into small blocks, representing the exact time

interval corresponding to each specific stimulus shown on the left of the plot, and Figure 7 provides an intuitive view of the memories (time delay) between consecutive stimuli and similarities of graphs activated by similar stimuli.
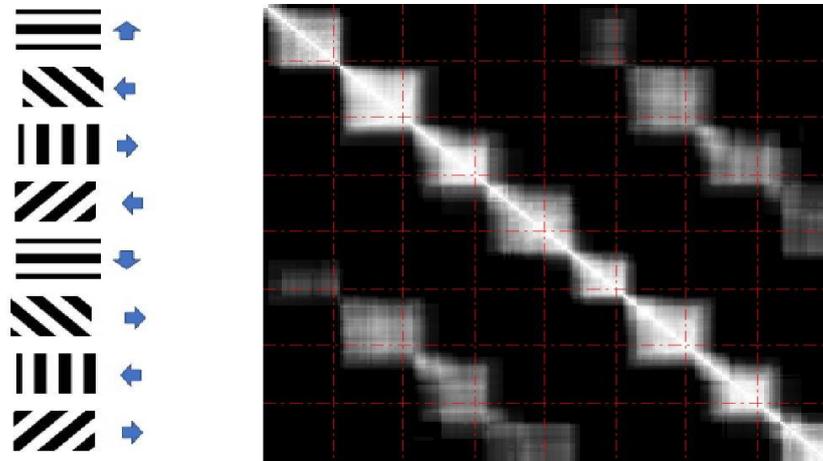

Figure 7. Correlation matrix between graphs.

5.3.4. Improvement with 2-Simplex Information

To reduce the computational complexity and explore information gain by considering 3-way interaction relative to pairwise connection, we repartition the artificial movie into 8- time intervals corresponding to 8 different stimuli. These are also grouped into 4 groups of related stimuli based on the orientation information. According to the result we have in Figure 7, we can observe that the fifth stimulus's neuronal response is highly correlated with that of the fourth stimulus, so two related stimuli, first and fifth stimuli, are not analyzed in this subsection. Further analysis focuses on the comparison of graph representations under other related stimuli (2nd–4th, 6th–8th stimuli).

We run the optimization model in Equations (3) and (5), respectively, with the same hyper-parameter settings on our neuronal-activity dataset. We apply the same post-processing after optimizing the model as in the previous subsection. From Equation (3), we reach the optimal solution $\mathbf{W}_t^1, \dots, \mathbf{W}_t^{20}$ and $\mathbf{W}_t^{(3d),1}, \dots, \mathbf{W}_t^{(3d),20}$ for 20 trials, where $t = 1, \dots, 8$. Then, we transform each weight matrix into an adjacency matrix and each simplex tensor into a binary tensor by setting the weights to 1 if they are greater than 0.5 and 0 otherwise. Afterward, we aggregate the adjacency matrices and the binary tensor, respectively, within the same time intervals from different trials and set the threshold of valid weight matrices to 8 and the simplex tensor threshold to 4. In other words, we only plot the graphs with edges repeating more than 8 times and 2-simplex connections repeating more than 4 times in the same time interval across the 20 trials, thereby capturing response consistency of neurons over trials. The results we get from Equation (3) are called graph representation without simplex information, and those we get from Equation (5) are called graph representation with simplex information. We use a red triangle to fill the space to represent the tight (simplex) connection among three neurons. As shown in Figure 6, plot (a) shows the graph representation calculated from Equation (5) under the second stimulus with red simplex information, while plot (c) shows the graph representation calculated from Equation (3) under the same stimulus without simplex information.

