The hourglass organization of the *C. elegans* connectome

Kaeser M. Sabrin[1], Yongbin Wei[2], Martijn van den Heuvel[2], and Constantine Dovrolis[1]

[1]School of Computer Science, Georgia Institute of Technology
[2]Center for Neurogenomics and Cognitive Research, Vrije Universiteit Amsterdam

April 5, 2019

Abstract

We approach the *C. elegans* connectome as an information processing network that receives input from about 90 sensory neurons, processes that information through a highly recurrent network of about 80 interneurons, and it produces a coordinated output from almost 120 motor neurons that control the nematode's muscles. We apply a recently developed network analysis framework ("hourglass effect") [1], and focus on the feedforward flow of information from sensory neurons to motor neurons. The analysis reveals that this feedforward flow traverses a small core ("hourglass waist") that consists of 10-15 interneurons – mostly the same set of neurons that were previously shown to constitute the "rich-club" of the *C. elegans* connectome [2]. This result is robust to the methodology that separates the feedforward from the feedback flow of information. The set of core interneurons remains mostly the same when we consider only chemical synapses or the combination of chemical synapses and gap junctions. The hourglass organization of the connectome suggests that *C. elegans* has some similarities with encoder-decoder artificial neural networks in which the input vector is first compressed to a low-dimensional latent space that encodes the input data in a more efficient manner, followed by a decoding network through which intermediate-level salient functions are combined in different ways to compute the correlated outputs of the network.

1 Introduction

Natural, technological and information-processing complex systems are often hierarchically modular [3, 4, 5, 6]. A modular system consists of smaller sub-systems (modules) that, at least in the ideal case, can function independently of whether or how they are connected to other modules: each module receives inputs from the environment or from other modules to perform a certain function [7, 8, 9]. Modular systems are often also hierarchical, meaning that simpler modules are embedded in, or reused by, modules of higher complexity [10, 11, 12, 13]. It has been shown that both modularity and hierarchy can emerge naturally as long as there is an underlying cost for the connections between different system units [14, 15].

In the technological world, modularity and hierarchy are often viewed as essential principles that provide benefits in terms of design effort (compared to "flat" or "monolithic" designs in which the entire system is a single module), development cost (design a module once, reuse it many times), and agility (upgrade, modify or replace modules without affecting the entire system) [16, 17, 18]. In the natural world, the benefits of modularity and hierarchy are often viewed in terms of evolvability (the ability to adapt and develop novel features can be accomplished with minor modifications in how existing modules are interconnected) [19, 20, 21] and robustness (the ability to maintain a certain function even when there are internal or external perturbations can be accomplished using available modules in different ways) [22, 23, 24].

It has been observed across several disciplines that hierarchically modular systems are often structured in a way that resembles a *bow-tie or hourglass* (depending on whether that structure is viewed horizontally or vertically) [25, 1]. This structure enables the system to generate many outputs from many inputs through a relatively small number of intermediate modules, referred to

as the "knot" of the bow-tie or the "waist" of the hourglass.[1] This "hourglass effect" has been observed in systems of embryogenesis [26, 27], metabolism [28, 29, 30], immunology [31, 32], signaling networks [33], vision and cognition [34, 35], deep neural networks [36], computer networking [37], manufacturing [38], as well as in the context of general core-periphery complex networks [39, 40]. The few intermediate modules in the hourglass waist are critical for the operation of the entire system, and so they are also more conserved during the evolution of the system compared to modules that are closer to inputs or outputs [37, 41, 42].

In earlier studies, the hourglass effect is defined in terms of the number of nodes at each layer, and a network is referred to as an hourglass if the width of the intermediate layers is much smaller relative to the width of the input and output layers [37, 43, 25]. For instance, [25] showed that in the special case of a linear input-output mapping, a layered and directed network can evolve to an hourglass structure if the relation between inputs and outputs can be represented with a rank-deficient matrix.

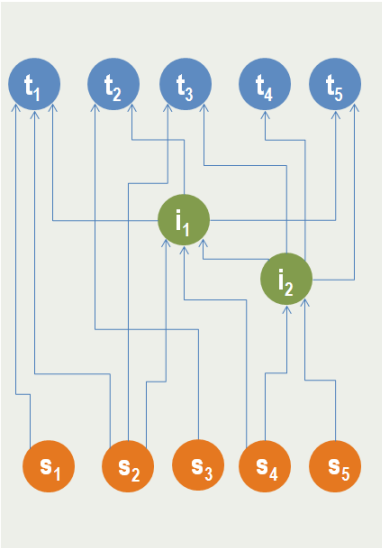
In this paper, we apply the hourglass analysis framework of [1] on the *C. elegans* connectome [44]. The *C. elegans* connectome can be thought of as an information processing network that transforms stimuli received by the environment, through sensory neurons, into coordinated bodily activities (such as locomotion) controlled by motor neurons [44]. Between the sensory and motor neurons, there is a highly recurrent network of interneurons that gradually transforms the input information to output motor activity. An important challenge in applying the hourglass analysis framework on *C. elegans* is that the former assumes that the network from a given set of input nodes (sources) to a given set of output nodes (targets) is a Directed Acyclic Graph (DAG). On the contrary, the *C. elegans* connectome includes many nested feedback loops between all three types of neurons. For this reason, we extend the analysis of [1] in networks that may include cycles as long as we are given a set of sources and a set of targets. The key idea is to identify the set of *feedforward paths* from each source towards targets, and to apply the hourglass analysis framework on the union of such paths, across all sources.

Our first result is that this connectome also exhibits the hourglass effect. This result is robust to the "routing methodology" that separates the feedforward from the feedback flow of information. Further, we observe the hourglass architecture when we consider just chemical synapses, or the combination of the latter with gap junctions. On the contrary, appropriately randomized networks do not exhibit the hourglass property. We also identify the neurons at the "waist" of the hourglass. Interestingly, they are mostly the same set of neurons that were previously shown to constitute the "rich-club" of the *C. elegans* connectome [2].

Finally, we explain the benefits of the hourglass architecture, in the context of neural information processing systems, using an encoder-decoder model that resembles recent architectures in artificial neural networks [45, 36]. The encoding component compresses the redundant stimuli provided by the sensory neurons into a low-dimensional latent feature space (represented by the core neurons at the hourglass waist) that encodes the source information in a more efficient manner, given the desired input-output mapping. Then, the decoding component of the network combines those latent features, which represent intermediate-level salient functions, in different ways to drive each output through the motor neurons. To illustrate this idea, consider the toy-example of of five binary sources and five targets in Figure 1.

The *C. elegans* literature has been trying to associate specific functions to important interneurons or to associate each interneuron with a specific functional circuit. The hourglass architecture suggests a different way to think about, analyze, and experiment with a neural system: the interneurons between sensory and motor neurons can be thought of as forming an *encoder-decoder network* that reduces the intrinsic dimensionality of the system and computes few intermediate-level functions that are commonly used across different functional/behavioral circuits. This is similar to the *information bottleneck* framework, in which given a joint probability distribution between an input vector X and an output vector Y, the goal is to compute an optimal intermediate-level representation T that is both compact and able to predict Y [46, 47].

[1] The two terms, bow-tie and hourglass, have not been always viewed as synonymous in the network science literature. In particular, the term bow-tie has been applied even to networks for which the knot includes a large fraction of the network's nodes.



$$
\begin{aligned}
t_1 &= s_1 \oplus s_2 \oplus (s_2 + s_4 + (s_4 \oplus s_5)) = s_1 \oplus s_2 \oplus i_1 \\
t_2 &= s_3 + s_2 + s_4 + (s_4 \oplus s_5) = s_3 + i_1 \\
t_3 &= s_2 \cdot (s_4 \oplus s_5) = s_2 \cdot i_2 \\
t_4 &= \overline{s_4 \oplus s_5} = \overline{i_2} \\
t_5 &= s_2 + s_4 + (s_4 \oplus s_5) + (s_4 \oplus s_5) = i_1 + i_2 \\
i_1 &= s_2 + s_4 + i_2 \\
i_2 &= s_4 \oplus s_5 \\
s_1 &= a \oplus c \\
s_2 &= b \cdot c \\
s_3 &= a \\
s_4 &= a + b \\
s_5 &= b
\end{aligned}
$$

Figure 1: A hypothetical boolean system with five sources and five targets. The sources are represented by orange nodes while the targets by blue nodes. Each target is a logic function of the sources. The sources are correlated, as shown by their logical expressions. A direct source-to-target computation would require 18 boolean operations. Instead, we can compute the targets with only 9 operations if we first compute the two intermediate green nodes shown (3 operations) and then reuse those nodes to compute the targets (6 operations). This cost reduction is possible because there are correlations between the target functions. The two intermediate nodes, which represent the hourglass waist in this example, compress the information provided by the sources, computing sub-functions that are re-used at least twice in the targets. In this example the encoding part of the network is the set of connections between sources and intermediate nodes, while the decoding part is the set of connections between intermediate nodes and targets. In general, the encoder and decoder components can include additional nodes, creating a deeper hourglass architecture.

2 Methods

2.1 Connectome

The dataset we analyze describes the neural network of the hermaphrodite *C. elegans*, as reported in [44]. This connectome is a directed network between 279 neurons (the 282 non-pharyngeal neurons excluding VC6 and CANL/R, which are missing connectivity data). Neurons can be connected with two types of connections: chemical synapses and gap junctions (or, electrical synapses). The former are typically slower but stronger connections, and they transfer information only in one direction. The latter can be thought of as bi-directional connections.

The *synaptic network* (i.e., the network formed by only chemical synapses) consists of 2194 neural connections, created by 6393 chemical synapses. The *weight* of a connection is defined as the number of chemical synapses between the corresponding pair of neurons. The *in-strength* or *out-strength* of a neuron is defined as the sum of connection weights entering or leaving that neuron, respectively.

The *complete network* includes both chemical synapses and gap junctions. There are 514 pairs of neurons connected through gap junctions, creating the same number of bi-directional connections between those neurons. Unless mentioned otherwise, we analyze the synaptic network. In section (3.7), we extend the analysis to consider the complete network, asking whether there are any major differences when we also consider gap junctions.

