PhD progress report

Tomas Fiers

Topic: Mapping the wires using neural activity

Supervisors: Mark Humphries & Matias Ison
Internal assessor: Rüdiger Thul

University of Nottingham
School of Psychology

December 2022

Contents

1 Introduction

This report gives a brief overview of the work that has been done since the last progress report,[1] and gives a timeline for completion of the PhD.

[1] The year 1 progress report can be found at https://1drv.ms/w/s!AgiWRxzu_jv8g_JixyHaDSHLulS6VA?e=a5Blyw

We start with a recap of the topic and what was presented in the last report.

Inferring neural connections from voltage traces

Network inference in systems neuroscience seeks to find the connections between individual neurons when anatomical wire tracing is infeasible (like it is *in vivo*). This is usually done based on spike-train recordings. But spiking is sparse, and so purely spike-based network inference algorithms typically perform poorly.

In this PhD, we propose to instead use *voltage imaging* data for network inference. Because this imaging technique can record subthreshold voltages, we can exploit a direct causal link between connectivity and activity: a succesful spike will elicit a precisely-timed post-synaptic potential (PSP) in the downstream neuron's voltage (figure 1). By looking for these PSPs in one neuron's voltage traces, synchronized to another neuron's spikes, we can infer the existence of direct connections.

Work up to last progress report

I reviewed the voltage imaging literature, from which we learn that we can expect recordings of about 10 minutes long, with noise at a "spike-SNR" of at least 10.[2] The number of neurons simultaneously recorded from *in vivo* is low (roughly 10 – 40); but is rapidly increasing. These are the parameters within which our network inference algorithms should work.

[2] A spike-SNR of 10 means that σ_{noise} is one-tenth of a spike's height.

We test the idea for network inference using spiking neural network simulations. We numerically integrate the Izhikevich equations, a 2D dynamical system whose bifurcations model different neuron types. External input is provided by Poisson spike trains, in a so called 'N-to-1' experiment, detailed in figure 2.

As a connection test, we use simple spike-triggered averaging (STA) (figure 3), which turned out to already have good detection performance.

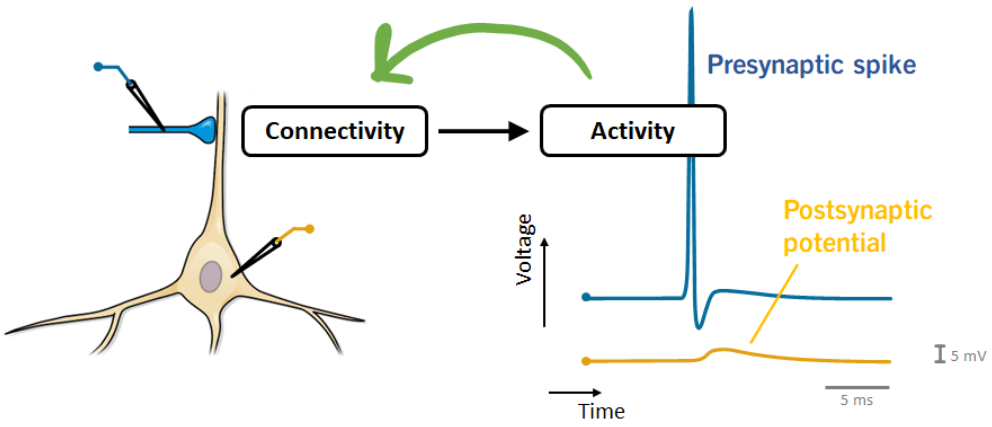

Figure 1: **The causal link between neural connectivity and activity.** On the left, a cartoon of a synaptic connection. The axon of presynaptic neuron M (in blue) impinges on postsynaptic neuron N (in brown). The electrode icons indicate that their membrane voltages are recorded (shown on the right). A succesful spike in neuron M will elicit a small but precisely-timed voltage bump in neuron N (the postsynaptic potential, PSP). There is thus a causal relationship between 1) the existence of a connection $M \rightarrow N$ and 2) both neurons' membrane voltages. This causal relationship (black arrow) is exploited to perform network inference from voltage recordings (green arrow).
Drawings adapted from Purves et al.'s "Neuroscience" textbook, 6th edition, 2018.