From the graph representations in Figure 8, we can observe that the pairwise connection representations with and without 2-simplex connections are similar to each other under the same or related stimuli. In contrast, the 2-simplex connections under related stimuli are intuitively different. Therefore, we vectorize the weight matrices and calculate the Pearson correlation coefficient between weight vectors under related stimuli. The result is shown in Table 2, and Graph i stands for the graph representation under i-th stimulus.
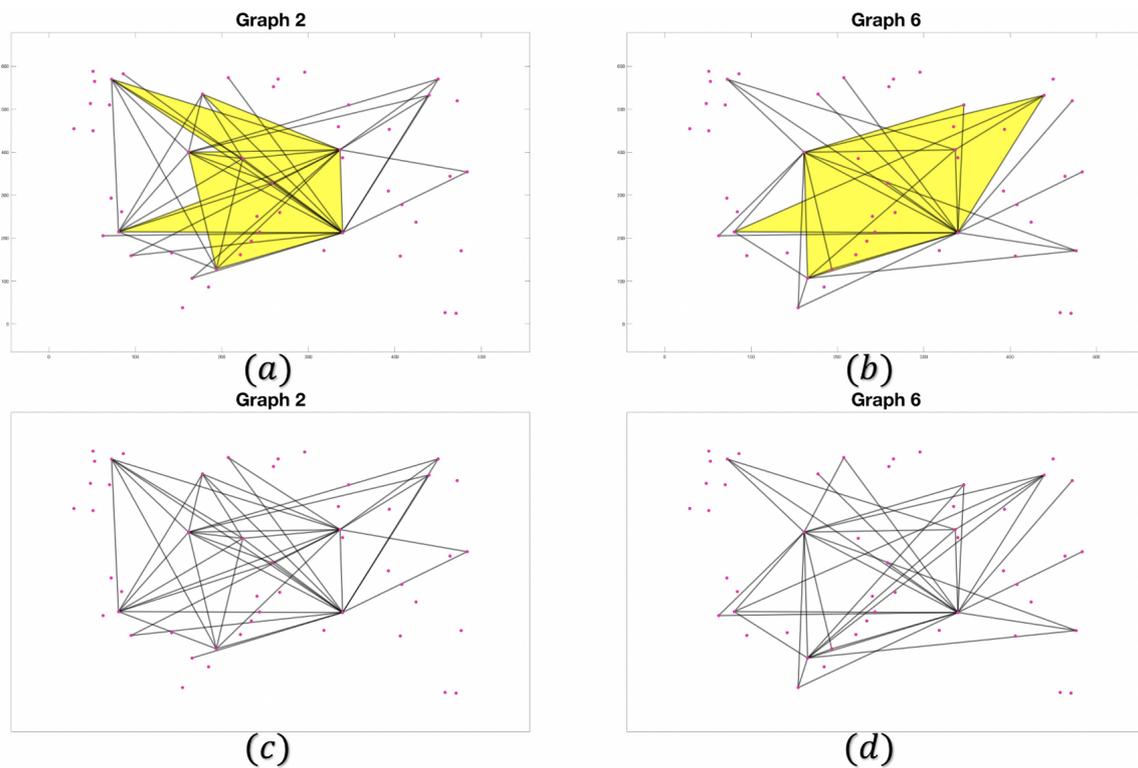
Figure 8. Comparison between related stimuli with and without simplex information: Second and sixth stimuli are related stimuli. The yellow triangle stands for 2-simplex connections. (**a**) The graph representation for the second stimulus with simplex information. (**b**) The graph representation for the sixth stimulus with simplex information. (**c**) The graph representation for the second stimulus without simplex information. (**d**) The graph representation for the sixth stimulus without simplex information.

Table 2. Comparison between related stimuli with and without simplex information.

Related Group	Correlation with 2-Simplex Connection	Correlation without 2-Simplex Connection
graph 2–graph 6	0.49	0.47
graph 3–graph 7	0.55	0.51
graph 4–graph 8	0.39	0.30