The *C. elegans* neurons can be classified as sensory (S), inter (I) and motor (M) neurons, based on their structure and function [48]. Sensory neurons transfer information from the external environment to the central nervous system (CNS). Motor neurons transfer information from the CNS to effector organs (e.g. glands or muscles). Interneurons process information within the CNS. The *C. elegans* connectome has 88 sensory neurons, 87 interneurons and 119 motor neurons. Some of these neurons however have a dual role: ten behave as S and M, two as S and I, and three as M and I. In our analysis, we consider the S-M and S-I rual-role neurons as sensory, and the M-I neurons as motor. Consequently, the final network consists of 88 sensory neurons, 82 interneurons, and 109 motor neurons.

We can think of *C. elegans* as an information processing system in which the *feedforward flow* of information, from sensory to motor neurons, transfers sensory cues from the environment to the CNS, processes those signals to extract actionable information, which is then used to drive the behavior/motion of the organism. This feedforward flow however is regulated by multiple feedback loops that transfer information in the opposite direction, as well as lateral connections between neurons of the same type.

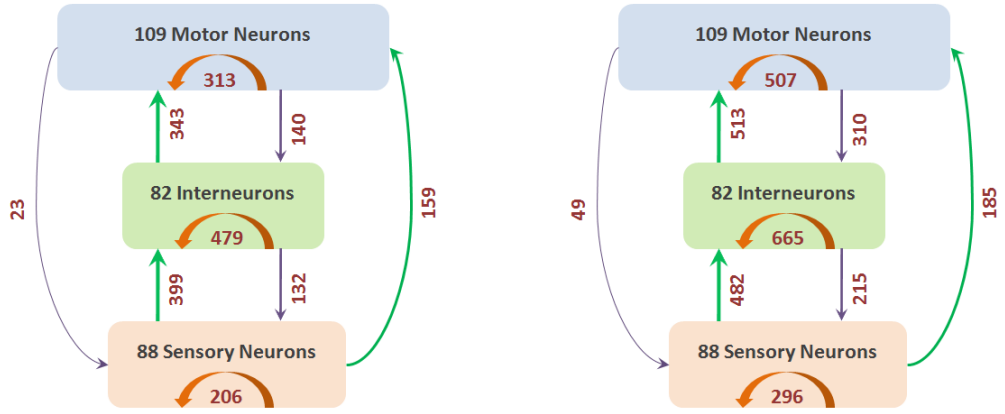
The connections that we refer to as *feedforward (FF)* are those from S to I, I to M, and S to M neurons. In the opposite direction (i.e., from I to S, M to I, and M to S neurons) the connections are referred to as *feedback (FB)*. Connections between neurons of the same type (i.e., S to S, I to I, and M to M neurons) are referred to as *lateral (LT)*. In the synaptic network, there are 901 FF connections, 998 LT connections, and 295 FB connections. Figure 2 shows the breakdown of these connection types in the synaptic and complete networks.

As shown in Figure 3(a), the FB connection weights are often lower than FF and LT weights. Also, when considering neuron pairs that are reciprocally connected with both FF and FB connections, it is more likely that the FF connection is stronger than the corresponding FB connection (see Figure 3(b)). These observations suggest that the distinction between FF and FB connections has some neurophysiological significance.

If we focus on the top-100 stronger connections, relative to all chemical connections, Table 1 shows that this set is dominated by feedforward S-to-I and I-to-M connections, as well as by lateral connections between I neurons and M neurons. None of the top-100 connections is of the feedback type. This observation suggests that feedback connections are weaker – one reason may be that they are involved mostly with the control of feedforward circuits, acting as modulators rather than drivers.

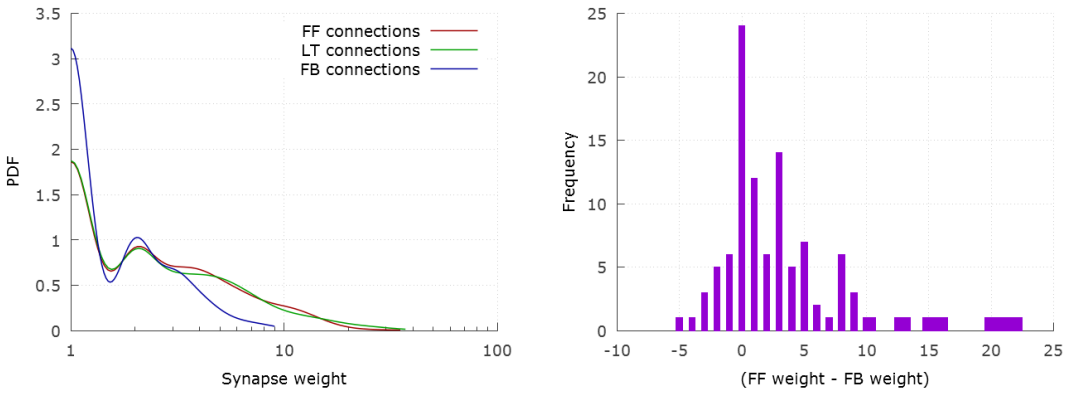
2.2 Feedforward Paths from Sensory to Motor Neurons

The "routing problem" in a communication network refers to the selection of an efficient path, or multiple paths, from each source node to each target. In neural networks, there is no established



(a) Synaptic network (b) Complete network

Figure 2: Neurons separated into three classes (S, I and M) based on structure and function. The number of connections between these three classes (and within each class) are shown with arrows (green for FF, orange for LT and purple for FB).

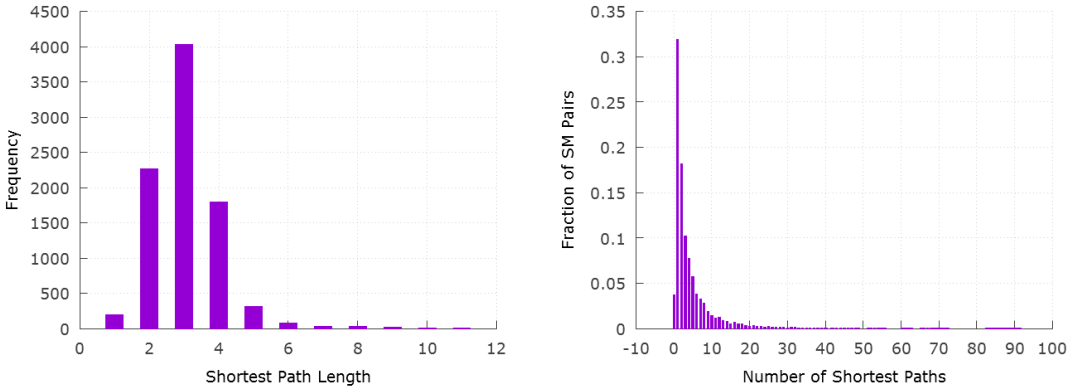


(a) (b)

Figure 3: Left: Weight distribution of FF, LT and FB connections. Right: Considering only pairs of neurons with reciprocal FF and FB connections, this histogram shows the difference of the FF weight minus the FB weight.

Connection type	Connection sub-type	Fraction in: entire connectome	Fraction in: top-100 weighted connections
FF	S-I	0.18	0.35
	S-M	0.08	0.02
	I-M	0.16	0.15
LT	S-S	0.09	0.00
	I-I	0.22	0.15
	M-M	0.14	0.33
FB	I-S	0.06	0.00
	M-I	0.06	0.00
	M-S	0.01	0.00

Table 1: The top-100 strongest connections are dominated by FF connections from S neurons to I or M neurons, and by LT connections between I neurons and M neurons. On the other hand, none of the FB connections appear in this set.



(a) (b)

Figure 4: Left: The length distribution for all shortest paths from sensory to motor neurons. Almost all shortest paths are shorter than 6 hops. Right: Distribution of the number of distinct shortest paths from a sensory neuron to a motor neuron. For about 50% of S-M pairs, there are more than two shortest paths.

"routing algorithm" that can accurately describe or model how information propagates from a sensory neuron to a motor neuron. Whether a neuron will fire or not depends on how many of its pre-synaptic neurons fire, the timing of those events, the physical size and location of the synapses in the dendritic tree, and several other factors. There are some first principles, however, that we can rely on to identify plausible routing schemes [49, 50]. These schemes should be viewed only as phenomenological models – we do not claim that neurons actually choose activation paths based on the following algorithms.

First, neurons cannot form routes based on information about the complete network or through coordination with all neurons (such as the routing algorithms used by the Internet or other technological systems). Instead, whether a neuron fires or not should be a function of only locally available information. So, we cannot expect that neural circuits use optimal routes that minimize the path length ("shortest path routing") or other path-level cost functions [51].

Second, evolution has most likely selected routing schemes that result in efficient (even though not necessarily optimal) neural communication. Consequently, we can reject routing schemes that exploit all possible paths between two neurons as many of those paths would be inefficient.

Third, for robustness and resilience reasons, it is likely that multiple paths are used to transfer information from each sensory neuron to a motor neuron – schemes that only select one path would be too fragile.

Fourth, given the low firing reliability of neurons, it is unlikely that a sensory neuron can communicate effectively with a motor neuron through multiple intermediate neurons. There should be a limit on the length of any plausible neural path [52].

Putting the previous four principles together, we are led to the following hypothesis: a sensory neuron S communicates with a motor neuron T through multiple paths that may be suboptimal but not much longer than the shortest path length from S to T.

Given this broad hypothesis, we identify several plausible routing schemes – and then examine whether our results are robust to the selection of a specific routing scheme.

To help choose reasonable parameter values for the various routing schemes we consider, we first examine the length and number of shortest paths from each sensory neurons S to each motor neuron M. Figure 4(a) shows the distribution of the length of these paths, measured in "hops" (i.e., connections between neurons). Almost all shortest paths from S to M neurons are between 2-4 hops. So, if the shortest connection from a sensory to a motor neuron is say 3 hops, the second and fourth principles suggest that we may also consider slightly longer paths, say 4 or 5 hops long.