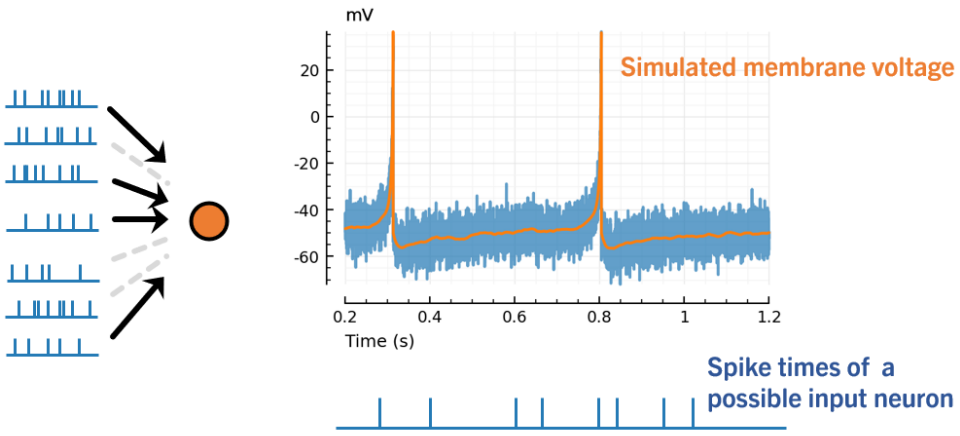

Figure 2: **The 'N-to-1' problem.**
Left: A neuron N (orange circle), and the spike trains of other neurons in the network (blue). Some of these other neurons impinge directly on N (black arrows), while others are not (directly) connected (dashed gray lines). Given only neuron N's voltage signal and the other neurons' spiketrains, we want to detect the direct inputs, while rejecting the not-directly-connected spiketrains.
Right: The simulated membrane voltage of the impinged-upon neuron (orange), and the same signal with Gaussian noise added, to simulate a voltage imaging signal (blue). Underneath the plot, one of the possible input spiketrains, time-aligned to the voltage signal. This alignment is used later to extract spike-triggered windows from the voltage signal.

