

RESEARCH

# Hierarchies & Lower Bounds in Theoretical Connectomics

Venkatakrishnan Ramaswamy

Correspondence:

vramaswamy@ncbs.res.in  
Simons Centre for the Study of  
Living Machines, National Centre  
for Biological Sciences, GKVK,  
Bellary Road, 560065 Bangalore,  
India

Full list of author information is  
available at the end of the article

## Abstract

Connectomics is a sub-field of Neuroscience aimed at determining connectomes – exact structures of neurons and their synaptic connections in nervous systems. A number of ongoing initiatives at the present time are working towards the goal of ascertaining the connectomes or parts thereof of various organisms. Determining the detailed physiological response properties of all the neurons in these connectomes is out of reach of current experimental technology. It is therefore unclear, to what extent knowledge of the connectome alone will advance a mechanistic understanding of computation occurring in these neuronal circuits, especially when the high-level function(s) of the said circuit is unknown.

We are pursuing a research program to build theory in order to investigate these issues. In previously published work [1], towards this end, we have developed a theory of connectomic constraints for feedforward networks of neurons. Specifically, for feedforward networks equipped with neurons that obey a deterministic spiking neuron model, we asked if just by knowing the structure of a network, we could rule out spike-timed computations that it could be doing, no matter what response properties each of its neurons may have. Our neurons were abstract mathematical objects that satisfied a small number of axioms that correspond to certain broadly-obeyed properties of neurons.

Here, we develop additional theoretical tools and notions to address these questions. The idea is to study the space of all possible spike-train to spike-train transformations. We are interested in asking how the subset of transformations spanned by networks of specific architectures can be related to hierarchical subsets of the space that are characterized by particular mathematical properties of transformations. In particular, given such a hierarchy of subsets, what is the “smallest” subset that contains the set of transformations spanned by networks of a specific class of architectures? Even if one cannot establish such a subset exactly, proving bounds on it (according to the hierarchy) might offer insight. After setting up the mathematical framework to make these notions precise, we construct explicit classes of hierarchies and prove a number of such lower bounds.

**Keywords:** Connectomics; Axiomatic Theory; Spiking Neurons

## 1 Introduction

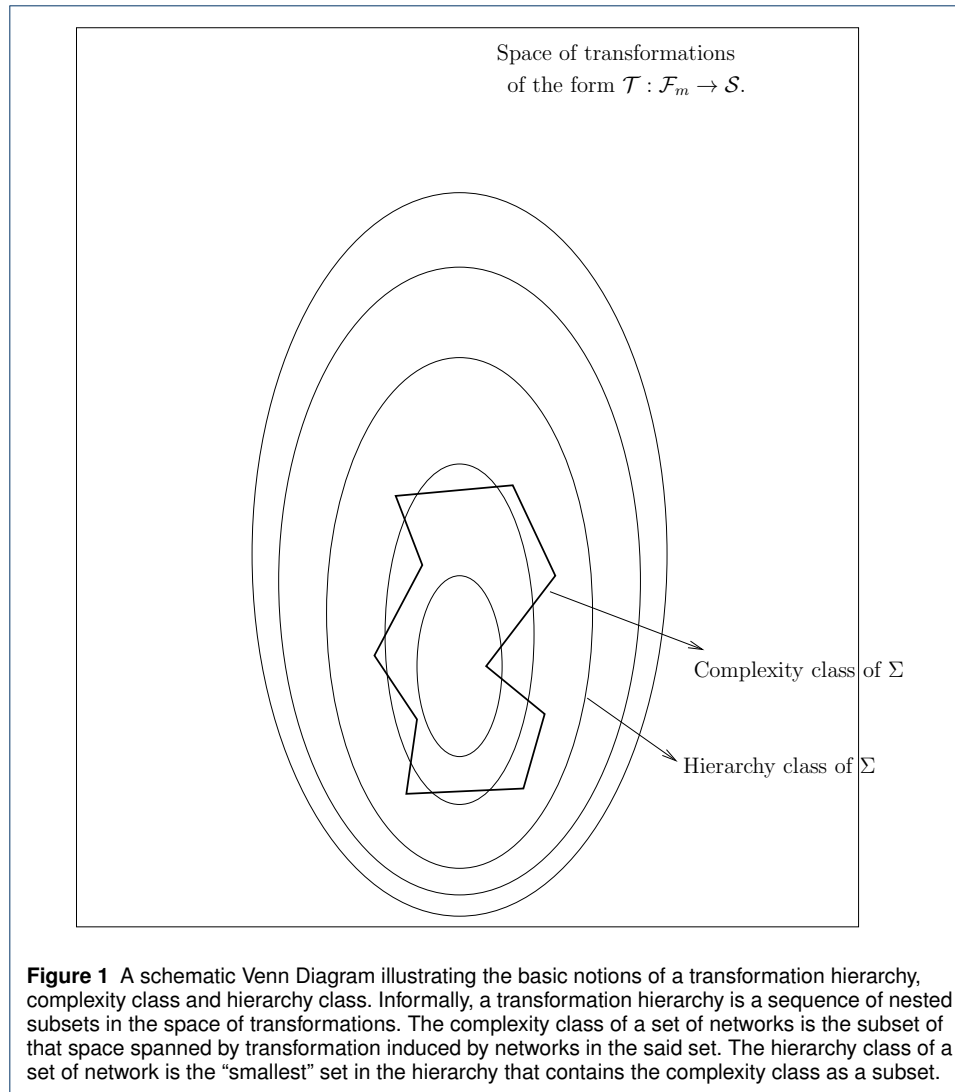
Progress in experimental techniques, since the turn of the century [2, 3, 4, 5, 6, 7], has led to a rekindling of interest in determining exact structures of (microscale) neural circuits that comprise nervous systems [8, 9, 10, 11, 12], which were rechristened as connectomes [13]. The first connectome – that of the nematode *Caenorhabditis elegans* – was determined in

1986 [14], after an effort that took over a decade, much of it manual. The only other complete connectome to be determined, to date, is that of the larval tadpole *Ciona intestinalis* [15], which has an asymmetric nervous system and in fact has fewer neurons than *C. elegans*; this was accomplished using modern automated techniques. That said, a significant percentage of the larval *Drosophila* connectome has already been reconstructed, although it is not publicly available at the time of writing [16]. Likewise, the electron microscopy volume of the entire larval zebrafish [17] and adult *Drosophila* [18] are publicly available, although they haven't been fully reconstructed at the time of writing. Indeed, a number of studies have already fruitfully used Connectomics to address several questions about neural circuits and their function [19, 20, 12, 21, 22, 23, 24, 25, 26, 27, 28]. Also, it has been possible to first perform functional imaging of neural circuit activity in-vivo and then perform connectomic reconstruction of circuits in the same piece of tissue [19, 20, 28].