Figure 4(b) shows the likelihood that an (S,M) pair is connected through x shortest paths. Note that only 4% of (S,M) pairs are not connected by any path, about 32% of (S,M) pairs are connected through only one shortest path, while the rest are connected with multiple shortest

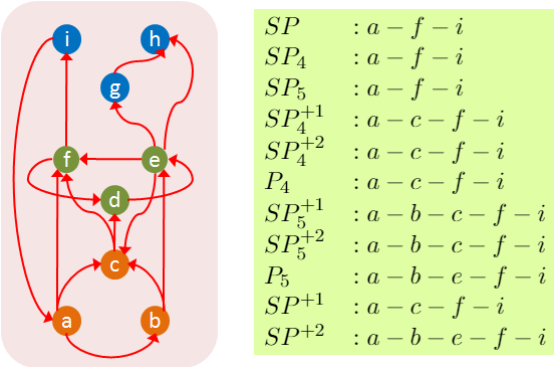


Figure 5: A representative path for each routing scheme when the source is a and the target is i. The orange nodes represent sensory neurons, the green nodes interneurons, and the blue nodes motor neurons.

paths.

The various routing schemes we consider in the rest of the paper are:

1. "SP": As a reference point, SP refers to the selection of only shortest paths from a sensory neuron s to a motor neuron t.
2. "SP_4": The subset of SP including paths that are at most 4 hops.
3. "SP_5": The subset of SP including paths that are at most 5 hops.
4. "SP^{+1}": The paths in SP together with all paths that are one hop longer than the shortest path from s to t.
5. "SP^{+2}": The paths in SP together with all paths that are one or two hops longer than the shortest path from s to t.
6. "SP_4^{+1}": The subset of SP^{+1} including paths that are at most 4 hops.
7. "SP_5^{+1}": The subset of SP^{+1} including paths that are at most 5 hops.
8. "SP_4^{+2}": The subset of SP^{+2} including paths that are at most 4 hops.
9. "SP_5^{+2}": The subset of SP^{+2} including paths that are at most 5 hops.
10. "P_4": All paths from s to t that are at most 4 hops long.
11. "P_5": All paths from s to t that are at most 5 hops long.

The last two routing schemes (P_4 and P_5) are not variations of shortest path but they are based on the notion of diffusion-based routing. In the latter, information propagates from a source towards a sink selecting among all possible connections either randomly (e.g., random-walk based models) [53] or based on a threshold function (e.g., a neuron fires if at least a certain function of its pre-synaptic neurons fire) [54].

2.3 Path Centrality Metric and τ-Core Selection

After utilizing one of the previous routing schemes to compute all paths from a sensory neuron to a motor neuron, we analyze these "source-target" paths based on the *hourglass framework*, developed in [1]. The objective of this analysis is to examine whether there is a small set of nodes through which almost all source-target paths go through. In other words, the hourglass analysis examines whether there is a small set of neurons that forms a bottleneck in the flow of information from sensory neurons towards motor neurons.

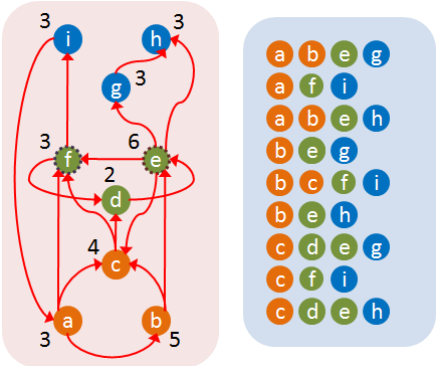


Figure 6: The path centrality of each node (shown at the left) based on the set of shortest paths SP (shown at the right). For τ=90%, a possible core is the two neurons $\{e, f\}$.

The path centrality $P(v)$ of a node v is defined as the number of source-target paths that traverse v. This metric has been also referred to as the *stress* of a node [55]. Figure 6 illustrates the path centrality of each node in a small network – just for this example, the paths have been computed based on the shortest path (SP) routing algorithm. Any other routing scheme could have been used instead.

The path centrality metric is more general than betweenness or closeness centrality that are only applicable to shortest paths. Katz centrality does not distinguish between terminal and intermediate nodes and it penalizes longer paths. Metrics such as degree, strength, PageRank or eigenvector centrality are heavily dependent on the local connectivity of nodes rather than on the paths that traverse each node.

Given a set of source-target paths, the next step of the analysis is to compute the τ-*core*, i.e., the smallest subset of nodes that can collectively cover a fraction τ of the given set of paths. The fraction τ is referred to as the *path coverage threshold* and it is meant to ignore a small fraction of paths that may be incorrect or invalid. Computing the τ-Core is an NP-Complete problem [1], and so we solve it with the following greedy heuristic (see [1] for an approximation bound):

- Initially, the core set is empty.
- In each iteration:
  1. Compute the path centrality of all remaining nodes.
  2. Include the node with maximum path centrality in the core set and remove all paths that traverse this node from the given set of paths.
- The algorithm terminates when we have covered at least a fraction τ of the given set of paths.

Figure 6 illustrates the core of a small network based on the shortest path routing mechanism, for τ=90%.

2.4 Hourglass Score

Informally, the hourglass property of a network can be defined as having a small core, even when the path coverage threshold τ is close to one. To make the previous definition more precise, we can compare the core size $C(\tau)$ of the given network $\mathbf{G}$ with the core size of a derived network that maintains the same set source-target dependencies of $\mathbf{G}$ but that is not an hourglass by construction.

To do so, we create a *flat dependency network* $\mathbf{G_f}$ from $\mathbf{G}$ as follows:

1. $\mathbf{G_f}$ has the same set of source and target nodes as $\mathbf{G}$ but it does not have any intermediate nodes.

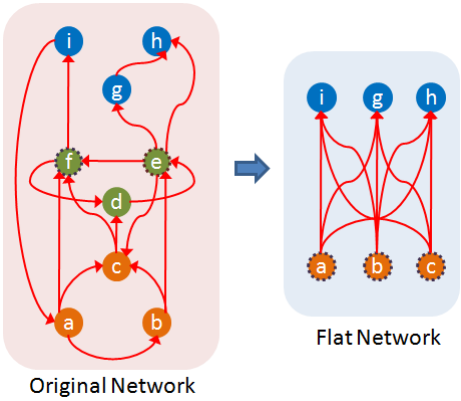


Figure 7: When the path coverage threshold is τ=90%, a core for the original network (left) is the set $\{e, f\}$. The weight of a connection in the flat network (right) represents the number of ST-paths between the corresponding source-target pair in the original network. All connections of this flat network have unit weight. The core of the flat network for the same τ consists of three nodes ($\{a, b, c\}$ or ($\{i, , g, h\}$). The H-score of the original network is $1 - \frac{2}{3} = 0.33$.

2. For every ST-path from a source s to a target t in $\mathbf{G}$, we add a direct connection from s to t in $\mathbf{G_f}$. If there are w connections from s to t in $\mathbf{G_f}$, they can be replaced with a single connection of weight w.

Note that $\mathbf{G_f}$ preserves the source-target dependencies of $\mathbf{G}$: each target in $\mathbf{G_f}$ is constructed based on the same set of "source ingredients" as in $\mathbf{G}$. Additionally, the number of ST-paths in the original dependency network is equal to the number of paths in the weighted flat network (a connection of weight w counts as w paths). However, the paths in $\mathbf{G_f}$ are direct, without forming any intermediate modules that could be reused across different targets. So, by construction, the flat network $\mathbf{G_f}$ cannot have the hourglass property.

Suppose that $C_f(\tau)$ represents the core size of the flat network $\mathbf{G_f}$. The core of $\mathbf{G_f}$ can include a combination of sources and targets, and it cannot be larger than either the set of sources or targets. Additionally, the core of the flat network is larger or equal than the core of the original network (because the core of the flat network also covers at least a fraction τ of the ST-paths of the original network – but the core of the original network may be smaller because it can also include intermediate nodes together with sources or targets). So,

$$C(\tau) \leq C_f(\tau) \leq \min\{S, T\} \tag{1}$$

To quantify the extent at which $\mathbf{G}$ exhibits the hourglass effect, we define the *Hourglass Score*, or *H-score*, as follows:

$$H(\tau) = 1 - \frac{C(\tau)}{C_f(\tau)} \tag{2}$$

Clearly, $0 \leq H(\tau) < 1$. The H-score of G is approximately one if the core size of the original network is negligible compared to the the core size of the corresponding flat network. Figure 7 illustrates the definition of this metric.

An ideal hourglass-like network would have a single intermediate node that is traversed by every single ST-path (i.e., $C(1)$=1), and a large number of sources and targets none of which originates or terminates, respectively, a large fraction of ST-paths (i.e., a large value of $C_f(1)$). The H-score of this network would be approximately equal to one.

2.5 Randomization Method

We examine the statistical significance of the observed hourglass score in a given network G using an ensemble of randomized networks $\{G_r\}$. The latter are constructed so that they preserve some key properties of G: the number of nodes and connections, the in-degree of each node, and the

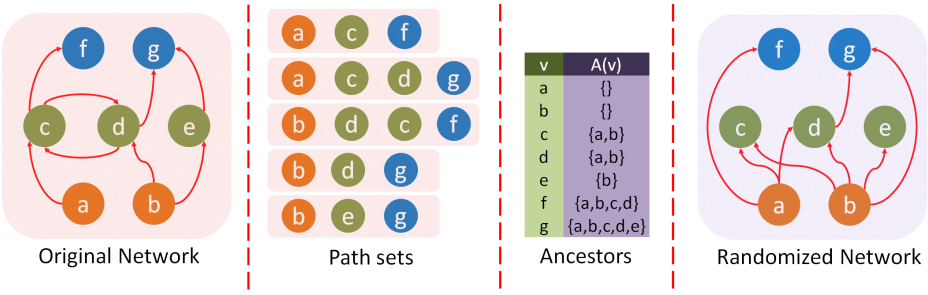


Figure 8: For the network and source-target paths given in the left two figures, we first compute the ancestors $A(v)$ of each node v (shown in the third figure from left). In the randomized network (shown at the right), we preserve the in-degree of each node v and randomly select incoming connections from the set $A(v)$.

partial ordering between nodes (explained next). The randomization reassigns connections between pairs of nodes and changes the out-degree of nodes, as described below.