Based on the result, $corr(w_2, w_6) < corr(w'_2, w'_6)$, shown in Table 2, and the intuitive observation of the difference between 2-simplex connections under different stimuli, we come up with a hypothesis that the pairwise connection space neglecting higher-order information (W_{2d}), can be decomposed into a new pairwise connection space (W'_{2d}) and a 2-simplex connection space (W_{3d}). The new pairwise connection space primarily includes orientation information, and the 2-simplex connection space contains more direction information. The hypothesis can be concluded per the following formula:

$$W_{2d} = W'_{2d} \bigoplus W_{3d} \tag{11}$$

To verify the hypothesis that 2-simplex connections may contain more directional information, and due to the limited neuronal-activity dataset for comparative test for the same stimulus, we randomly divide 20 trials into 2 groups each of 10 trials. By setting the thresholds for pairwise connection and 2-simplex connection matrix to 5 and 2, respectively, we apply the same post-process to both results with or without simplex information. In a similar way described earlier, we calculate the Pearson correlation coefficient between pairwise connectivity results with or without 2-simplex connections under the same and related stimuli. Our analysis is based on the results under related stimuli in 2 groups.

Figures 9 and 10 show all the correlations calculated in the experiments for pairwise connectivity and 2-simplex connectivity. In Figure 9, the 2 subplots in the first row are graph representations with and without 2-simplex connections for one stimulus in the first group; the 2 subplots in the second row are graph representations with and without 2-simplex connections for the same stimulus in the second group; the 2 subplots in the third row are graph representations with and without 2-simplex connections for the related stimulus in the first group, and the 2 subplots in the fourth row are graph representations with and without 2-simplex connections for the related stimulus in the second group. The marked numbers in Figure 9 represent 6 correlations for pairwise connectivity, which are: (1, 5): correlations between pairwise connection under related stimuli in the same group; (2, 6): correlations between pairwise connection under the same stimulus in different groups; (3, 4): correlations between pairwise connection under related stimuli in different groups. The marked numbers in Figure 10 represent 6 correlations for 2-simplex connectivity, which are: (1′, 2′): correlations between 2-simplex connection under the same stimulus in different groups; (3′, 4′, 5′, 6′): correlations between 2-simplex connection under related stimuli within each group or in the different groups. Table 3 shows all the correlations between graph representations without simplex information. Table 4 shows all the correlations between graph representations with simplex information, and Table 5 shows all the correlations between 2-simplex connections. We run the program 20 times and calculate the mean value of correlations.

Table 3. Correlation between pairwise connections without 2-simplex connectivity.

Related Stimuli	1	2	3	4	5	6
graph 2–graph 6	0.486	0.623	0.250	0.443	0.266	0.266
graph 3–graph 7	0.475	0.531	0.337	0.419	0.361	0.361
graph 4–graph 8	0.219	0.640	0.150	0.142	0.201	0.201

Table 4. Correlation between pairwise connections with 2-simplex connectivity.

Related Stimuli	1	2	3	4	5	6
graph 2–graph 6	0.499	0.727	0.444	0.448	0.471	0.685
graph 3–graph 7	0.464	0.650	0.488	0.457	0.474	0.651
graph 4–graph 8	0.268	0.610	0.246	0.209	0.247	0.567

Table 5. Correlation between 2-Simplex connectivity.

Related Stimuli	1′	2′	3′	4′	5′	6′
graph 2–graph 6	0.697	0.347	0.205	0.204	0.204	0.180
graph 3–graph 7	0.653	0.508	0.331	0.373	0.351	0.317
graph 4–graph 8	0.254	0.234	0.000	0.000	0.000	0.000