Current techniques, however, do not allow determination of detailed physiological response properties of all neurons in the connectomes and this is likely to be the case in the foreseeable future. It is therefore unclear to what extent knowledge of the connectome will inform us about mechanistic computation occurring in the neural circuits being reconstructed. Indeed, this has been one criticism of the connectomics efforts [29], in that it is unclear what exactly we will learn by knowing the connectome. Part of the criticism originates from a lack of theory. In particular, there is a need for theory that can use connectomic data to infer constraints on computation in neural circuits and generate hypotheses about mechanistic computation in the neural circuits. We are pursuing a research program to build such theory. To this end, in previously published work [1], we have developed a theory of connectomic constraints on computation in neural circuits. Specifically, for feedforward networks equipped with neurons that obey a deterministic spiking neuron model, we asked if just by knowing the structure of a network, we can rule out computations that it could be doing, no matter what response properties each of its neurons may have. We also stipulated the need to demonstrate a network with a different structure comprising “simple” neurons that could indeed effect the computation in question. After setting up a mathematical framework within which these questions could be precisely posed, we showed results of this form (which we call *complexity results*) for certain classes of network architectures. We also proved, mathematically, that for certain other classes of network architectures, given our limited assumptions on the individual neurons<sup>[1]</sup>, there are fundamental limits to constraints imposed by network structure alone. Our neurons were abstract mathematical objects that satisfied a small number of axioms that correspond to certain broadly obeyed properties of neurons. Complexity results, thus, were in the form of mathematical proofs that use these axioms and the structure of the network in question to establish explicit spike-train to spike-train transformations that could not be effected by any network of the said structure, which could in turn be effected by a network of a different structure. Indeed, the broad program in this line of research is to start from first principles, as we have done in [1] and prove such results for networks of axiomatic neurons with progressively larger number of axioms and also generalize the theory to treat the case of recurrent networks. The idea is to eventually use this type of theory as a starting point to rule out specific computations in neural circuits for which connectomic data is available. This might then be used to formulate hypotheses about mechanistic computation in such circuits, which could be tested experimentally. In addition to this goal, the theory would also allow us to understand which

---

<sup>[1]</sup>The assumptions manifest as axioms in our abstract model.



aspects of network structure are crucial to the manifestation of what kinds of spike-timed computation. This could, in principle, aid in understanding why some types of network motifs might be conserved across individuals or species, especially in the emerging field of Comparative Connectomics.

Here, we develop additional theoretical tools and notions to address these questions. The idea is to study the space of all possible spike-train to spike-train transformations. In particular, we are interested in asking how the subset of transformations spanned by networks of specific architectures can be related to subsets of the space that are characterized by particular mathematical properties of transformations. While the former types of subsets are related to networks of neurons, the latter type are related to transformations alone and mathematical properties thereof. More concretely, using mathematical properties of transformations (i.e. without any reference to neurons or networks), the idea is to identify a sequence of subsets of this space, with each subset contained in the subsequent one in the sequence. We call this sequence of subsets a *Transformation Hierarchy*. This allows us to relate sets in a transformation hierarchy to sets of transformations spanned by specific network architectures, which we call the *Complexity Classes* of those architectures. We

do this by finding the “smallest” set in the hierarchy that contains the complexity class in question as a subset. This set is called the *Hierarchy Class* of the architecture with respect to the said transformation hierarchy. Figure 1 provides a Venn Diagram illustrating these notions. Even if we cannot establish the hierarchy class of a given architecture with respect to a hierarchy, proving bounds<sup>[2]</sup> on them might offer insight. As an application of these notions, we have constructed explicit classes of transformation hierarchies. For every set (starting from a certain set) in each such hierarchy, we demonstrate network architectures, for which the set in question is a lower bound on the hierarchy class of the said network architecture.

The reader familiar with Theoretical Computer Science will observe that the approach taken here is somewhat reminiscent of that in Computational Complexity Theory, although the settings, details and questions are very different.

## 2 Definitions and Preliminaries

The treatment here is largely self-contained. In order to make this so, in this section, we reproduce verbatim from [1], definitions that constitute the basic mathematical formalism used to describe spike trains and operations on them.

An *action potential* or *spike* is a stereotypical event characterized by the time instant at which it is initiated in the neuron, which is referred to as its *spike time*. Spike times are represented relative to the present by real numbers, with positive values denoting past spike times and negative values denoting future spike times. A *spike-train*  $\vec{x} = \langle x^1, x^2, \dots, x^k, \dots \rangle$  is a strictly increasing sequence of spike times, with every pair of spike times being at least  $\alpha$  apart, where  $\alpha > 0$  is the absolute refractory period<sup>[3]</sup> and  $x^i$  is the spike time of spike  $i$ . An *empty spike-train*, denoted by  $\vec{\phi}$ , is one which has no spikes. A *time-bounded spike-train* (with *bound*  $(a, b)$ ) is one where all spike times lie in the bounded interval  $(a, b)$ , for some  $a, b \in \mathbb{R}$ . We use  $\mathcal{S}$  to denote the set of all spike trains and  $\bar{\mathcal{S}}_{(a,b)}$  to denote the set of all time-bounded spike-trains with bound  $(a, b)$ . A spike-train is said to have a *gap* in the interval  $(c, d)$ , if it has no spikes in that time interval. Furthermore, this gap is said to be of *length*  $d - c$ .

We use the term *spike-train ensemble* to denote a collection of spike-trains. Thus, formally, a *spike-train ensemble*  $\chi = \langle \vec{x}_1, \dots, \vec{x}_m \rangle$  is a tuple of spike-trains. The *order* of a spike-train ensemble is the number of spike-trains in it. For example,  $\chi = \langle \vec{x}_1, \dots, \vec{x}_m \rangle$  is a spike-train ensemble of order  $m$ . A *time-bounded spike-train ensemble* (with *bound*  $(a, b)$ ) is one in which each of its spike-trains is time-bounded (with *bound*  $(a, b)$ ). A spike-train ensemble  $\chi$  is said have a *gap* in the interval  $(c, d)$ , if each of its spike trains has a gap in the interval  $(c, d)$ .

Next, we define some operators to time-shift, segment and assemble/disassemble spike-trains from spike-train ensembles. Let  $\vec{x} = \langle x^1, x^2, \dots, x^k, \dots \rangle$  be a spike-train and  $\chi = \langle \vec{x}_1, \dots, \vec{x}_m \rangle$  be a spike-train ensemble. The *time-shift operator for spike-trains* is used to time-shift all the spikes in a spike-train. Thus,  $\sigma_t(\vec{x}) = \langle x^1 - t, x^2 - t, \dots, x^k - t, \dots \rangle$ . The *time-shift operator for spike-train ensembles* is defined as  $\sigma_t(\chi) = \langle \sigma_t(\vec{x}_1), \dots, \sigma_t(\vec{x}_m) \rangle$ .

<sup>[2]</sup>Formally, the bounds are with respect to the partial ordering induced by set inclusion.

<sup>[3]</sup>We assume a single fixed absolute refractory period for all neurons, for convenience, although our results would be no different if different neurons had different absolute refractory periods.