Suppose we are given G and a set of paths P from sources to targets. If there is a path in which node v appears after node u *and* there is no path in which u appears after v, we say that u is an ancestor of v and write $u \in A(v)$. For a pair of nodes (u, v), we can have one of the following cases: (1) u is an ancestor on v, (2) v is an ancestor of u, (3) both u and v depend on each other, and (4) u and v do not depend on each other. We aim to preserve the partial ordering of nodes, as follows:

1. if u is *not* an ancestor of v in G, then it cannot be that u becomes an ancestor of v in a randomized network,
2. the set of ancestors of v in a randomized network is a subset of the set of ancestors $A(v)$ in G.

The construction of randomization networks proceeds as follows: for each node v in the original network, we first remove all incoming connections. We then randomly pick in-degree(v) distinct nodes from $A(v)$ and add connections from them to v. If in-degree$(v) > |A(v)|$ then we add in-degree$(v) - |A(v)|$ additional connections ("multi-connections") from randomly selected nodes in $A(v)$ to v. The randomization mechanism is illustrated in Figure 8. It should be mentioned that there are several other randomization methods, preserving different network features [56]. None of them however preserve the partial ordering between nodes, which is an essential feature of a network in which a set of input-output dependency paths captures how information flows from sources to targets.

2.6 Location Metric

We also associate a *location* with each node to capture its relative position in the feedforward network between sources and targets. One way to place intermediate nodes between sources and targets is to consider the number of paths $P_S(v)$ from sources (excluding v if it is a source itself) to v as a proxy for v's *complexity* and the number of paths $P_T(v)$ from v to targets (excluding v if it is a target itself) as a proxy for v's *generality*. Nodes with zero in-degree (which cover most sources) have the lowest complexity value (equal to 0), while nodes with zero out-degree (which cover most targets) have the lowest generality value (equal to 0). The following equation defines a location metric based on $P_S(v)$ and $P_T(v)$,

$$L(v) = \frac{P_S(v)}{P_S(v) + P_T(v)} \tag{3}$$

$L(v)$ varies between 0 (for zero in-degree sources) and 1 (for zero out-degree targets). If there is a small number of paths from sources to a node v (low complexity) but a large number of paths from v to targets (high complexity), v's role in the network is more similar to sources than targets, and so its location should be closer to 0 than 1. The opposite is true for nodes that have high complexity but low generality.

2.7 Encoder-Decoder Architecture

Returning to the illustration of Figure-1, the number of paths from the set of sources S to a specific target t is denoted by $P_S(t)$, and it is equal to the number of source literals in the mathematical expression for t. If a Boolean expression involves n literals, it requires $n-1$ Boolean operations. So, $P_S(t)$ can be thought of as the *cost* for computing t from S.

More generally, even if a feedforward network does not represent Boolean expressions, we can think of the number of paths in $P_S(t)$ as a cost metric for "computing" the target t from the set of sources S: the larger $P_S(t)$ is, the more ways exist in which the information provided by the set of sources S affects the function of t.

Informally, an hourglass architecture is a network in which the information provided by a large set of sources S is first encoded (or compressed) into a small set Z of intermediate nodes at the "waist" of the hourglass. Then, the functions provided by the nodes in Z are decoded in computing the targets in T. Additionally, in an hourglass architecture there should be relatively few paths that bypass Z.

The question we focus on here is: *how does an hourglass architecture decrease the cost of computing a set of targets T from a set of sources S, and how large is that decrease in the case of C. elegans?*

Let $C_S(T)$ be the cumulative cost for computing the set of targets T from the set of sources S:

$$C_S(T) = \sum_{t \in T} P_S(t) \tag{4}$$

Given a set of intermediate nodes Z, we can produce the targets T in a two-step process: first, compute each node in Z from the sources S, and then compute each target in T from the set of intermediate nodes Z. There may be some source-to-target paths however that bypass the nodes in Z – we need to consider the cost of those "bypass-Z" paths as an extra term that depends on the selection of Z. So, the cost $C_{S,Z}(T)$ of computing T from S given Z is:

$$C_{S,Z}(T) = \sum_{z \in Z} P_S(z) + \sum_{t \in T} P_Z(t) + \sum_{t \in T} P_{S,b}(t) \tag{5}$$

where the first summation term is the cost of computing Z from sources, the second is the cost of computing targets from Z, and the third is the cost of bypass-Z paths.

The *encoding-decoding gain* Φ_Z, defined below, quantifies how significant is the cost reduction provided by such an encoder-decoder architecture,

$$\Phi_Z = \frac{C_S(T)}{C_{S,Z}(T)} \tag{6}$$

If $\Phi_Z \leq 1$, the intermediate nodes do not offer any cost reduction. On the other extreme, if there is a single intermediate node z that depends on all n sources and all m targets depend only on z, then Φ_Z gets its maximum value, $n\,m/(n+m)$.

To illustrate, consider a three-layer network with n sources, k intermediate nodes and m targets, in which every intermediate node depends on every source, and every target depends only on every intermediate node. Suppose that the set Z consists of $k' < k$ of the intermediate nodes. It is easy to see that:

$$\Phi_Z = \frac{k\,n\,m}{k'(n+m) + (k-k')\,n\,m} \tag{7}$$

If $n > 2$ and $m > 2$, we have that $n + m < n\,m$, meaning that Φ_Z is maximized (equal to $n\,m/(n+m)$) when Z includes all k intermediate nodes ($k = k'$).

On the other hand, if the network includes k^+ additional intermediate nodes that only connect to one source and one target, the maximum value of Φ_Z results when the set Z includes only the k densely connected nodes and leaves the k^+ nodes in the bypass paths:

$$\max_Z\{\Phi_Z\} = \frac{k\,n\,m + k^+}{k(n+m) + k^+} \tag{8}$$

		Eleven sets of paths from sensory to motor neurons										
		SP	SP_4	SP_5	SP_4^{+1}	SP_4^{+2}	P_4	SP_5^{+1}	SP_5^{+2}	P_5	SP^{+1}	SP^{+2}
Number of paths		41,305	36,942	40,801	239,941	435,877	441,153	392,895	1,926,944	3,245,610	434,930	3,434,325
10,50,90-percentile of path lengths		3,3,5	3,3,4	3,3,4	3,4,4	3,4,4	3,4,4	3,4,5	4,5,5	4,5,5	3,4,5	4,5,6
Sensory-motor neuron pairs connected		96%	91%	96%	91%	91%	91%	96%	96%	95%	96%	96%
Utilized connections	Total	90%	90%	90%	95%	95%	95%	95%	95%	95%	95%	95%
	FF	95%	95%	95%	95%	95%	95%	95%	95%	95%	95%	95%
	LT	85%	85%	85%	95%	95%	95%	95%	95%	95%	95%	95%

Table 2: Properties of the eleven paths sets from sensory to motor neurons computed using the routing methods of Section 2.2.

Returning to the network of Figure 1, the direct cost $C_S(T)$ is $\sum_{t \in T} P_S(t)$=6+5+3+2+6=22. The cost of constructing the nodes in Z from sources is $\sum_{z \in Z} P_S(z)$= 4+2=6, the cost of constructing targets from Z nodes is $\sum_{t \in T} P_Z(t)$=1+1+1+1+2=6, while the cost of bypass-Z paths is $\sum_{t \in T} P_{S,b}(t)$=2+1+1+0+0=4. So, the encoding-decoding gain is 22/16=1.375 while its maximum possible value is 25/10=2.5.

3 Results

3.1 Feedforward Paths

In Section 2.2, we defined eleven different routing methods for computing paths from sensory to motor neurons in *C. elegans*. Table 2 shows some relevant properties for each of these path sets.

The number of all possible pairs of sensory-motor neurons is 9,492. About 90%-95% of these pairs are connected with any of the eleven path sets. Even with the smallest path set (SP_4), there are typically multiple paths for every sensory-motor pair. The sensory-motor neural paths are typically short: the median is 3 hops meaning that there are typically two other neurons between a sensory and a motor neuron.

The number of paths increases ten-fold when we allow one more hop than the shortest path (SP^{+1}), and about eight-fold more when we allow a second extra hop (SP^{+2}).

A final observation is that about 5%-10% of the connectome (after the removal of FB connections between the three different classes of neurons) are not traversed by any of these paths. This suggests that these connections are utilized only in feedback circuits between neurons of the same type (e.g., feedback between interneurons).

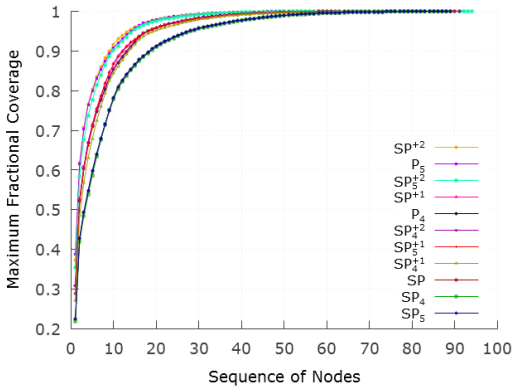
3.2 Hourglass Analysis of Feedforward Paths

Given a set $\mathcal{P}$ of feedforward paths from sensory to motor neurons, we now apply the hourglass analysis framework (see Section 2.4). In particular, the goal is to compute the smallest set of neurons that can cover a percentage τ of all paths in $\mathcal{P}$. That set of neurons is referred to as *τ-Core*.

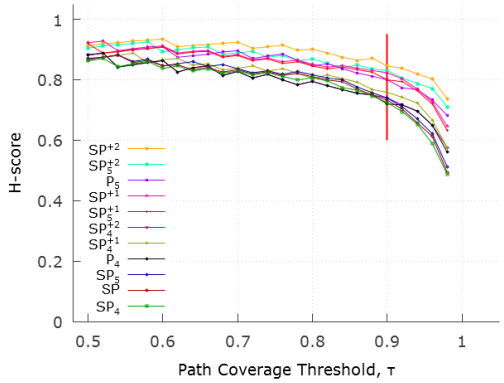
Figure 9(a) shows the cumulative path coverage as a function of the number of neurons in the τ-Core (increasing values of τ require a larger set of core neurons). All path sets have the same "sharp knee" property: almost all (80%-90%, depending on the routing method) of the feedforward paths traverse a small set of about 10 neurons. Routing methods that produce more paths (such as SP^{+2} tend to have a smaller τ-Core than more constrained routing methods (such as SP_4).