We observe that by decomposing the original pairwise connectivity information into new pairwise connectivity information and 2-simplex connectivity information, the correlations of pairwise connectivity representations between related stimuli or same stimulus in different groups (in Table 3) mostly increases by 10 percent compared with those between the original pairwise connectivity representations (in Table 2). Moreover, the first and second columns of Table 4 show the correlations of 2-simplex connectivity representations between the same stimulus in different groups, which is around 15 percent higher than those between related stimuli in the third to the sixth columns.
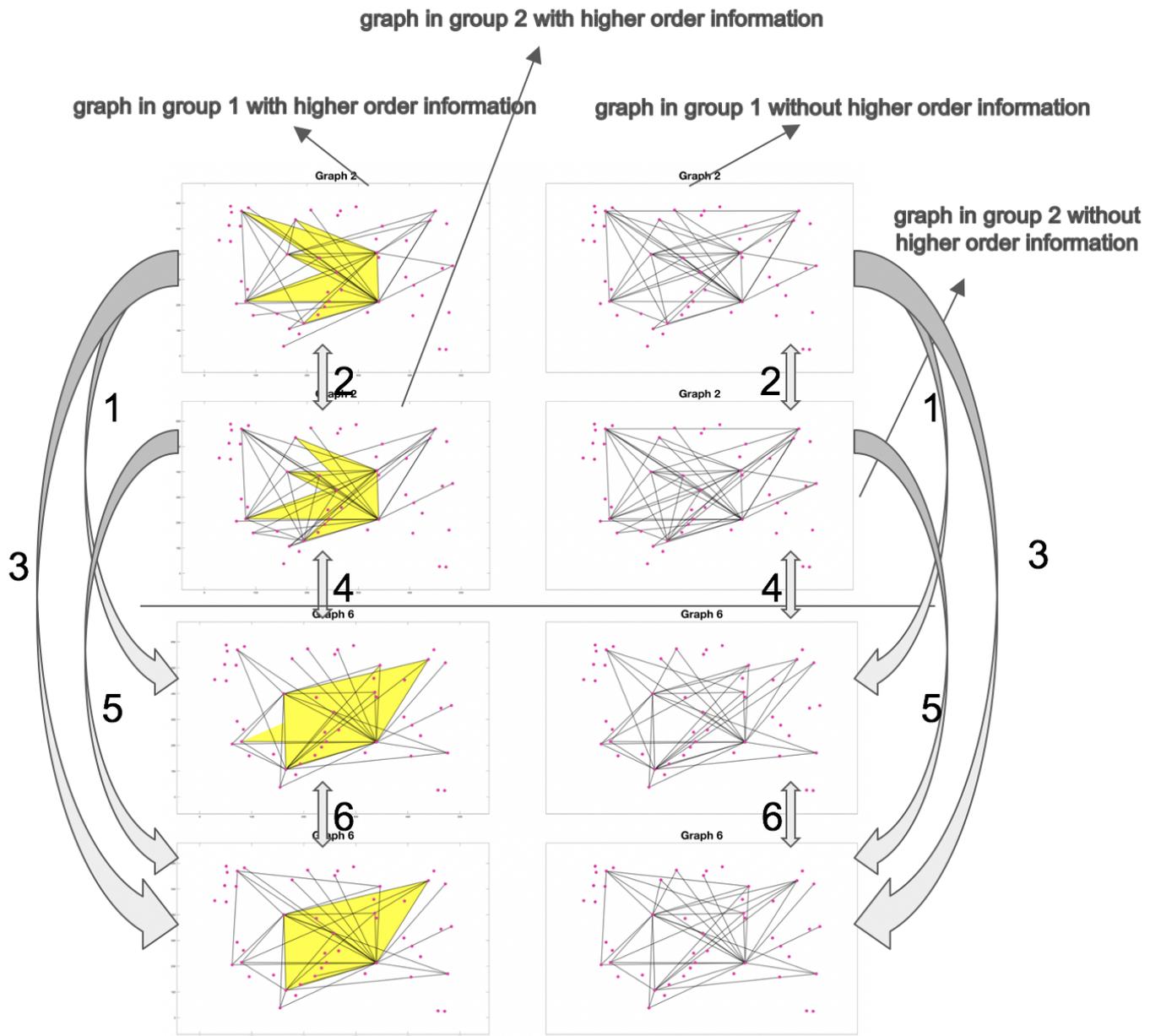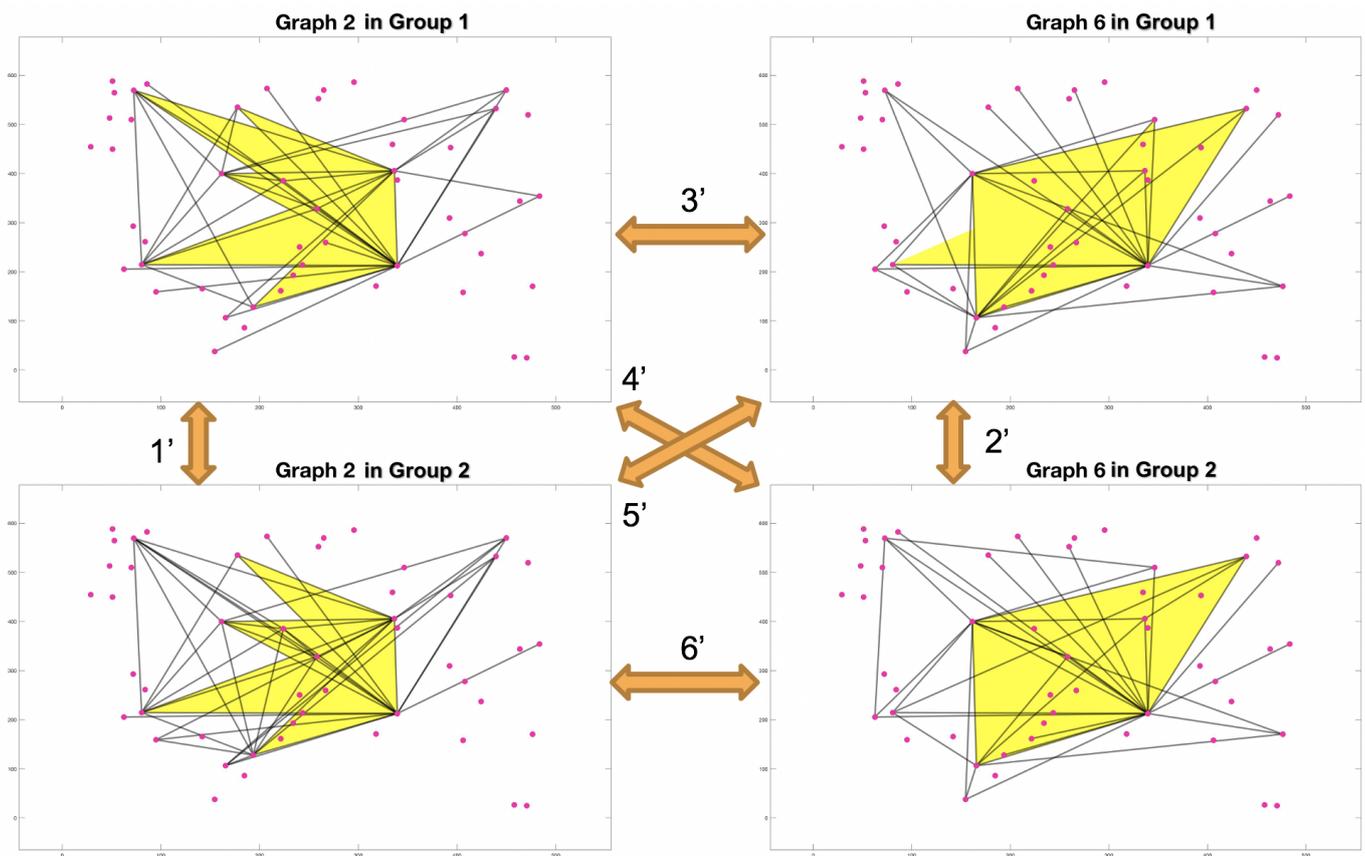
Figure 9. Pairwise connectivity comparison.