The *truncation operator for spike-trains* is used to “cut out” specific segments of a spike-train. It is defined as follows:  $\Xi_{[a,b]}(\vec{x})$  is the time-bounded spike-train with bound  $[a, b]$  that is identical to  $\vec{x}$  in the interval  $[a, b]$ .  $\Xi_{(a,b)}(\vec{x})$ ,  $\Xi_{(a,b]}(\vec{x})$  and  $\Xi_{[a,b)}(\vec{x})$  are defined likewise. In the same vein,  $\Xi_{[a,\infty)}(\vec{x})$  is the spike-train that is identical to  $\vec{x}$  in the interval  $[a, \infty)$  and has no spikes in the interval  $(-\infty, a)$ . Similarly,  $\Xi_{(-\infty,b]}(\vec{x})$  is the spike-train that is identical to  $\vec{x}$  in the interval  $(-\infty, b]$  and has no spikes in the interval  $(b, \infty)$ .  $\Xi_{(a,\infty)}(\vec{x})$  and  $\Xi_{(-\infty,b)}(\vec{x})$  are also defined similarly. The *truncation operator for spike-train ensembles* is defined as  $\Xi_{[a,b]}(\chi) = \langle \Xi_{[a,b]}(\vec{x}_1), \dots, \Xi_{[a,b]}(\vec{x}_m) \rangle$ .  $\Xi_{(a,b)}(\chi)$ ,  $\Xi_{(a,b]}(\chi)$ ,  $\Xi_{[a,b)}(\chi)$ ,  $\Xi_{[a,\infty)}(\chi)$ ,  $\Xi_{(-\infty,b]}(\chi)$ ,  $\Xi_{(a,\infty)}(\chi)$  and  $\Xi_{(-\infty,b)}(\chi)$  are defined likewise. Furthermore,  $\Xi_t(\cdot)$  is shorthand for  $\Xi_{[t,t]}(\cdot)$ . The *projection operator for spike-train ensembles* is used to “pull-out” a specific spike-train from a spike-train ensemble. It is defined as  $\Pi_i(\chi) = \vec{x}_i$ , where  $1 \leq i \leq m$ . Let  $\vec{y}_1, \vec{y}_2, \dots, \vec{y}_n$  be spike-trains. The *join operator for spike-trains* is used to “bundle-up” a set of spike-trains to obtain a spike-train ensemble. It is defined as  $\vec{y}_1 \sqcup \vec{y}_2 \sqcup \dots \sqcup \vec{y}_n = \bigsqcup_{i=1}^n \vec{y}_i = \langle \vec{y}_1, \vec{y}_2, \dots, \vec{y}_n \rangle$ .

The neuron model used is a deterministic spiking neuron model. One example would be the abstract model used in [1], although a more restrictive model with more axioms would also allow for the notions defined in this paper.

Next, we have the notion of a Flush Criterion that appears as Definition 5 in [1].

**Definition** (Flush Criterion) A spike-train ensemble  $\chi$  is said to satisfy a *T-Flush Criterion*, if all its spikes lie in the interval  $(0, T)$ , i.e. it has no spikes upto time instant  $T$  and since time instant 0.

Let the set of spike-train ensembles of order  $m$  that satisfy the T-Flush criterion be  $\mathcal{F}_m^T$ . Let  $\mathcal{F}_m = \bigcup_{T \in \mathbb{R}^+} \mathcal{F}_m^T$ .

Each feedforward network induces<sup>[4]</sup> a transformation of the form  $\mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$ , where  $\mathcal{S}$  is the space of all spike trains.

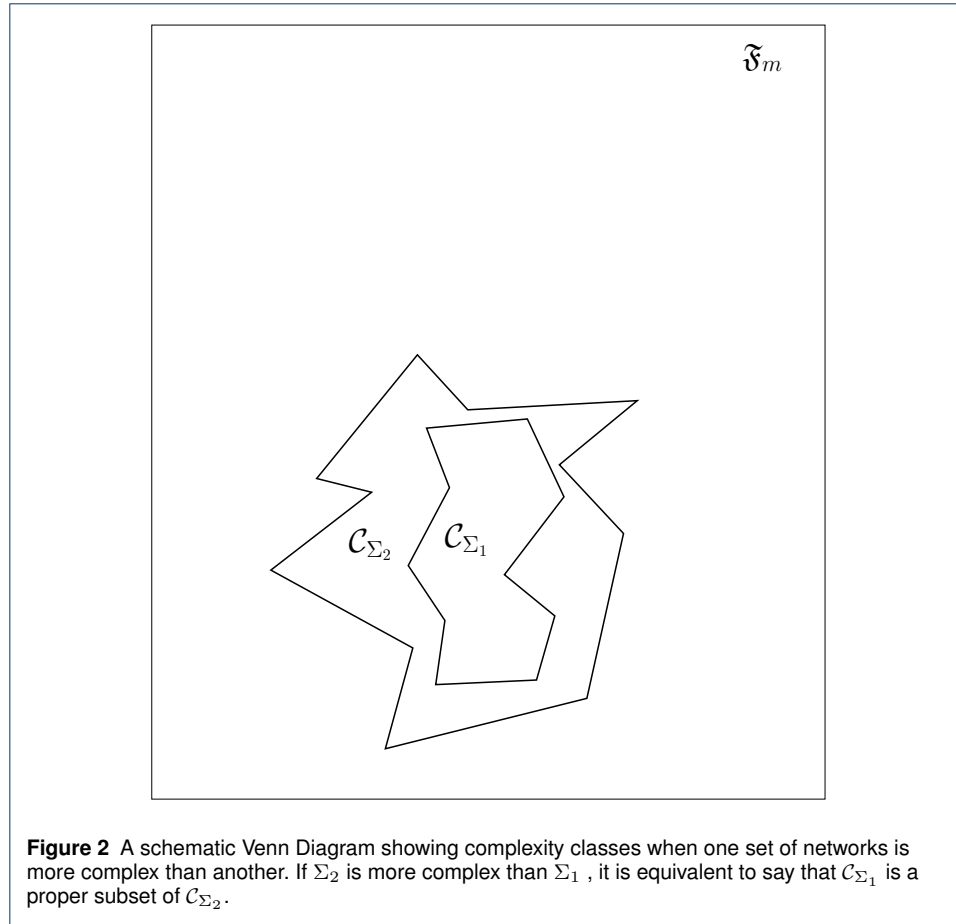
Next, we define the relation *more complex than*. For brevity, the definition below combines Definition 6 and Lemma 5 in [1] in order to define this relation equivalently in terms of spike-train ensembles satisfying the Flush Criterion.

**Definition** (Transformational Complexity) Let  $\Sigma_1$  and  $\Sigma_2$  be two sets of feedforward networks, each network being of order  $m$ , with  $\Sigma_1 \subseteq \Sigma_2$ . The set  $\Sigma_2$  is said to be *more complex than*  $\Sigma_1$ , if there exists an  $\mathcal{N}' \in \Sigma_2$  such that for all  $\mathcal{N} \in \Sigma_1$ ,  $\mathcal{T}_{\mathcal{N}'} \neq \mathcal{T}_{\mathcal{N}}$ .

Each result of the above form is called a *complexity result*.

It is important to emphasize that we have skipped a great deal of other notation, notions, lemmas and theorems in [1] that are not central to the treatment here.

<sup>[4]</sup>In fact, each feedforward network induces a transformation even in the case when input to it is a stream of spikes from a biologically-relevant spiking regime (not preceded by quiescence). The theory towards this is developed in [1]. It turns out (as exposted in [1]) that one can work with  $\mathcal{F}_m$  without loss of generality in so far as establishing complexity results is concerned.



### 3 Complexity Classes, Transformation Hierarchies and Hierarchy Classes

Let  $\mathfrak{F}_m$  be the space of all possible transformations of the form  $\mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$  that map spike-train ensembles of order  $m$  which satisfy the Flush criterion to output spike trains. Each acyclic network of order  $m$  induces one such transformation. A set of networks of order  $m$  therefore induces a class of such transformations, which we call the *complexity class* of that set.