Figure 9(b) examines the effect of the path coverage threshold τ on the hourglass metric (H-score), defined in Section 2.4. For all path sets, the H-score is close to one (its theoretical maximum value) as long as $\tau < 90\%$. This suggests an hourglass-like architecture, independent of which routing scheme has produced the set of feedforward paths.

Table 3 shows the sequence of core neurons (for $\tau = 90\%$) for each path set. The first 10-11 of those neurons appear in almost every path set. The remaining neurons appear in more constrained path sets (such as SP) and they only cover a small fraction of additional paths (1%-3%).



(a) Cumulative path coverage by top-X core neurons



(b) Hourglass metric (H-score) as a function of path coverage threshold τ

Figure 9: **Left:** Cumulative path coverage by the top-X core neurons for $X = 1 \ldots 100$. Depending on path selection method, the core size varies between 10-20 neurons when the path coverage threshold τ is 90%. For a given τ, routing methods that produce more paths (such as SP^{+2} or P_5) result in a smaller core. **Right:** Effect of τ on hourglass metric (H-score) for each path set. The ordering of the legends is the same with the top-down ordering of the curves.

Core Neuron	SP	SP_4	SP_5	SP_4^{+1}	SP_4^{+2}	P_4	SP_5^{+1}	SP_5^{+2}	P_5	SP^{+1}	SP^{+2}	Rich-club
AVAL	0.25	0.24	0.25	0.24	0.24	0.24	0.26	0.25	0.25	0.26	0.27	✓
AVAR	0.23	0.22	0.23	0.30	0.34	0.34	0.32	0.40	0.43	0.32	0.41	✓
AVBL	0.08	0.07	0.08	0.10	0.09	0.09	0.09	0.10	0.10	0.09	0.10	✓
AVBR	0.04	0.04	0.04	0.04	0.03	0.03	0.04	0.03	0.03	0.04	0.03	✓
PVCL	0.05	0.05	0.05	0.07	0.07	0.07	0.07	0.07	0.07	0.06	0.06	✓
PVCR	0.03	0.03	0.03	0.02	0.02	0.02	0.02			0.03	0.04	✓
AVEL	0.06	0.06	0.06	0.05	0.04	0.04	0.05	0.04	0.03	0.05	0.03	✓
AVER	0.06	0.06	0.06	0.06	0.05	0.05	0.06	0.05	0.04	0.05	0.04	✓
AVDL	0.02	0.02	0.02	0.02	0.01		0.01					✓
AVDR	0.04	0.04	0.04	0.04	0.03	0.03	0.03	0.02	0.02	0.03	0.01	✓
DVA	0.05	0.05	0.04	0.02	0.03	0.03	0.03	0.03	0.02	0.04	0.01	✓
HSNL	0.01		0.01									
HSNR	0.02	0.03	0.03	0.02	0.02	0.01	0.01	0.01		0.02		
RIAL	0.01	0.02	0.01	0.01	0.02	0.02	0.01		0.01	0.01		
RIAR	0.01	0.02	0.01			0.01						
RIMR	0.01	0.01	0.01	0.01	0.01	0.02						
RMGL	0.01	0.01	0.01									
PVR	0.01	0.01	0.01									
AIBR	0.01	0.01	0.01									
AIZL		0.01										
H-score	0.79	0.79	0.79	0.82	0.84	0.74	0.85	0.86	0.81	0.84	0.87	

Table 3: The identified core neurons when the path coverage threshold is $\tau = 90\%$ for each path set. For each core neuron, we show the fraction of paths that the corresponding neuron contributes to the core. The last column shows the 11 "rich-club" neurons, as identified in [2].

Functional Circuit	Core Neurons					
	AVAL/R	AVBL/R	PVCL/R	AVEL/R	AVDL/R	DVA
Tap-withdrawal [58]	✓	✓	✓		✓	✓
Touch sensitivity [59]	✓	✓	✓		✓	
Mechanosensation [60] – avoidance, foraging, – light-harsh touch, tap [61]	✓	✓	✓		✓	
Locomotion [62]	✓	✓	✓	✓	✓	
Repulsive motion [63]	✓	✓	✓	✓	✓	
Social feeding [64]	✓	✓	✓	✓	✓	
Navigation [65]	✓	✓	✓	✓	✓	
Muscle contraction, Locomotion modulation [66]						✓
Escape response [67]	✓	✓		✓	✓	
Proprioception [68, 69]						✓

Table 4: Some functional circuits associated with core neurons based on the *C. elegans* literature.

If we focus on those first 10-11 core neurons, we observe that, first, they are included in the 90%-core of all path sets we consider (with few exceptions).

Second, that set of core neurons tend to include bilateral pairs of interneurons (namely: AVA, AVB, PVC, AVE, and AVD) – the DVA stretch sensitive core neuron does not appear bilaterally.

Third, the first 11 core neurons with almost every routing method are the same wit the "Rich-club" neurons that were identified using a very different analysis framework [2]. A rich-club is a subgraph of high-degree nodes that are much more densely interconnected with each other than what is expected based on their degrees [57]. In other words, the rich-club concept is based on the analysis of local connectivity rather than paths.

The core neurons that are identified by both the rich-club analysis and the hourglass analysis are neurons that appear early in development – before the first visible signs of motor activity (twitching) [2].

To simplify the presentation of the results, in the rest of this paper we will focus on the "SP^{+2}" path set. This path set results in the largest number of paths and a core of 10 neurons when τ=90%.

3.3 The neurons at the waist of the hourglass

We now review in more detail the functional role of the ten core neurons (using the SP^{+2} set of paths) based on what is known about these neurons from the literature. Seven of the core neurons are located in the head region (AVAR/L, AVBR/L, AVER/L, AVDR) and three are in the tail region (PVCR/L, DVA). The original then core neurons contain nine command interneurons that play a pivotal role in forward and backward locomotion [2]. The other non-command interneuron of the core, DVA, is a proprioceptive interneuron modulating the locomotion circuit [2]. If we want to extend the set of core neurons slightly by covering τ=95% of all paths instead of 90%, we add four more neurons into the core (HSNR, AVDL, RIAL, RIMR).

In Table 4, we have collected a number of circuits from the *C. elegans* literature that associate a specific functional role with each core neuron. The core neurons appear in several circuits, mostly related to spontaneous or planned movement. Many of the adaptive behaviors of the organism such as feeding, egg-laying, escape and navigation result from this require a common set of underlying simpler tasks. Some of the circuits shown (e.g. thermotaxis, chemosensation, olfactory behavior) perform tasks that start with some sensory neurons, followed by a locomotory response that is modulated by some core interneurons.

3.4 Randomization Experiments

We generate 1000 random networks using the algorithm described in Section 2.5. The randomization process preserves the in-degree of each neuron and the hierarchical ordering between neurons (i.e., if neuron v depends on neuron u but u does not depend on v in the original connectome, it

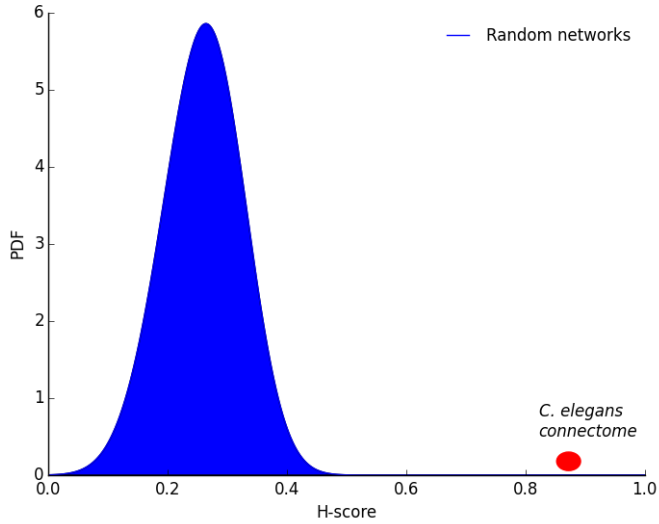


Figure 10: Distribution of H-score for randomized networks in which we preserve the in-degree of each neuron and the hierarchical ordering between neurons. The probability of observing the H-score value of the original network in randomized networks is less than 10^{-3}.

cannot be that u depends on v in a randomized network). Figure 10 shows the H-score distribution of the randomized networks. The H-score of the random networks is significantly less than the corresponding original network ($p < 10^{-3}$), suggesting that the hourglass effect we observe in the *C. elegans* connectome is not a statistical artifact.

3.5 Hourglass organization based on the location metric

In Section 2.6, we defined a location metric that associates each neuron v with a value between 0 and 1, depending on the number of paths from sensory neurons to v and from v to motor neurons.

Figure 11 shows the location of each neuron in a vertical orientation. Most sensory neurons have zero incoming connections (and thus no incoming paths), and so their location is 0. Similarly most motor neurons have zero outgoing connections and so their location is 1. The location of the ten core neurons is shown with dotted rectangles – they are concentrated close to the middle of the location range, meaning that their number of paths from sensory neurons is roughly the same with their number of paths to motor neurons. This visualization has been produced with the SP^{+2} path set but other path sets give similar results.

3.6 Encoder-Decoder Architecture

What is the functional significance of the hourglass organization in the context of *C. elegans*? Is it just an interesting graph-theoretic property or does this architecture provide an adaptive advantage that could be selected by evolution? In this section, we apply the cost-based framework of Section 2.7 to answer this question.

We can think of *C. elegans* as an information processing system that transforms input information, collected and encoded by sensory neurons, to output information that is represented by the activity of motor neurons. The analysis of the previous sections has identified a number of *core neurons* that most of the sensory-to-motor neural pathways go through. The exact number of core neurons depends on the fraction τ of all sensory-to-motor paths covered by the core. Table 3 shows the 10 core neurons when τ=90% (for the SP^{+2} set of paths).