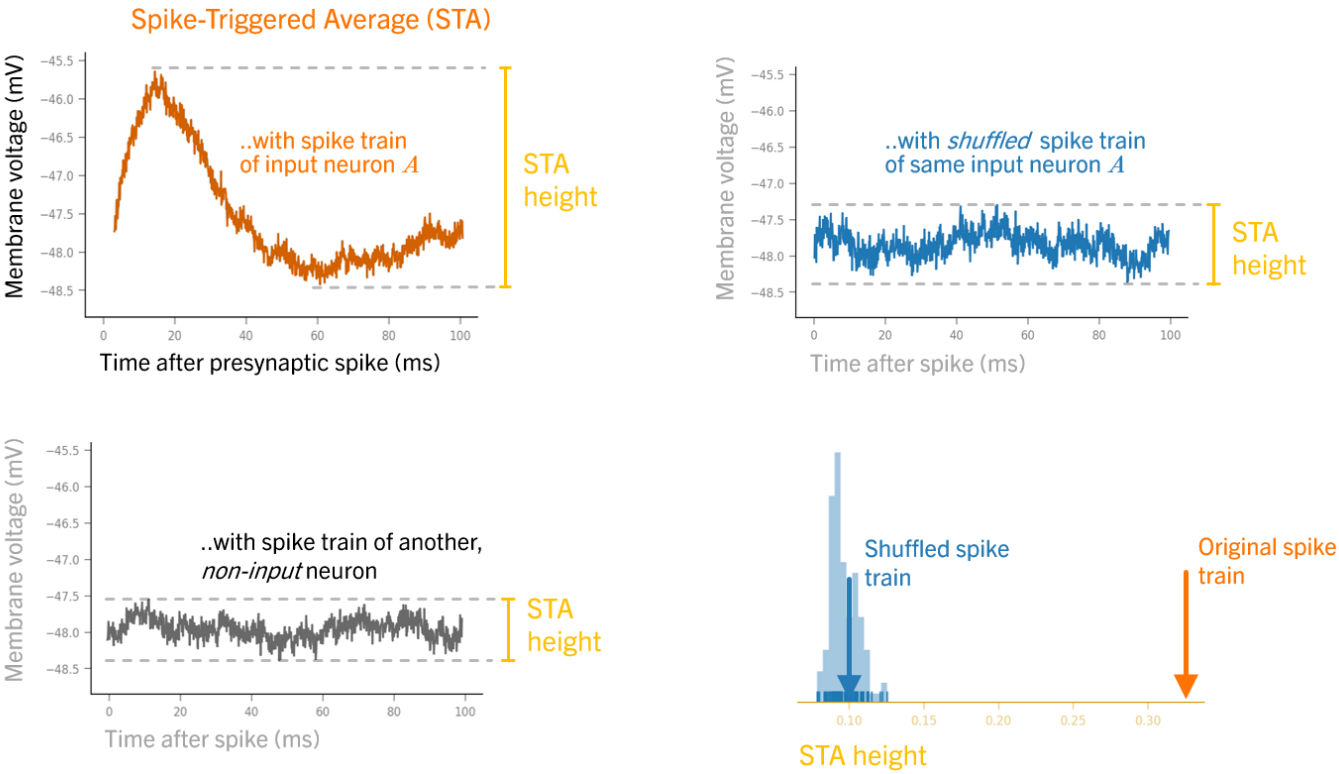

Figure 3: **A simple connection test: STA height with shuffle control.** The spikes of a possible input neuron are aligned to the voltage trace of the neuron of interest N, as in figure 2. For every such spike, a 100-ms long window is cut out of the voltage of N. The average of all these windows is called the spike-triggered average (STA).

Left: Two example STAs of neuron N's membrane voltage: one for an actually connected input neuron, M (top, orange); and one for a non-input neuron (below, gray). Given an STA signal x, we will use its height $h = \max(x) - \min(x)$ (also known as 'peak-to-peak' or 'ptp') to test whether two neurons are connected.

Right: An STA of N's membrane voltage using a shuffled version of M's spike times (which is made by randomly permuting the inter-spike-intervals of M). This 'shuffled STA height' provides a control for the STA height connection test statistic: "what do we expect the STA height to be if there is *no* connection $M \to N$". By calculating different such shuffles, we obtain a null-distribution for the STA height test statistic. And by comparing the real STA height to this distribution, we can calculate a p-value. Here, the real STA is larger than all shuffle controls, of which there are 100. So $p < 0.01$, and at $\alpha = 0.05$, we conclude there is indeed a connection $M \to N$.

2 N-to-1 experiments

Previously, we found that connections can be detected well in the N-to-1 setup of figure 2, using the STA-height shuffle test (figure 3): all connections are detected, at arbitrarily low false positive rates (see e.g. figure 4, at number of connected inputs = 21, for $\alpha = 0.05$).

We examined that result further by modifying different parameters of the simulation, such as adding more noise, and increasing the number of input neurons (the N in 'N-to-1)'. We also added inhibitory input neurons, and tested different E:I ratios, and E:I connection strength ratios (where connection strength Δg is the increase in postsynaptic conductance g upon spike arrival). Finally, we investigated the effects of different inhibitory reversal potentials.

We found that for all the tested E:I scenarios, detection performance (both sensitivity and precision, for both excitatory and inhibitory inputs) remained high (figure 5). However, when the number of input spike trains N was increased to higher (and more realistic) numbers,[3] detection performance degraded (figure 4).

We gave each Poisson input a firing rate according to a log-normal distribution (biological firing rates also have a heavy-tailed distribution). We expected that the few neurons with high firing rates would still be detectable in the high-N case. But a bug in the simulation code made it impossible for input connections to have high spike rates in the high-N regime.[4] This made the high-N results invalid. So that specific analysis will have to be redone. (See Timeline at the end of this report).

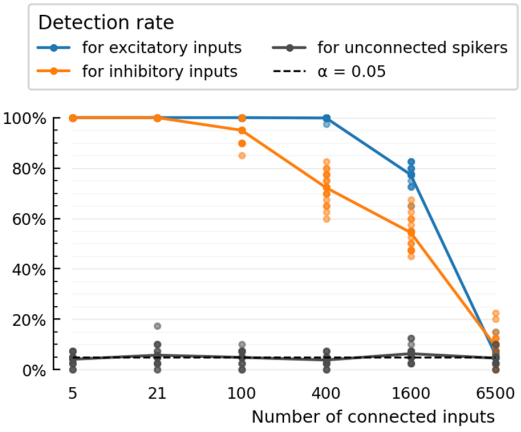

Figure 4: **Inference performance for different N.**
10-minute simulations.
16 simulations per condition (different random number generator seeds).
$N_{exc}/N_{inh} = \Delta g_{inh}/\Delta g_{exc} = 4$.