**Definition 1** (Complexity Class) Let  $\Sigma$  be a set of acyclic networks of order  $m$ . The *complexity class* of  $\Sigma$ ,  $C_\Sigma$ , is defined to be the set  $\bigcup_{\mathcal{N} \in \Sigma} \mathcal{T}_{\mathcal{N}}$ .

As is clear from Lemma 6 in [1], no complexity class spans the entire space  $\mathfrak{F}_m$ . Lemma 5 from [1] implies that questions of relative complexity of sets of networks can be posed in terms of questions about containment of their complexity classes. Figure 2 illustrates the situation and the next proposition formalizes it.

**Lemma 1** Let  $\Sigma_1$  and  $\Sigma_2$  be two sets of acyclic networks, each network being of order  $m$ , with  $\Sigma_1 \subseteq \Sigma_2$ . Further, let  $C_{\Sigma_1}$  and  $C_{\Sigma_2}$  be the corresponding complexity classes. Then,  $\Sigma_2$  is more complex than  $\Sigma_1$  if and only if  $C_{\Sigma_1} \subset C_{\Sigma_2}$ .

*Proof* First, suppose  $\Sigma_2$  is more complex than  $\Sigma_1$ . That  $\mathcal{C}_{\Sigma_1} \subseteq \mathcal{C}_{\Sigma_2}$  follows immediately from the fact that  $\Sigma_1 \subseteq \Sigma_2$ . Lemma 5 from [1] implies that  $\exists \mathcal{N}' \in \Sigma_2$  such that  $\forall \mathcal{N} \in \Sigma_1, \mathcal{T}_{\mathcal{N}'} \neq \mathcal{T}_{\mathcal{N}}$ . That is,  $\mathcal{T}_{\mathcal{N}'} \in \mathcal{C}_{\Sigma_2}$  and  $\mathcal{T}_{\mathcal{N}'} \notin \mathcal{C}_{\Sigma_1}$ . Since  $\mathcal{C}_{\Sigma_1} \subseteq \mathcal{C}_{\Sigma_2}$ , it follows that  $\mathcal{C}_{\Sigma_1} \subset \mathcal{C}_{\Sigma_2}$ .

To prove the other direction, assume  $\mathcal{C}_{\Sigma_1} \subset \mathcal{C}_{\Sigma_2}$ . Therefore,  $\exists \mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$  such that  $\mathcal{T} \in \mathcal{C}_{\Sigma_2}$  and  $\mathcal{T} \notin \mathcal{C}_{\Sigma_1}$ . By definition of  $\mathcal{C}_{\Sigma_2}$ ,  $\exists \mathcal{N} \in \Sigma_2$ , so that  $\mathcal{T}_{\mathcal{N}} = \mathcal{T}$ . Since  $\mathcal{T} \notin \mathcal{C}_{\Sigma_1}$ , the definition of  $\mathcal{C}_{\Sigma_1}$  implies that  $\forall \mathcal{N}' \in \Sigma_1, \mathcal{T}_{\mathcal{N}'} \neq \mathcal{T}$ . Therefore,  $\Sigma_2$  is more complex than  $\Sigma_1$ .  $\square$

Next, we make precise the notion of a Transformation Hierarchy. Informally, a Transformation Hierarchy is a sequence of subsets of this space, with each subset contained in the subsequent one in the sequence.

**Definition 2** (Transformation Hierarchy) A Transformation Hierarchy  $\mathcal{H}$  in  $\mathfrak{F}_m$  is a sequence of subsets  $\langle H_1, H_2, \dots, H_i, \dots, \mathfrak{F}_m \rangle$  of  $\mathfrak{F}_m$  with  $H_i \subset H_{i+1}, \forall i = 1, 2, \dots$

Note that the above definition exists independent of the existence of networks. That is, a hierarchy is defined only in terms of properties of transformations in its constituent sets. Figure 3 provides an illustration. The next definition provides a connection between sets of acyclic networks and hierarchies. Each set of acyclic networks  $\Sigma$  is associated with a specific set in the hierarchy called its *hierarchy class*, so that the hierarchy class is the smallest set in the hierarchy that contains the complexity class of  $\Sigma$ .

**Definition 3** (Hierarchy Class) Let  $\Sigma$  be a set of acyclic networks, each of order  $m$ , and let  $\mathcal{H} = \langle H_1, H_2, \dots, H_i, \dots, \mathfrak{F}_m \rangle$  be a transformation hierarchy in  $\mathfrak{F}_m$ . The *Hierarchy Class*  $H_{\Sigma}$  of  $\Sigma$  in  $\mathcal{H}$  is the set  $H_i$  with  $\mathcal{C}_{\Sigma} \subseteq H_i$  and  $\mathcal{C}_{\Sigma} \not\subseteq H_{i-1}$ , if such an  $H_i$  exists and  $\mathfrak{F}_m$  otherwise.

Note that the Hierarchy class of a set of networks is well-defined. This is because  $\mathcal{C}_{\Sigma} \subset \mathfrak{F}_m$ , so there is atleast one set in the hierarchy that contains every complexity class. Also, the sets in the hierarchy are well-ordered<sup>[5]</sup> due to which every collection of sets from the hierarchy (which contain  $\mathcal{C}_{\Sigma}$ ) has a smallest set.

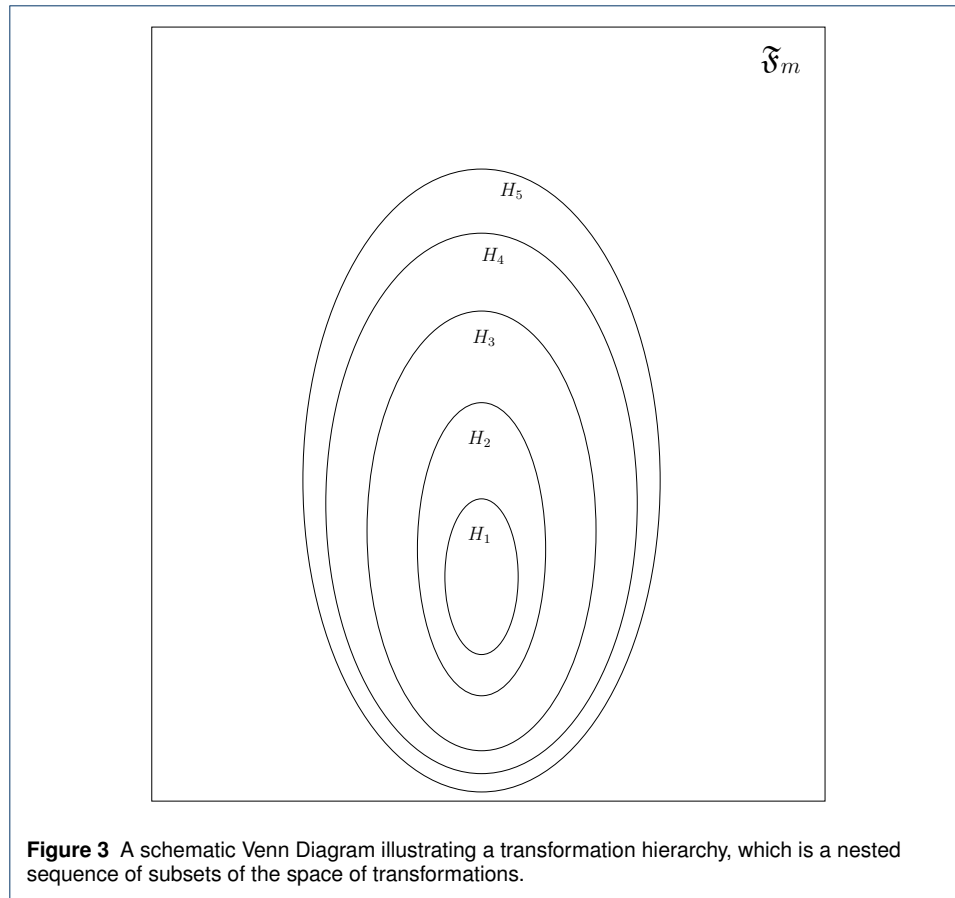
The above definitions allow us to create a variety of hierarchies based on specific properties of transformations. If we can then say something about the hierarchy classes of specific architectures in each hierarchy, it enables us to get a better understanding of various aspects of the transformations effected by networks with this architecture.