Suppose that a given set of core neurons forms the intermediate set Z of Section 2.7. We can then compute the number of paths $P_S(Z)$ from the set S of all sensory neurons to the neurons in Z as a proxy for the information processing cost of an encoding operation that transforms S to Z. Similarly, the number of paths $P_S(Z)$ from the neurons in Z to the set T of all motor neurons

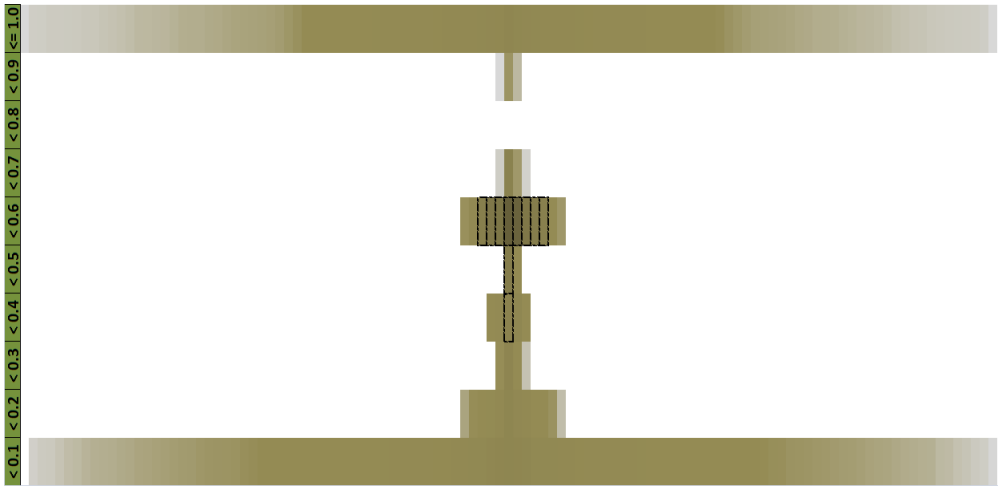


Figure 11: Visualization of the *C. elegans* connectome based on the location metric. We discretize the location metric in 10 bins (each bin accounting for 1/10 of the $0-1$ range). The path centrality of each node is represented by its color intensity (darker for higher path centrality). Nodes with higher centrality are placed closer to the vertical mid-line. The core nodes are represented by dotted rectangles. The location of all core nodes falls between 0.3-0.6.

can be thought of as a proxy for the information processing cost of a decoding operation that transforms Z to T. We also need to consider any sensory-to-motor paths $P_{S,b}(T)$ that bypass the core neurons in Z — this is a proxy for the cost of any additional information processing that is specific to each motor neuron and that is not provided by the encoding-decoding function of Z.

These three cost terms are shown in Figure 12 as we increase the number of core neurons included in Z (i.e., as we increase the threshold τ). The bypass-Z cost is the dominant cost term until we include about 15 neurons in Z. This suggests that the information provided by sensory neurons cannot be captured well with fewer neurons. On the other hand, the costs of the encoding and decoding operations ($P_S(Z)$ and $P_S(Z)$, respectively) increase with the number of neurons in Z as expected.

The encoder-decoder gain ratio Φ_Z (see Equation 6) shows that the maximum cost reduction takes place when we consider the first 16 core neurons (corresponds to τ=95% for the SP^{+2} set of paths). In that case, the encoder-decoder architecture achieves an eight-fold decrease (Φ_Z=8.2) in terms of information processing cost relative to a hypothetical architecture in which the information processing cost of each motor neuron is computed separately, based on the number of paths from sensory neurons to that motor neuron.

3.7 Including Gap Junctions: the Complete Network

Gap junctions provide a different type of connectivity between neurons than chemical synapses. Chemical synapses use neurotransmitters to transfer information from the presynaptic to the post-synaptic neuron, while gap junctions work by creating undirected electrical channels and they provide faster (but typically weaker) neuronal coupling. The directionality of the current flow in gap junctions cannot be detected from electron micrographs. Hence they are treated as bidirectional in the *C. elegans* connectome.

In this section, we consider both gap junctions and chemical synapses, forming what we refer to as the *complete network*. Gap junctions connect 253 (out of 279) neurons through 514 undirected connections (that we treat as 1028 directional connections). Recall that the number of (directed) chemical connections is 2194.

64% of the gap junction connections do not co-occur with chemical connections between the same pair of neurons, while 83% of the synaptic connections do not co-occur with gap junction connections. In other words, the inclusion of gap junctions changes significantly the connectivity between neurons.

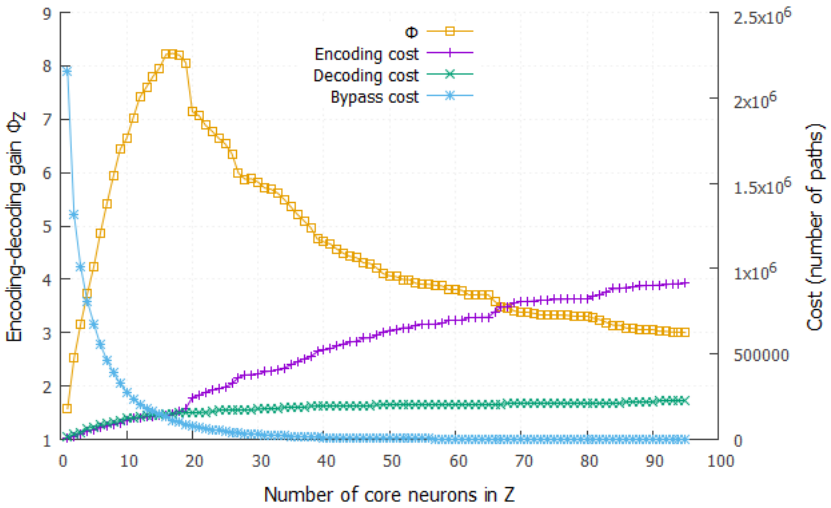


Figure 12: The encoder-decoder gain ratio Φ_Z as the number of core neurons in the encoding set Z increases. The maximum value of Φ_Z is about 8.2 when Z includes the first 16 core neurons. Based on the cost framework of Section 2.7, this means that the hourglass organization of the *C. elegans* connectome reduces the sensory-to-motor information processing cost eight-fold. The figure also shows the three relevant cost terms: cost of encoding the information provided by sensory neuron using neurons in Z, cost of decoding that information to drive all motor neurons, and cost of processing pathways that bypass the core.

Neuron	*AVAL*	*AVAR*	*AVBL*	*AVBR*	*AVEL*	*AVER*	*PVCL*	AIBR	*AVDR*	*DVA*	*PVCR*	VD01
Fraction of new paths covered	0.29	0.23	0.16	0.11	0.05	0.04	0.03	0.03	0.02	0.02	0.01	0.01

Table 5: The identified 12 core neurons in the complete network. The 10 neurons shown in red-italic were also the core of the synaptic network.

The complete network has 1180 FF, 1468 LT and 574 FB connections. If we remove feedback connections, as we did for the synaptic network, we end up with a total of 2648 directed connections.

The inclusion of gap junctions also increases significantly the number of paths between sensory and motor neurons, independent of the routing method. If we focus on SP^{+2}, the number of paths increases by a factor of 2.3 (about 7.7 millions).

Figure 13(a) shows the cumulative path coverage as a function of the number of nodes in the core. Figure 13(b) examines the effect of the path coverage threshold τ on the resulting H-score. Both curves are quite similar to the corresponding results for the synaptic network.

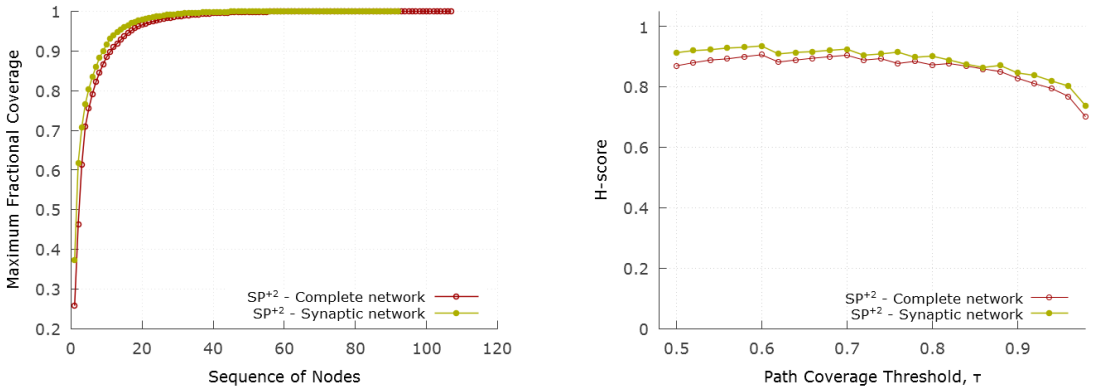
With τ=90%, the resulting core nodes are shown in Table 5. The H-score for the complete network is 0.83 (compared to 0.87 for the synaptic network).

The two core neurons that appear in the hourglass waist of the complete network are:

- AIBR: related to locomotion, food and odor evoked behaviors, local search, lifespan and starvation response.
- VD01: related to motor - Sinusoidal body movement-locomotion.

4 Discussion

The hourglass architecture offers a new way to understand the functional role of core neurons in the *C. elegans* connectome. An hourglass network integrates and compresses many input signals into a small number of core functions at the waist of the hourglass. Then, those core functions are combined in different ways to compute the output of the system [70]. The *C. elegans* network has the topological properties of a hourglass architecture. So, instead of associating the core neurons



(a) Cumulative path coverage by top-X core neurons

(b) Hourglass metric (H-score) as a function of path coverage threshold τ

Figure 13: Hourglass analysis for complete network with the SP^{+2} set of paths.

with specific circuits or movements, it is possible that the core neurons compress the information provided by almost all sensory neurons, and encode few (around ten) common sub-functions that are then re-used by several motor neurons.