[3] Up to 7000 inputs – roughly what may be found in the cortex.

[4] If the number of spikes, aggregated over all input connections, was too high (namely more than one spike per timestep, i.e. at about 10 000 total input spikes per second), the input spike queue became backed up, and the spikes were not all processed in time.

Figure 5: **Inference performance in different E:I regimes.**
Left: the output neuron fires more (warmer colours) if there are less inhibitory inputs (y-axis), or if the inhibitory inputs are weaker (x-axis). On the diagonals, these two ratios are balanced.
Right: Area under the ROC-curve, which plots the true positive versus the false positive rate over the entire range $[0, 1]$ of p-value threshold values α. 1 is perfect performance (yellow), 0.5 is chance (gray). Each square is the average over six simulations.

3 Random recurrent network

In the previous experiments, only one neuron's voltage was simulated. The inputs were Poisson spike trains. (I.e, we explicitly investigated the N-to-1 problem). In the next experiments, we simulate the voltages of a full network of neurons, which are recurrently connected to each other. The goal is to investigate the effect on network inference of potentially correlated inputs and indirect connections.

Connectivity structure

We choose the simple and common 'fully random' connectivity rule,[5] where any neuron has a connection to another with a uniform random probability (we choose $p_{\text{conn}} = 0.04$). After generating an adjacency matrix this way (`A = rand(N, N) .≤ 0.04`, where `rand` draws from $\sim U[0, 1]$), we remove autapses. We choose the number of neurons $N = 1000$.

[5] Other common choices for connectivity structure are scale-free networks, and 'local' networks.

A property of fully random networks is that they are strongly interconnected. In our network, any neuron is reachable from any other in at most three steps (three synapses); most are reachable in just two.

External input

Each neuron is provided with external input by adding Gaussian noise to its membrane voltage. Every time step (Δt = 0.1 ms), a sample drawn from a normal distribution with mean -0.5 pA and $\sigma = 5$ pA is added to the membrane current. (As membrane current is by convention negative, this corresponds to an on-average positive influence on membrane voltage).

EI balance

Similar to the N-to-1 experiments, we make 1 out of 5 neurons inhibitory. As before, this is done by setting the synaptic reversal potential at the outputs of inhibitory neurons to -80 mV (instead of the 0 mV for excitatory neurons). To make sure that each neuron receives a balanced mix of excitation and inhibition, and given that there are 4:1 excitatory to inhibitory neurons, we make excitatory neurons 4x weaker: their synaptic strength (Δg, the instantaneous increase in postsynaptic conductance g on spike arrival) is 4x as small as that of inhibitory neurons.

We also simulate a network with different synaptic weights. Those weights are based on the simulations in Roxin *et al*'s 2011 paper "On the Distribution of Firing Rates in Networks of Cortical Neurons"[6]

[6] This paper seeks to explain how heavy-tailed firing rate distributions emerge in randomly connected spiking neural networks. We wanted to emulate the simulations in this work, to obtain a heavy-tailed firing rate distribution.

Normalizing excitatory-to-excitatory (E→E) connections to 1, Roxin et al's synaptic weights are:

```
E→E: 1
E→I: 18 (instead of 1)
I→E: 36 (instead of 4)
I→I: 31 (instead of 4)
```

We will call these weights 'Roxin2011'; and the former, more classical weights '1144'.

Subsampling

We simulate all 1000 neurons' voltages, but, to save memory and disk space, do not record all these traces.[7] In most of the following results, we recorded the voltage traces of 40 excitatory and 10 inhibitory neurons (5% of all neurons).

Additionally, when performing connection tests on the inputs of a recorded neuron, we do not test the spiketrains of all 999 other neurons. Instead, we test only a (biased) sample of the possible inputs, to save processing time. This sample is constructed as follows. We test all the a-priori known true direct inputs – both excitatory and inhibitory – and add a random sample of 40 not-directly-connected neurons.

On average, each neuron has ± 32 excitatory inputs and ± 8 inhibitory inputs. This means that, from the 1000 × 1000 possible connections, we only test about 4000, or 0.4%.[8]

Connection testing

We use the same connectivity test as in the N-to-1 case before (figure 3): for every tested (`pre`, `post`) neuron pair, the STA of the `post` neuron's voltage is calculated with respect to the spike times of the `pre` neuron, and with respect to shuffled versions of that spiketrain. A connection is then inferred if the peak-to-peak height of the real STA is higher than that of 95% of the shuffled STAs.[9]

The perfomance of this test when applied to the network is shown in figure 6, broken down for the different `pre` and `post` neuron types. We find that we can detect on average 85% of inhibitory inputs, and 57% of excitatory inputs. This difference between I and E is explained by the 4× higher synaptic strength of inhibitory inputs.