Even if we cannot identify the hierarchy class of a given architecture in a hierarchy, proving bounds<sup>[6]</sup> on them might give us some insight. Also, as alluded to before, these bounds can be used to establish that one set of networks is more complex than another.

A set in a hierarchy is an *upper bound* on a hierarchy class if it contains the hierarchy class as a subset. Likewise, a set in a hierarchy is a *lower bound* on a hierarchy class if the hierarchy class contains the set as a subset.

<sup>[5]</sup>The ordering is by set inclusion.

<sup>[6]</sup>Again, formally, the bounds are with respect to the partial ordering induced by set inclusion.



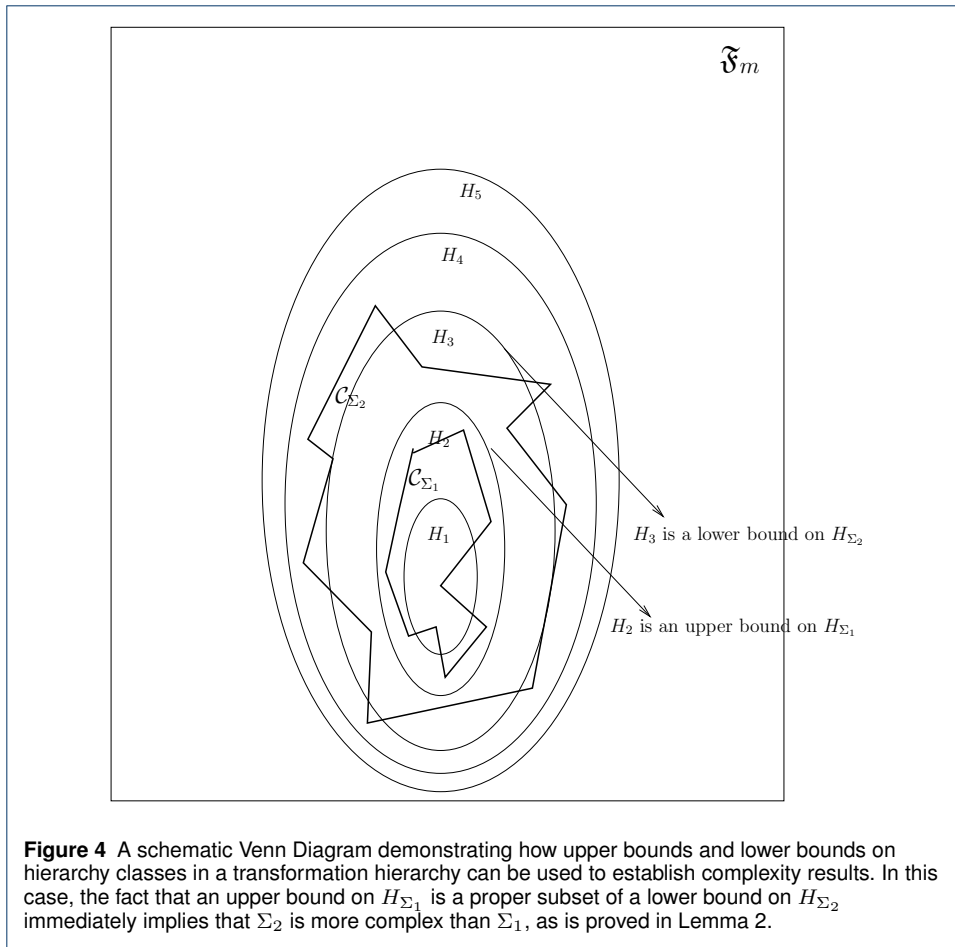
Bounds on hierarchy classes of specific sets of networks can be used to establish complexity results. If there are two sets of networks, the first contained in the second and if an upper bound on the hierarchy class of the first set is “smaller” than a lower bound on the hierarchy class of the second (with respect to the same hierarchy), then the second set is more complex than the first. Figure 4 gives a picture. Note that this is just a sufficient condition, not a necessary one, for one set to be more complex than the other<sup>[7]</sup>. The next lemma formalizes the above observations.

**Lemma 2** *Let  $\Sigma_1$  and  $\Sigma_2$  be two sets of acyclic networks, each comprising networks of order  $m$ , with  $\Sigma_1 \subseteq \Sigma_2$ . Furthermore, let  $H_{\Sigma_1}$  and  $H_{\Sigma_2}$  be the corresponding hierarchy classes in a transformation hierarchy  $\mathcal{H} = \langle H_1, H_2, \dots, H_i, \dots, \mathfrak{F}_m \rangle$  in  $\mathfrak{F}_m$ . Moreover, let  $H_u$  be an upper bound on  $H_{\Sigma_1}$  and  $H_l$  be a lower bound on  $H_{\Sigma_2}$ . If  $H_u \subset H_l$ , then  $\Sigma_2$  is more complex than  $\Sigma_1$ .*

*Proof* Let  $\mathcal{C}_{\Sigma_1}$  and  $\mathcal{C}_{\Sigma_2}$  be the complexity classes of  $\Sigma_1$  and  $\Sigma_2$  respectively. By hypothesis,  $\mathcal{C}_{\Sigma_1} \subseteq H_u$  and  $H_l \subseteq \mathcal{C}_{\Sigma_2}$ . Since,  $H_u \subset H_l$ , we have  $\mathcal{C}_{\Sigma_1} \subset \mathcal{C}_{\Sigma_2}$ . From Lemma 1, it now follows that  $\Sigma_2$  is more complex than  $\Sigma_1$ .  $\square$

<sup>[7]</sup>That is, depending on the hierarchy in question, it is possible that both sets have the same hierarchy class, yet one is more complex than the other.





Indeed, this suggests an economical way to prove complexity results, since the upper bounds and lower bounds could apply to several sets of networks.

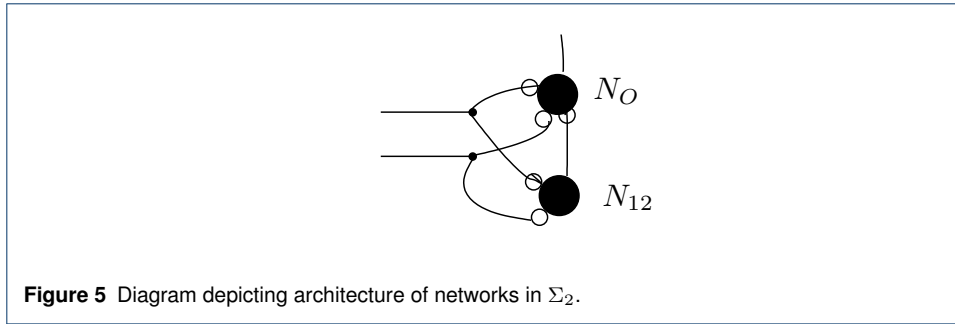
In the next section, we apply these notions to explicitly construct a transformation hierarchy and prove some lower bounds for some architectures according to the hierarchy.