Figure 10. 2-Simplex connectivity comparison.

6. Discussion

Learning and analyzing structures from signals, as captured by a graph, is an interesting topic in not only neural-science but also in the computer science realm. Graph structures are instrumental in conveying the potentially complex interacting structure of signals and efficiently preserving all associated information.

Model: The challenge of this neuronal dataset is the low sample size as tens of data points can hardly support learning an adequate graph model, thus making static graph embedding unattainable by partitioning the data into smaller batches. With no prior knowledge about data and no interpolation, our proposed model acquires the data's dynamics and inherent structure. We have also introduced a functional and practical means of tracking the homology of the underlying complex with the advantage of gleaning motion information as explained.

Dataset: From this neuron signal dataset, by way of the mathematical model, we observe variations of neuron connectivity over trials, while preserving similar patterns for similar stimuli in the V1 area, and by looking at different time scales, we also see hysteresis from one stimulus to another. These observations can be seen as an essential step for studying a brain's functional connectivity in response to specific stimuli.

Results: Based on the results of the given neuronal-activity dataset, we were able to conclude that orientation information in the V1 area is adequately reflected by pairwise connectivity. In contrast, direction information may be gleaned from a structure obtained by the connections among neurons beyond pairwise connections. This is an empirical finding based on the model awaiting experimental biological validation.

7. Future Work

Natural movie: While our model was tested on artificial movie data, much work remains in investigating natural movie stimuli. While we expect higher-order information structures will have a stronger presence, variability will also be a great challenge.

8. Conclusions

This paper introduces an optimization driven approach to learning the dynamics of graphs. Three alternative perspectives were presented to capturing the evolution of a graph in time while accounting for the dynamics generated by the nodal activities. In particular, we addressed the difficulty of the low sample rate for detecting graphs and discovered the functional connectivity due to specific stimuli instead of revealing the physical connections of neurons. Future work will include the collaborative activity of neurons and graph transformation to characterize the evolution better.

Author Contributions: Methodology, B.J., Y.H., A.P. and H.K.; software, B.J.; validation, B.J.; formal analysis, B.J.; investigation, B.J.; resources, Y.Y. and S.L.S.; writing—original draft preparation, B.J.; writing—review and editing, H.K. and Y.H.; visualization, B.J.; supervision, H.K.; project administration, H.K.; funding acquisition, H.K. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Jagadish, S.; Parikh, J. Discovery of Friends Using Social Network Graph Properties. U.S. Patent 8,744,976, 3 June 2014.
2. Huang, Y.; Krim, H.; Panahi, A.; Dai, L. Community detection and improved detectability in multiplex networks. *IEEE Trans. Netw. Sci. Eng.* **2019**, *7*, 1697–1709. [CrossRef]
3. Audi, G.; Wapstra, A.; Thibault, C. The AME2003 atomic mass evaluation: (II). Tables, graphs and references. *Nucl. Phys. A* **2003**, *729*, 337–676. [CrossRef]
4. Canutescu, A.A.; Shelenkov, A.A.; Dunbrack, R.L., Jr. A graph-theory algorithm for rapid protein side-chain prediction. *Protein Sci.* **2003**, *12*, 2001–2014. [CrossRef] [PubMed]
5. Sporns, O. *Discovering the Human Connectome*; MIT Press: Cambridge, MA, USA, 2012.
6. Polanía, R.; Paulus, W.; Antal, A.; Nitsche, M.A. Introducing graph theory to track for neuroplastic alterations in the resting human brain: a transcranial direct current stimulation study. *Neuroimage* **2011**, *54*, 2287–2296. [CrossRef] [PubMed]
7. Ocker, G.K.; Hu, Y.; Buice, M.A.; Doiron, B.; Josić, K.; Rosenbaum, R.; Shea-Brown, E. From the statistics of connectivity to the statistics of spike times in neuronal networks. *Curr. Opin. Neurobiol.* **2017**, *46*, 109–119. [CrossRef] [PubMed]
8. Yu, Y.; Stirman, J.N.; Dorsett, C.R.; Smith, S.L. Mesoscale correlation structure with single cell resolution during visual coding. *bioRxiv* **2018**, 469114. doi:10.1101/469114. [CrossRef]
9. Sompolinsky, H.; Yoon, H.; Kang, K.; Shamir, M. Population coding in neuronal systems with correlated noise. *Phys. Rev. E* **2001**, *64*, 051904. [CrossRef] [PubMed]
10. Maretic, H.P.; Thanou, D.; Frossard, P. Graph learning under sparsity priors. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 6523–6527.
11. Chepuri, S.P.; Liu, S.; Leus, G.; Hero, A.O. Learning sparse graphs under smoothness prior. In Proceedings of the 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA, USA, 5–9 March 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 6508–6512.
12. Goyal, P.; Chhetri, S.R.; Canedo, A. dyngraph2vec: Capturing network dynamics using dynamic graph representation learning. *Knowl. Based Syst.* **2019**, *187*, 104816. [CrossRef]
13. Goyal, P.; Kamra, N.; He, X.; Liu, Y. Dyngem: Deep embedding method for dynamic graphs. *arXiv* **2018**, arXiv:1805.11273.
14. Zhang, Y.; Guizani, M. *Game Theory for Wireless Communications and Networking*; CRC Press: Boca Raton, FL, USA, 2011.
15. Baloch, S.; Krim, H.; Kogan, I.; Zenkov, D. Rotation invariant topology coding of 2D and 3D objects using Morse theory. In Proceedings of the IEEE International Conference on Image Processing 2005, Genova, Italy, 14 September 2005; IEEE: Piscataway, NJ, USA, 2005; Volume 3, p. III–796. doi: 10.1109/ICIP.2005.1530512. [CrossRef]