We also note the higher-than-expected false positive rates of about 14%, significantly higher than the expected p-value threshold $\alpha = 5\%$. This is expanded upon in the next subsection.

As before, we add noise to the voltage traces, to simulate the effect of voltage imaging (VI), and test its effect on detection performance. In figure 7, we see that for a realistic VI noise level of 10,[10] detection performance stays fairly high. It is only for very high levels of noise that network inference becomes impossible.

[7] For a 10-minute simulation with a timestep of 0.1 ms, one voltage trace takes 48 MB (at 64 bit per sample). Our 1000 neurons thus take 48 GB – and that is just for one simulation (one set of parameters).
We *do* record the spike trains of all neurons. Saving just spike times takes considerably less space: a neuron spiking at 10 Hz for 10 minutes emits 6000 spikes, which, at 64 bit per timestamp, takes just 48 kB.
In other words, at a 0.1 ms sample rate, a spike train occupies about 1000× less memory than the corresponding voltage trace.

[8] Calculation behind these numbers:
- 1000 neurons × 80% excitatory × 4% probability of an input connection = 32 excitatory inputs on average.
- 50 'post' neurons (40 excitatory and 10 inhibitory voltage-recorded neurons) × 80 'pre' neurons (±40 connected + 40 unconnected) = 4000 tested connections.
- To be precise, instead of 1000 × 1000, there are rather $1000^2 - 1000$ possible connections, as we would not test for autapses.

[9] 95% = $1 - \alpha$, where $\alpha = 0.05$ is the p-value threshold, which is expected to be equal to the false positive rate.

[10] A spike-SNR of 10 means that σ_{noise} is one-tenth of spike height.

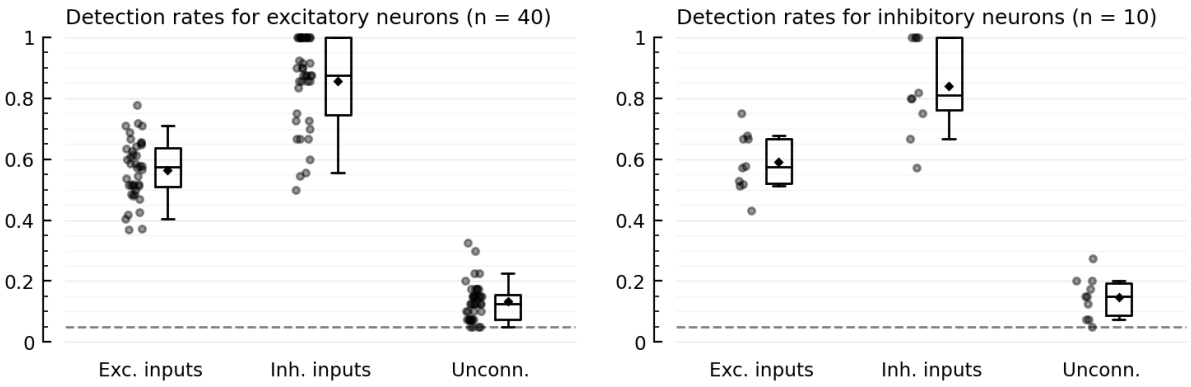

Figure 6: **Performance of the STA 'peak-to-peak' connection test in the random network.** 10 minute recording in the network, with the realistic, '1144' weights. Each dot is the true positive rate for one `post` neuron. The gray dotted line indicates the p-value threshold α of 0.05.