#### 4 Lower Bounds on the Hierarchy Classes of some Architectures in a certain Transformation Hierarchy

In this section, we construct specific sequences of subsets of  $\mathfrak{F}_m$  and show that they constitute a transformation hierarchy. Next, we establish some lower bounds on the hierarchy classes of some architectures in this hierarchy. To prove that a certain set in a hierarchy is a lower bound on a hierarchy class, it suffices to show a transformation that is not in the set, yet is in the complexity class in question.

We start off by defining a class of transformations parameterized by a positive integer. For the sake of exposition, we will start by defining a *First-order Transformation* which we will then generalize to a  $k^{th}$ -order Transformation. We will then show that for all  $j \geq i$  every  $i^{th}$  order transformation is also a  $j^{th}$  order transformation.

Intuitively, a first-order transformation has the flavor of an SRM<sub>0</sub> neuron model [30], in that each synapse has a “kernel” function such that effects of inputs spikes according to this kernel are summed over all input spikes across all synapses. The transformation prescribes



an output spike if and only if this sum equals a certain “threshold”, which is a positive number.

**Definition 4** (First-order Transformation) A transformation  $\mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$  is said to be a *First-order Transformation* if there exists a  $\tau \in \mathbb{R}^+$  and functions  $f_j : \mathbb{R} \rightarrow \mathbb{R}$ , for  $1 \leq j \leq m$ , so that for every  $\chi \in \mathcal{F}_m$  and  $t \in \mathbb{R}$ , we have  $\Xi_t \mathcal{T}(\chi) = \langle t \rangle$  if and only if we have  $\sum_{j=1}^m \sum_{i=1}^{l_j} f_j(x_j^i) = \tau$ , where  $\Xi_{[0,\infty)} \sigma_t(\chi) = \langle \vec{x}_1, \dots, \vec{x}_m \rangle$  with  $\vec{x}_j = \langle x_j^1, x_j^2, \dots, x_j^{l_j} \rangle$ , for  $1 \leq j \leq m$ .

Informally, a  $k^{th}$ -order transformation is a generalization of a first-order transformation with higher-dimensional kernel functions. Thus, a second-order transformation, for example, has functions that take every pair of spikes and add up their “effects”, in addition to “first-order” effects.

**Definition 5** ( $k^{th}$ -order Transformation) A transformation  $\mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$  is said to be a  $k^{th}$ -order Transformation if there exists a  $\tau \in \mathbb{R}^+$  and functions  $f_{j_1} : \mathbb{R} \rightarrow \mathbb{R}$ ,  $f_{j_1 j_2} : \mathbb{R}^2 \rightarrow \mathbb{R}$ ,  $\dots$ ,  $f_{j_1 j_2 \dots j_k} : \mathbb{R}^k \rightarrow \mathbb{R}$ , with  $1 \leq j_p \leq m$ , where  $1 \leq p \leq k$ , so that for every  $\chi \in \mathcal{F}_m$  and  $t \in \mathbb{R}$ , we have  $\Xi_t \mathcal{T}(\chi) = \langle t \rangle$  if and only if we have  $\sum_{j_1=1}^m \sum_{i_1=1}^{l_{j_1}} f_{j_1}(x_{j_1}^{i_1}) + \sum_{j_1=1}^m \sum_{j_2=1}^m \sum_{i_1=1}^{l_{j_1}} \sum_{i_2=1}^{l_{j_2}} f_{j_1 j_2}(x_{j_1}^{i_1}, x_{j_2}^{i_2}) + \dots + \sum_{j_1=1}^m \dots \sum_{j_k=1}^m \sum_{i_1=1}^{l_{j_1}} \dots \sum_{i_k=1}^{l_{j_k}} f_{j_1 \dots j_k}(x_{j_1}^{i_1}, \dots, x_{j_k}^{i_k}) = \tau$ , where  $\Xi_{[0,\infty)} \sigma_t(\chi) = \langle \vec{x}_1, \dots, \vec{x}_m \rangle$  with  $\vec{x}_j = \langle x_j^1, x_j^2, \dots, x_j^{l_j} \rangle$ , for  $1 \leq j \leq m$ .

For all  $j \geq i$ , every  $i^{th}$ -order transformation is also a  $j^{th}$ -order transformation. This can be seen by setting the value of all the functions whose domain has dimensionality greater than  $i$  to be zero everywhere. Therefore, for all  $j \geq i$ , the set of all  $i^{th}$ -order transformations is a subset of the set of all  $j^{th}$ -order transformations. This naturally induces a transformation hierarchy in  $\mathfrak{F}_m$

**Proposition 1** For  $k \in \mathbb{Z}^+$ , let  $\mathcal{O}_m^k \subseteq \mathfrak{F}_m$  be the set of all  $k^{th}$ -order transformations of the form  $\mathcal{T} : \mathcal{F}_m \rightarrow \mathcal{S}$ . Then,  $\mathcal{O}_m = \langle \mathcal{O}_m^1, \mathcal{O}_m^2, \dots, \mathfrak{F}_m \rangle$  is a transformation hierarchy in  $\mathfrak{F}_m$ .

For certain acyclic network architectures, we now establish some lower bounds on their hierarchy classes in the above-mentioned transformation hierarchy.

**Theorem 1** *Let  $\Sigma_2$  be the set the set of all networks with the architecture of the network in Figure 5. Then  $\mathcal{O}_2^1$  is a lower bound on the hierarchy class of  $\Sigma_2$  in  $\mathcal{O}_2$ .*

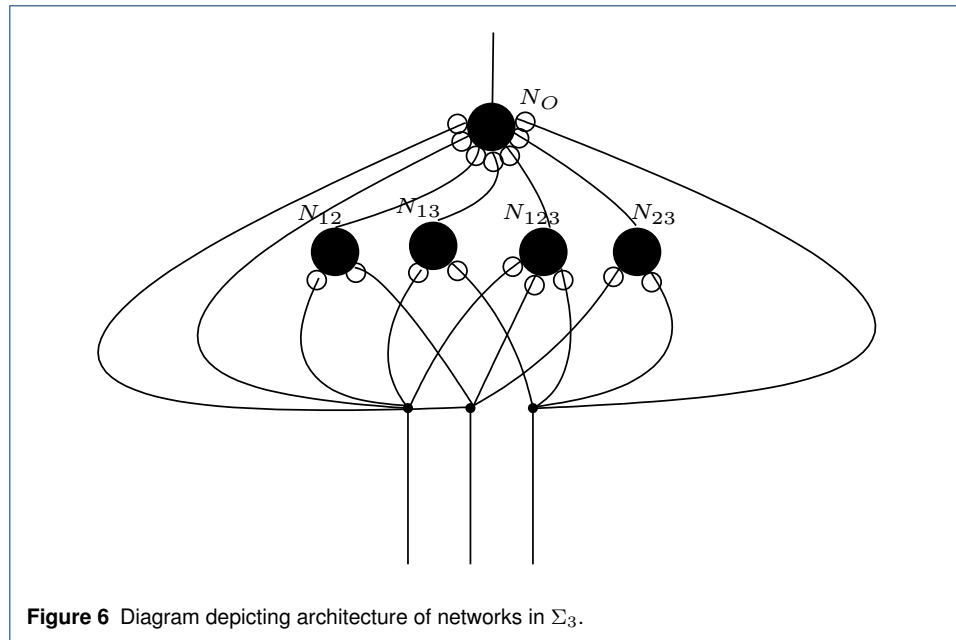
*Proof* We prove that  $\mathcal{O}_2^1$  is a lower bound on the hierarchy class of  $\Sigma_2$  in  $\mathcal{O}_2$  by showing a transformation that a network in  $\Sigma_2$  can effect, but which lies outside  $\mathcal{O}_2^1$ .