In the rest of this Section, we review some prior work in the *C. elegans* literature that has produced results that relate to the hourglass effect. Varshney et al. [44] analyzed the structural properties of the *C. elegans* connectome and found that several central neurons (based on closeness centrality) play a key role in information processing. Among them are command inter-neurons such as *AVA, AVB, AVE* that are responsible for locomotion control. On the other hand, neurons such *DVA* or *ADE* have high out-closeness centrality and a good position to propagate a signal to the rest of the network. Most of the "central" neurons in that study are also included in the hourglass core.

Furthermore, those "central" neurons have been found to be tightly interconnected to each other, forming a "rich-club" [2]. Further, evidences have shown that rich-club neurons are among the first to develop, in the time period before the animal can twitch [2]. The core neurons that we identify highly overlap with the rich-club neurons, although the two analysis frameworks are significantly different.

The existence of a set of highly connected and central hub nodes, termed as "rich-club", has been established in the *C. elegans* connectome by Towlson et al. [2]. Further, they showed that those rich-club neurons are among the first to develop. The core neurons that we identify highly overlap with the rich-club neurons, although the two analysis frameworks are significantly different.

The modular organization of the *C. elegans* connectome has been discovered by Sohn et al. [71] through cluster analysis. Their analysis showed that communities correspond well to known functional circuits and it helped uncover the role of a few previously unknown neurons. They also identified a hierarchical organization among five key clusters that form a backbone for higher-order complex behaviors.

The modular or mesoscale organization of the *C. elegans* connectome has been also analyzed by Pan et al. [72]. Their findings corroborate the notion of functional modularity, with neurons in the same module located close together and contributing in the same task. They contrasted this notion of modularity to that arising from wiring length minimization and communication cost efficiency and concluded that functional relevance is more important in shaping structure. Along these lines, they identified critical neurons in every module, most of which have well known functionality including the command interneurons that also appear in the hourglass set of core neurons.

The physical placement of neurons in *C. elegans* has been thought to be not exclusively optimized for global minimum wiring but rather for a variety of other factors of which the minimization of pair-wise processing steps is important[51]. For example, Kaiser and Hilgetag [51] showed that the total wiring length can be reduced by 48% by optimally placing the neurons. However that

would significantly increase the number of processing nodes along shortest paths between components as well. Similar findings were also revealed by Chen et al. [73], concluding that the placement of neurons does not globally minimize wiring length. These studies emphasize the notion of choosing shorter communication paths between neuron pairs and supports our approach of choosing paths that are shortest, or close to shortest, in terms of processing steps.

The posterior nervous systems of the *C. elegans* was analyzed by Jarrell et al. [74] to understand the decision making mechanism and behavior of the nematode. One of their conclusions was that the nervous system has a mostly feedforward architecture that runs from sensory to motor neurons via interneurons. There is also some feedback circuitry in the nervous system and the actual physical output of the worm (i.e. motion etc.) feeds back to sensory neurons to allow closed-loop control. There are however many more feedforward loops (termed lateral connections in our analysis) that provide localized coordination most notably visible within interneurons.

Analysis by Csoma et al. [75] challenged the well rooted notion of shortest path based communication routing in the human brain network. They collected empirical data through diffusion MRI and concluded that although a large number paths conform to the shortest path assumption, a significant fraction (20-40%) are inflated up to 4-5 hops.

Research by Avena-Koenigsberger et al. [76, 49] analyzed in depth communication strategies in the human brain and also challenged the shortest path assumption. They discussed how the computation of shortest path routing is not feasible in the brain circuitry, and the shortest path routes would leave out around 80% of neural connections. They examined the spectrum of routing strategies hinging upon the amount of global information and communication required. At one end of the spectrum, we have random-walk routing mechanisms that are wasteful and often fail to achieve efficient routes but require no knowledge. On the other end we have shortest-path routing requiring global wiring knowledge at each neuron. As a more realistic choice, they studied the k-shortest path based approach (with k being 100). Their findings show that this strategy increases the utilization of connections. Therefore, we have used a more relaxed constraint to chose paths between any two nodes by allowing all possible paths that are up to 2 hops longer than the shortest path between the corresponding pair.

At a higher level, the hourglass effect is also significant for other reasons. One of them is that the modules at the waist of the network create a "bottleneck" in the flow of information from sources (or inputs) to targets (or outputs). Such bottleneck network effects have been studied in the literature under different names. For instance, the term "core-periphery networks" has been broadly used in network science to refer to various static and dynamic topological properties (e.g., rich-club effect, onion-like networks) that result from a dense, cohesive core that is connected to sparsely connected peripheral nodes (but not necessarily organized in an acyclic input-output hierarchy) [77, 39, 78]. In information theory, bottlenecks have been studied as a way to compress a given random vector X into an lower dimensionality intermediate representation T that is still able to predict the desired output Y well. The optimal T then depends on this trade-off between how well can Y be predicted and how complex T is [46, 47].

The hourglass effect is also significant for the evolvability and robustness of hierarchically modular systems. Intuitively, the hourglass effect should allow a system to accommodate frequent changes in its sources or targets (i.e., to be able to evolve as the environment changes) because the few modules at the waist "decouple" the large number of sources from the large number of targets. If there is a change in the inputs (sources), the outputs do not need to be modified as long as the modules at the waist can still function properly. Similarly, if there is need for a new target, it may be much easier (or cheaper) to construct it reusing the modules at the waist rather than directly relying on sources. This is related to the notion of "constraints that de-constrain", introduced by Kirschner and Gerhart in the context of biological development and evolvability [79]. At the same time however, the presence of these critical modules at the waist (the "constraints") limit the space of all possible outputs that the system can generate ("phenotype space"), at least for a given maximum cost.

In conclusion, our study suggests that the *C. elegans*' connectome follows an "hourglass architecture". In networks of this type, the input vector is first compressed to a low-dimensional latent space that encodes in a more efficient manner the input data, given the desired input-output mapping, followed by a decoding network through which those intermediate-level representations are combined in different ways to compute the correlated outputs of the network. The hourglass

architecture pave a new avenue to think about, analyze, and experiment with neural systems.

References

[1] Kaeser M Sabrin and Constantine Dovrolis. The hourglass effect in hierarchical dependency networks. *Network Science*, 5(4):490–528, 2017.

[2] Emma K Towlson, Petra E Vértes, Sebastian E Ahnert, William R Schafer, and Edward T Bullmore. The rich club of the *C. elegans* neuronal connectome. *Journal of Neuroscience*, 33(15):6380–6387, 2013.

[3] Danielle S Bassett and Michael S Gazzaniga. Understanding complexity in the human brain. *Trends in cognitive sciences*, 15(5):200–209, 2011.

[4] David Meunier, Renaud Lambiotte, and Edward T Bullmore. Modular and hierarchically modular organization of brain networks. *Frontiers in Neuroscience*, 4:200, 2010.

[5] David Lorge Parnas, Paul C Clements, and David M Weiss. The modular structure of complex systems. In *Proceedings of the 7th International Conference on Software Engineering*, pages 408–417. IEEE Press, 1984.

[6] Melissa A Schilling. Toward a general modular systems theory and its application to interfirm product modularity. *Academy of Management Review*, 25(2):312–334, 2000.

[7] Carliss Young Baldwin and Kim B Clark. *Design Rules: The Power of Modularity*, volume 1. MIT press, Cambridge, 2000.

[8] Werner Callebaut and Diego Rasskin-Gutman. *Modularity: Understanding the Development and Evolution of Natural Complex Systems.* MIT press, Cambridge, 2005.

[9] Günter P Wagner, Mihaela Pavlicev, and James M Cheverud. The road to modularity. *Nature Reviews Genetics*, 8(12):921–931, 2007.

[10] Erzsébet Ravasz and Albert-László Barabási. Hierarchical organization in complex networks. *Physical Review E*, 67(2):026112, 2003.

[11] Marta Sales-Pardo, Roger Guimera, André A Moreira, and Luís A Nunes Amaral. Extracting the hierarchical organization of complex systems. *Proceedings of the National Academy of Sciences*, 104(39):15224–15229, 2007.

[12] Herbert A Simon. *The architecture of complexity.* Springer, 1991.

[13] Haiyuan Yu and Mark Gerstein. Genomic analysis of the hierarchical structure of regulatory networks. *Proceedings of the National Academy of Sciences*, 103(40):14724–14731, 2006.

[14] Jeff Clune, Jean-Baptiste Mouret, and Hod Lipson. The evolutionary origins of modularity. *Proceedings of the Royal Society of London B: Biological Sciences*, 280(1755):20122863, 2013.

[15] Henok Mengistu, Joost Huizinga, Jean-Baptiste Mouret, and Jeff Clune. The evolutionary origins of hierarchy. *PLoS computational biology*, 12(6):e1004829, 2016.

[16] Miguel A Fortuna, Juan A Bonachela, and Simon A Levin. Evolution of a modular software network. *Proceedings of the National Academy of Sciences*, 108(50):19985–19989, 2011.

[17] Chun-Che Huang and Andrew Kusiak. Modularity in design of products and systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 28(1):66–77, 1998.

[18] Christopher R Myers. Software systems as complex networks: Structure, function, and evolvability of software collaboration graphs. *Physical Review E*, 68(4):046116, 2003.

[19] Nadav Kashtan and Uri Alon. Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences of the United States of America*, 102(39):13773–13778, 2005.

[20] Nadav Kashtan, Elad Noor, and Uri Alon. Varying environments can speed up evolution. *Proceedings of the National Academy of Sciences*, 104(34):13711–13716, 2007.

[21] Dirk M Lorenz, Alice Jeng, and Michael W Deem. The emergence of modularity in biological systems. *Physics of Life Reviews*, 8(2):129–160, 2011.

[22] H Kirsten and Paulien Hogeweg. Evolution of networks for body plan patterning; interplay of modularity, robustness and evolvability. *PLoS Comput Biol*, 7(10):e1002208, 2011.

[23] Hiroaki Kitano. Biological robustness. *Nature Reviews Genetics*, 5(11):826–837, 2004.

[24] Jörg Stelling, Uwe Sauer, Zoltan Szallasi, Francis J Doyle, and John Doyle. Robustness of cellular functions. *Cell*, 118(6):675–685, 2004.