16. Lee, H.; Chung, M.K.; Kang, H.; Lee, D.S. Hole detection in metabolic connectivity of Alzheimer's disease using k- Laplacian. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Boston, MA, USA, 14–18 September 2014; Springer: Berlin, Germany, 2014; pp. 297–304.

17. Dong, X.; Thanou, D.; Frossard, P.; Vandergheynst, P. Learning Laplacian matrix in smooth graph signal representations. *IEEE Trans. Signal Process.* **2016**, *64*, 6160–6173. [CrossRef]

18. Chung, F.R.; Graham, F.C. *Spectral Graph Theory*; Number 92; American Mathematical Society: Providence, RI, USA, 1997.

19. MacLane, S. *Homology*; Springer Science & Business Media: Berlin, Germany, 2012.

20. Parikh, N.; Boyd, S. Proximal algorithms. *Found. Trends Optim.* **2014**, *1*, 127–239. [CrossRef]

21. Combettes; P.L.; Pesquet, J.C. A proximal decomposition method for solving convex variational inverse problem. *Inverse Probl.* **2008**, *24*, 065014. [CrossRef]

22. Kass, R.E.; Amari, S.I.; Arai, K.; Brown, E.N.; Diekman, C.O.; Diesmann, M.; Doiron, B.; Eden, U.T.; Fairhall, A.L.; Fiddyment, G.M.; et al. Computational neuroscience: Mathematical and statistical perspectives. *Annu. Rev. Stat. Appl.* **2018**, *5*, 183–214. [CrossRef] [PubMed]

23. Crick, F.; Koch, C. Are we aware of neural activity in primary visual cortex? *Nature* **1995**, *375*, 121. [CrossRef] [PubMed]

24. Niell, C.M.; Stryker, M.P. Modulation of visual responses by behavioral state in mouse visual cortex. *Neuron* **2010**, *65*, 472–479. [CrossRef] [PubMed]

25. Hubel, D.H.; Wiesel, T.N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.* **1962**, *160*, 106–154. [CrossRef] [PubMed]

26. Adelson, E.H.; Bergen, J.R. Spatiotemporal energy models for the perception of motion. *JOSA A* **1985**, *2*, 284–299. [CrossRef] [PubMed]

27. Bradley, D.C.; Goyal, M.S. Velocity computation in the primate visual system. *Nat. Rev. Neurosci.* **2008**, *9*, 686–695. [CrossRef] [PubMed]

28. Ji, N.; Freeman, J.; Smith, S.L. Technologies for imaging neural activity in large volumes. *Nat. Neurosci.* **2016**, *19*, 1154. [CrossRef] [PubMed]