For the sake of brevity, we describe the salient responses of the neurons from which it is straightforward to construct an SRM<sub>0</sub> model for them. For the sake of contradiction assume that the transformation effected by the network is a first-order transformation. In Figure 5, neuron  $N_{12}$  is an inhibitory neuron and the neuron  $N_0$  is an excitatory neuron. Both the input spike trains provide excitatory input to both neurons. We assume, for the sake of contradiction, that the potential of the output neuron  $N_0$  can be written down as a first-order transformations. The argument is made on two input spikes that occurred  $t_1$  and  $t_2$  seconds ago in the first and second input spike train respectively. Consider the values of the functions  $f_1(t_1)$  and  $f_2(t_2)$ , where  $f_1(\cdot)$  and  $f_2(\cdot)$  are component functions of the putative first-order function in the transformation. The neuron  $N_0$  is set up so that it produces a spike now, if a spike happens either at  $t_1$  alone or  $t_2$  alone. Since the transformation is first-order, this gives us two equations  $f_1(t_1) = \tau$  and  $f_2(t_2) = \tau$ . When there is a spike both at positions  $t_1$  and  $t_2$ ,  $N_0$  would reach threshold earlier. However, the occurrence of both these spikes causes  $N_{12}$  to spike. This, in turn, causes an inhibitory effect on the membrane potential of  $N_0$ , which compensates for the extra excitation, so that it spikes exactly once, now. We therefore have the equation  $f_1(t_1) + f_2(t_2) = \tau$ . These three equations (in the two variables  $f_1(t_1)$  and  $f_2(t_2)$ ) form an inconsistent system of linear equations, and therefore  $f_1(t_1)$  and  $f_2(t_2)$  do not exist, contradicting our hypothesis. Therefore the transformation induced by the current network is not a first-order transformation. Thus,  $\mathcal{O}_2^1$  is a lower bound on the hierarchy class of  $\Sigma_2$  in  $\mathcal{O}_2$ .  $\square$

Next, we apply a similar strategy to derive a lower bound on the hierarchy class of another network architecture. It will then be clear how one can generalize the present technique.

**Theorem 2** *Let  $\Sigma_3$  be the set the set of all networks with the architecture of the network in Figure 6. Then  $\mathcal{O}_3^2$  is a lower bound on the hierarchy class of  $\Sigma_3$  in  $\mathcal{O}_3$ .*

*Proof* As before, we prove this by showing a transformation that a network in  $\Sigma_3$  can effect, but which lies outside  $\mathcal{O}_3^2$ . The argument is made on three spike positions  $t_1$ ,  $t_2$  and  $t_3$  in the past, and the values of the functions  $f_1(t_1)$ ,  $f_2(t_2)$ ,  $f_3(t_3)$ ,  $f_{12}(t_1, t_2)$ ,  $f_{23}(t_2, t_3)$  and  $f_{31}(t_3, t_1)$ , which are component functions of the putative second-order transformation. Again,  $N_0$  is set up so it spikes on each of the three individual spikes occurring alone. This gives us the equations  $f_1(t_1) = \tau$ ,  $f_2(t_2) = \tau$  and  $f_3(t_3) = \tau$ .  $N_{12}$  works exactly as in the previous example, and so do  $N_{23}$  and  $N_{31}$ , so as to make the output neuron spike whenever every pair of spikes occur. This gives us the equations  $f_1(t_1) + f_2(t_2) + f_{12}(t_1, t_2) = \tau$ ,  $f_2(t_2) + f_3(t_3) + f_{23}(t_2, t_3) = \tau$  and  $f_3(t_3) + f_1(t_1) + f_{31}(t_3, t_1) = \tau$ . Now, when spikes occur simultaneously at all three times, the inhibition provided by  $N_{12}$ ,  $N_{23}$  and  $N_{31}$  causes the membrane potential of  $N_0$  to always stay below threshold. The neuron  $N_{123}$  now provides enough excitation to  $N_0$ , in order to make it spike now. This gives us the equation  $f_1(t_1) + f_2(t_2) + f_3(t_3) + f_{12}(t_1, t_2) + f_{23}(t_2, t_3) + f_{31}(t_3, t_1) = \tau$ . It is straightforward to verify that this system of 7 equations in 6 variables is inconsistent. Therefore,  $\mathcal{O}_3^2$  is a lower bound on the hierarchy class of  $\Sigma_3$  in  $\mathcal{O}_3$ .  $\square$



**Figure 6** Diagram depicting architecture of networks in  $\Sigma_3$ .

It is straightforward to obtain similar results for higher-order transformations with this broad technique.

## 5 Discussion

This work is part of an ongoing research program to build first-principles theory for Connectomics. The idea is to have a theoretical understanding of how structure of neuronal networks constrains spike-timed computations performed by them. This could be useful in generating hypotheses about mechanistic computation in networks when one has connectomic information. Indeed, the field of (microscale) Connectomics is scaling up in its ability to produce increasingly large datasets in diverse organisms and such theory is needed in the context of such data. More fundamentally, the theory will also enable us to understand what structural properties of a network are crucial in enabling it to effect particular classes of spike-timed computations. This has potential application in Comparative Connectomics, an emerging sub-field, where the goal is to compare connectomes either of different individuals of the same species or across individuals of different species in order to determine which structural aspects of connectivity are preserved or altered.

Here, our focus has been on studying the space of spike-train to spike-train transformations. We define notions that allow us to carve up this space into nested sequences of subsets and ask if we can encapsulate the subset of transformations spanned by specific network architectures within sets in this hierarchy. Even if we cannot determine the exact hierarchy classes, establishing bounds on them would be progress. Indeed, we explicitly establish a class of such hierarchies and establish lower bounds corresponding to certain classes of network architectures with respect to said hierarchies.

There are at least two reasons for relating complexity classes to transformation hierarchies. First, complexity classes themselves seem to be hard to characterize succinctly in terms of properties of transformations they contain. Instead, we try to understand complexity classes of network architectures relative to these sets in the hierarchy which are easier to characterize using mathematical properties of transformations. The second reason for

this approach is that it provides us another – and a possibly more wholesale – way to prove complexity results via bounds on the corresponding hierarchy classes.

#### Competing interests

The author declares that he has no competing interests.

#### Author's contributions

The author performed the research and wrote the paper.