[25] Tamar Friedlander, Avraham E Mayo, Tsvi Tlusty, and Uri Alon. Evolution of bow-tie architectures in biology. *PLoS computational biology*, 11(3):e1004055, 2015.

[26] Tanita Casci. Development: Hourglass theory gets molecular approval. *Nature Reviews Genetics*, 12(2):76–76, 2011.

[27] Marcel Quint, Hajk-Georg Drost, Alexander Gabel, Kristian Karsten Ullrich, Markus Bönn, and Ivo Grosse. A transcriptomic hourglass in plant embryogenesis. *Nature*, 490(7418):98–101, 2012.

[28] Hong-Wu Ma and An-Ping Zeng. The connectivity structure, giant strong component and centrality of metabolic networks. *Bioinformatics*, 19(11):1423–1430, 2003.

[29] R Tanaka, M Csete, and J Doyle. Highly optimised global organisation of metabolic networks. *IEE Proceedings-Systems Biology*, 152(4):179–184, 2005.

[30] Jing Zhao, Hong Yu, Jian-Hua Luo, Zhi-Wei Cao, and Yi-Xue Li. Hierarchical modularity of nested bow-ties in metabolic networks. *BMC Bioinformatics*, 7(1):386, 2006.

[31] Bruce Beutler. Inferences, questions and possibilities in toll-like receptor signalling. *Nature*, 430(6996):257–263, 2004.

[32] Kanae Oda and Hiroaki Kitano. A comprehensive map of the toll-like receptor signaling network. *Molecular Systems Biology*, 2(1), 2006.

[33] Jochen Supper, Lucía Spangenberg, Hannes Planatscher, Andreas Dräger, Adrian Schröder, and Andreas Zell. Bowtiebuilder: modeling signal transduction pathways. *BMC Systems Biology*, 3(1):1, 2009.

[34] R Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005.

[35] Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2(11):1019–1025, 1999.

[36] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[37] Saamer Akhshabi and Constantine Dovrolis. The evolution of layered protocol stacks leads to an hourglass-shaped architecture. *ACM SIGCOMM Computer Communication Review*, 41(4):206–217, 2011.

[38] Jayashankar M Swaminathan, Stephen F Smith, and Norman M Sadeh. Modeling supply chain dynamics: A multiagent approach*. *Decision Sciences*, 29(3):607–632, 1998.

[39] Peter Csermely, András London, Ling-Yun Wu, and Brian Uzzi. Structure and dynamics of core/periphery networks. *Journal of Complex Networks*, 1(2):93–123, 2013.

[40] Petter Holme. Core-periphery organization of complex networks. *Physical Review E*, 72(4):046111, 2005.

[41] Marie Csete and John Doyle. Bow ties, metabolism and disease. *TRENDS in Biotechnology*, 22(9):446–450, 2004.

[42] Tomislav Domazet-Lošo and Diethard Tautz. A phylogenetically based transcriptome age index mirrors ontogenetic divergence patterns. *Nature*, 468(7325):815–818, 2010.

[43] Saamer Akhshabi, Shrutii Sarda, Constantine Dovrolis, and Soojin Yi. An explanatory evo-devo model for the developmental hourglass. *f1000research*, 3, 2014.

[44] Lav R Varshney, Beth L Chen, Eric Paniagua, David H Hall, and Dmitri B Chklovskii. Structural properties of the *Caenorhabditis elegans* neuronal network. *PLoS computational biology*, 7(2):e1001066, 2011.

[45] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[46] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[47] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

[48] Jane B Reece, Lisa A Urry, Michael Lee Cain, Steven Alexander Wasserman, Peter V Minorsky, Robert B Jackson, et al. *Campbell biology*. Pearson Boston, 2014.

[49] Andrea Avena-Koenigsberger, Bratislav Misic, and Olaf Sporns. Communication dynamics in complex brain networks. *Nature Reviews Neuroscience*, 19(1):17, 2018.

[50] Ed Bullmore and Olaf Sporns. The economy of brain network organization. *Nature Reviews Neuroscience*, 13(5):336, 2012.

[51] Marcus Kaiser and Claus C Hilgetag. Nonoptimal component placement, but short processing paths, due to long-distance projections in neural systems. *PLoS computational biology*, 2(7):e95, 2006.

[52] Ashish Raj and Yu-hsien Chen. The wiring economy principle: connectivity determines anatomy in the human brain. *PloS one*, 6(9):e14832, 2011.

[53] Farras Abdelnour, Henning U Voss, and Ashish Raj. Network diffusion accurately models the relationship between structural and functional brain connectivity networks. *Neuroimage*, 90:335–347, 2014.

[54] Bratislav Mišić, Richard F Betzel, Azadeh Nematzadeh, Joaquin Goni, Alessandra Griffa, Patric Hagmann, Alessandro Flammini, Yong-Yeol Ahn, and Olaf Sporns. Cooperative and competitive spreading dynamics on the human connectome. *Neuron*, 86(6):1518–1529, 2015.

[55] Vatche Ishakian, Dóra Erdös, Evimaria Terzi, and Azer Bestavros. A framework for the evaluation and management of network centrality. In *SDM*, pages 427–438. SIAM, 2012.

[56] Brian Karrer and Mark EJ Newman. Random graph models for directed acyclic networks. *Physical Review E*, 80(4):046110, 2009.

[57] Vittoria Colizza, Alessandro Flammini, M Angeles Serrano, and Alessandro Vespignani. Detecting rich-club ordering in complex networks. *Nature physics*, 2(2):110, 2006.

[58] Mathias Lechner, Radu Grosu, and Ramin M Hasani. Worm-level control through search-based reinforcement learning. *arXiv preprint arXiv:1711.03467*, 2017.

[59] Martin Chalfie, John E Sulston, JOHN G White, Eileen Southgate, J Nicol Thomson, and Sydney Brenner. The neural circuit for touch sensitivity in *Caenorhabditis elegans*. *Journal of Neuroscience*, 5(4):956–964, 1985.

[60] Miriam B Goodman. Mechanosensation. *WormBook: the online review of C. elegans biology*, pages 1–14, 2006.

[61] Donald L Riddle, Thomas Blumenthal, Barbara J Meyer, and James R Priess. Mechanosensory control of locomotion. In *C. elegans II*. Cold Spring Harbor Laboratory Press, 1997.

[62] M Driscoll and J Kaplan. Mechanotransduction. In *C. elegans II*. Cold Spring Harbor Laboratory Press, 1997.

[63] Zengcai V Guo, Anne C Hart, and Sharad Ramanathan. Optical interrogation of neural circuits in *Caenorhabditis elegans*. *Nature methods*, 6(12):891, 2009.

[64] Shawn R Lockery. Neuroscience: A social hub for worms. *Nature*, 458(7242):1124, 2009.

[65] Jesse M Gray, Joseph J Hill, and Cornelia I Bargmann. A circuit for navigation in *Caenorhabditis elegans*. *Proceedings of the National Academy of Sciences of the United States of America*, 102(9):3184–3191, 2005.

[66] Wei Li, Zhaoyang Feng, Paul W Sternberg, and XZ Shawn Xu. A *C. elegans* stretch receptor neuron revealed by a mechanosensitive trp channel homologue. *Nature*, 440(7084):684, 2006.

[67] Jennifer K Pirri and Mark J Alkema. The neuroethology of *C. elegans* escape. *Current opinion in neurobiology*, 22(2):187–193, 2012.

[68] Steven J Husson, Alexander Gottschalk, and Andrew M Leifer. Optogenetic manipulation of neural activity in *C. elegans*: from synapse to circuits and behaviour. *Biology of the Cell*, 105(6):235–250, 2013.

[69] William R Schafer. Mechanosensory molecules and circuits in *C. elegans*. *Pflügers Archiv-European Journal of Physiology*, 467(1):39–48, 2015.

[70] Payam Siyari, Bistra Dilkina, and Constantine Dovrolis. Emergence and evolution of hierarchical structure in complex systems. *arXiv preprint arXiv:1805.04924*, 2018.

[71] Yunkyu Sohn, Myung-Kyu Choi, Yong-Yeol Ahn, Junho Lee, and Jaeseung Jeong. Topological cluster analysis reveals the systemic organization of the *Caenorhabditis elegans* connectome. *PLoS computational biology*, 7(5):e1001139, 2011.

[72] Raj Kumar Pan, Nivedita Chatterjee, and Sitabhra Sinha. Mesoscopic organization reveals the constraints governing *Caenorhabditis elegans* nervous system. *PloS one*, 5(2):e9240, 2010.

[73] Beth L Chen, David H Hall, and Dmitri B Chklovskii. Wiring optimization can relate neuronal structure and function. *Proceedings of the National Academy of Sciences*, 103(12):4723–4728, 2006.

[74] Travis A Jarrell, Yi Wang, Adam E Bloniarz, Christopher A Brittin, Meng Xu, J Nichol Thomson, Donna G Albertson, David H Hall, and Scott W Emmons. The connectome of a decision-making neural network. *Science*, 337(6093):437–444, 2012.

[75] Attila Csoma, Attila Kőrösi, Gábor Rétvári, Zalán Heszberger, József Bíró, Mariann Slíz, Andrea Avena-Koenigsberger, Alessandra Griffa, Patric Hagmann, and András Gulyás. Routes obey hierarchy in complex networks. *Scientific reports*, 7(1):7243, 2017.

[76] Andrea Avena-Koenigsberger, Bratislav Mišić, Robert XD Hawkins, Alessandra Griffa, Patric Hagmann, Joaquín Goñi, and Olaf Sporns. Path ensembles and a tradeoff between communication efficiency and resilience in the human connectome. *Brain Structure and Function*, 222(1):603–618, 2017.

[77] Stephen P Borgatti and Martin G Everett. Models of core/periphery structures. *Social networks*, 21(4):375–395, 2000.

[78] M Puck Rombach, Mason A Porter, James H Fowler, and Peter J Mucha. Core-periphery structure in networks. *SIAM Journal on Applied mathematics*, 74(1):167–190, 2014.

[79] Marc Kirschner and John Gerhart. Evolvability. *Proceedings of the National Academy of Sciences*, 95(15):8420–8427, 1998.