#### Acknowledgements

The author wishes to thank Arunava Banerjee for discussions. The work was supported in part by the Simons Foundation and in part by a US National Science Foundation grant (NSF IIS-0902230) to Arunava Banerjee. Part of the work appears in the author's Ph.D. dissertation at the University of Florida.

#### References

1. Ramaswamy, V., Banerjee, A.: Connectomic constraints on computation in feedforward networks of spiking neurons. *Journal of Computational Neuroscience* **37**(2), 209–228 (2014)
2. Denk, W., Horstmann, H.: Serial block-face scanning electron microscopy to reconstruct three-dimensional tissue nanostructure. *PLoS Biology* **2**(11), 329 (2004)
3. Turaga, S.C., Murray, J.F., Jain, V., Roth, F., Helmstaedter, M., Briggman, K., Denk, W., Seung, H.S.: Convolutional networks can learn to generate affinity graphs for image segmentation. *Neural Computation* **22**(2), 511–538 (2010)
4. Helmstaedter, M., Briggman, K.L., Denk, W.: High-accuracy neurite reconstruction for high-throughput neuroanatomy. *Nature neuroscience* **14**(8), 1081–1088 (2011)
5. Mikula, S., Binding, J., Denk, W.: Staining and embedding the whole mouse brain for electron microscopy. *Nature methods* **9**(12), 1198–1201 (2012)
6. Beier, T., Pape, C., Rahaman, N., Prange, T., Berg, S., Bock, D.D., Cardona, A., Knott, G.W., Plaza, S.M., Scheffer, L.K., *et al.*: Multicut brings automated neurite segmentation closer to human performance. *Nature Methods* **14**(2), 101 (2017)
7. Januszewski, M., Kornfeld, J., Li, P.H., Pope, A., Blakely, T., Lindsey, L., Maitin-Shepard, J., Tyka, M., Denk, W., Jain, V.: High-precision automated reconstruction of neurons with flood-filling networks. *Nature methods* **15**(8), 605 (2018)
8. Chklovskii, D.B., Vitaladevuni, S., Scheffer, L.K.: Semi-automated reconstruction of neural circuits using electron microscopy. *Current opinion in neurobiology* **20**(5), 667–675 (2010)
9. Seung, H.S.: Towards functional connectomics. *Nature* **471**(7337), 170–172 (2011)
10. Denk, W., Briggman, K.L., Helmstaedter, M.: Structural neurobiology: missing link to a mechanistic understanding of neural computation. *Nature Reviews Neuroscience* **13**(5), 351–358 (2012)
11. Reid, R.C.: From functional architecture to functional connectomics. *Neuron* **75**(2), 209–217 (2012)
12. Helmstaedter, M., Briggman, K.L., Turaga, S.C., Jain, V., Seung, H.S., Denk, W.: Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature* **500**(7461), 168–174 (2013)
13. Sporns, O., Tononi, G., Kötter, R.: The human connectome: a structural description of the human brain. *PLoS computational biology* **1**(4), 42 (2005)
14. White, J.G., Southgate, E., Thomson, J.N., Brenner, S.: The structure of the nervous system of the nematode *Caenorhabditis elegans*: the mind of a worm. *Phil. Trans. R. Soc. Lond* **314**, 1–340 (1986)
15. Ryan, K., Lu, Z., Meinertzhagen, I.A.: The CNS connectome of a tadpole larva of *Ciona intestinalis* (L.) highlights sidedness in the brain of a chordate sibling. *Elife* **5**, 16962 (2016)
16. Cardona, A. Personal Communication
17. Hildebrand, D.G.C., Cicconet, M., Torres, R.M., Choi, W., Quan, T.M., Moon, J., Wetzel, A.W., Champion, A.S., Graham, B.J., Randlett, O., *et al.*: Whole-brain serial-section electron microscopy in larval zebrafish. *Nature* **545**(7654), 345 (2017)
18. Zheng, Z., Lauritzen, J.S., Perlman, E., Robinson, C.G., Nichols, M., Milkie, D., Torrens, O., Price, J., Fisher, C.B., Sharifi, N., *et al.*: A complete electron microscopy volume of the brain of adult *Drosophila melanogaster*. *Cell* **174**(3), 730–743 (2018)
19. Bock, D.D., Lee, W.-C.A., Kerlin, A.M., Andermann, M.L., Hood, G., Wetzel, A.W., Yurgenson, S., Soucy, E.R., Kim, H.S., Reid, R.C.: Network anatomy and in vivo physiology of visual cortical neurons. *Nature* **471**(7337), 177–182 (2011)
20. Briggman, K.L., Helmstaedter, M., Denk, W.: Wiring specificity in the direction-selectivity circuit of the retina. *Nature* **471**(7337), 183–188 (2011)
21. Takemura, S.-y., Bharioke, A., Lu, Z., Nern, A., Vitaladevuni, S., Rivlin, P.K., Katz, W.T., Olbris, D.J., Plaza, S.M., Winston, P., *et al.*: A visual motion detection circuit suggested by *Drosophila* connectomics. *Nature* **500**(7461), 175–181 (2013)
22. Kim, J.S., Greene, M.J., Zlateski, A., Lee, K., Richardson, M., Turaga, S.C., Purcaro, M., Balkam, M., Robinson, A., Behabadi, B.F., *et al.*: Space–time wiring specificity supports direction selectivity in the retina. *Nature* **509**(7500), 331 (2014)
23. Kasthuri, N., Hayworth, K.J., Berger, D.R., Schalek, R.L., Conchello, J.A., Knowles-Barley, S., Lee, D., Vázquez-Reina, A., Kaynig, V., Jones, T.R., *et al.*: Saturated reconstruction of a volume of neocortex. *Cell* **162**(3), 648–661 (2015)
24. Berck, M.E., Khandelwal, A., Claus, L., Hernandez-Nunez, L., Si, G., Tabone, C.J., Li, F., Truman, J.W., Fetter, R.D., Louis, M., *et al.*: The wiring diagram of a glomerular olfactory system. *Elife* **5**, 14859 (2016)
25. Larderet, I., Fritsch, P.M., Gendre, N., Neagu-Maier, G.L., Fetter, R.D., Schneider-Mizell, C.M., Truman, J.W., Zlatić, M., Cardona, A., Sprecher, S.G.: Organization of the *Drosophila* larval visual circuit. *Elife* **6**, 28387 (2017)

26. Gerhard, S., Andrade, I., Fetter, R.D., Cardona, A., Schneider-Mizell, C.M.: Conserved neural circuit structure across drosophila larval development revealed by comparative connectomics. *Elife* **6**, 29089 (2017)
27. Eichler, K., Li, F., Litwin-Kumar, A., Park, Y., Andrade, I., Schneider-Mizell, C.M., Saumweber, T., Huser, A., Eschbach, C., Gerber, B., *et al.*: The complete connectome of a learning and memory centre in an insect brain. *Nature* **548**(7666), 175 (2017)
28. Felsenberg, J., Jacob, P.F., Walker, T., Barnstedt, O., Edmondson-Stait, A.J., Pleijzier, M.W., Otto, N., Schlegel, P., Sharifi, N., Perisse, E., *et al.*: Integration of parallel opposing memories underlies memory extinction. *Cell* **175**(3), 709–722 (2018)
29. Morgan, J.L., Lichtman, J.W.: Why not connectomics? *Nature methods* **10**(6), 494–500 (2013)
30. Gerstner, W., Kistler, W.M.: *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press, New York, NY (2002)