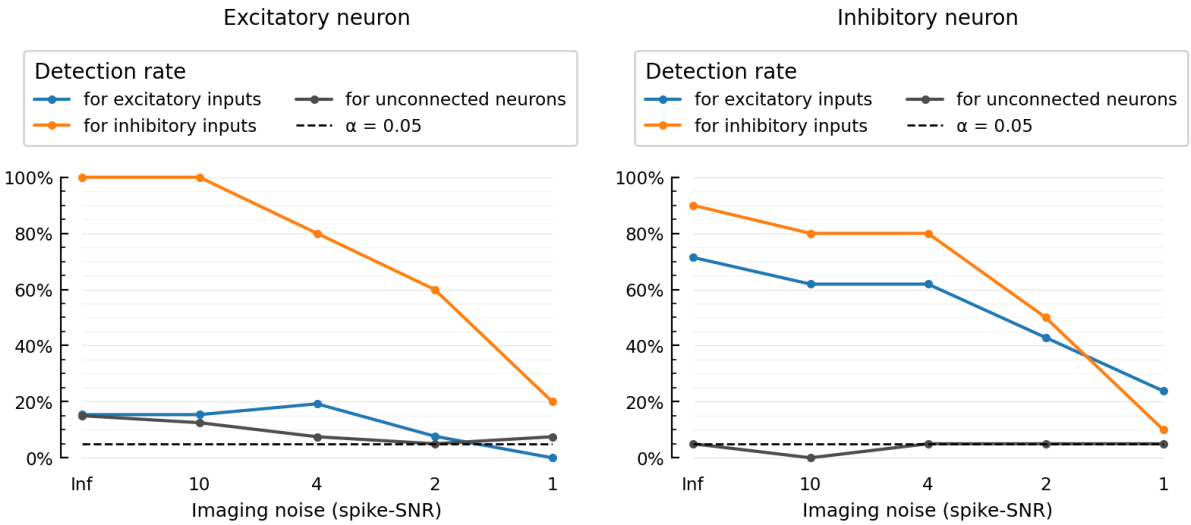

Figure 7: **Network inference only breaks down for very high recording noise.** 10 minute recording in the network with the 'Roxin2011' weights.

Note that figure 7 is for a network with the 'Roxin2011' synaptic weights. Hence why the detection rates at zero noise ('Inf' spike-SNR) are different than in figure 6 (which is also for zero noise, but for a network with the '1144' synaptic weights).

False positive detections

We incorrectly classify more neurons as connected than would be expected from the p-value threshold α = 5% (figure 6) We think that this is due to disynaptic connections: the `pre` neuron that is falsely classified as a direct input would then actually be a strong direct input to intermediary neurons that *are* direct inputs to the voltage-recorded, `post` neuron.

In our fully-randomly connected network, for almost any pair of neurons (A,B) there are many A→X→B (i.e. length-2) paths. So the 'disynaptic hypothesis' for the higher-than-expected false positive rate cannot be tested very directly. However, we did find that falsely detected `pre` neurons had *more* such length-2 paths to the `post` neuron than correctly rejected non-inputs.[11] And in the STAs of the `post` neuron's voltage with the false positive `pre` spiketimes, we find that the 'dip' (the location of the extremum of the 'bump' in the STA) occurs at about 2× the time where it occurs in the STAs of direct inputs (monosynaptic connections) – whereas the location of the dip is random for correctly rejected non-inputs.

[11] Additionally, we find that the intermediate neurons on these length-2 paths spiked more for falsely detected than for correctly rejected `pre` neurons, and that they were more likely to be inhibitory (inhibitory neurons are 4× stronger than excitatory).

4 New connection tests

All previous results used the 'peak-to-peak' STA height as test statistic to determine whether a neuron pair is connected or not. To test whether detection performance can be improved beyond the results found with that method (as seen in figures 6 and 7), we experimented with two new test statistics. (Both new connection testing methods still use shuffled versions of the `pre` neuron's spike train to obtain a null-distribution of the test statistic).

Template correlation

First, we use the Pearson correlation between the STA and a 'template STA' as test statistic (figure 8). An ideal template to correlate with would be the average STA of all true connections in the network.[12] The knowledge of these connections is of course not available a-priori (it is what we are trying to infer). But it turns out we can obtain a signal that is very close to this ideal template with a two-step approach.

[12] Specifically, either all excitatory connections, or all inhibitory connections. When correlating some connection's STA with an 'excitatory' (upwards) template, the sign of the correlation tells us whether the connection is excitatory (+) or inhibitory (−).

In the first step, we use the previous test statistic (peak-to-peak height of the STA); but with a stricter p-value threshold (a higher α). This gives us a sample of true connections with few false positives (but with many missed connections). The STAs of these found connections are then averaged, to obtain an estimate for the STA template. We find that this template matches the ideal template closely (see figure 8, right).

In the second step, we correlate this found template with the STAs of all possible connections (and with their shuffle-control counterparts). The correlation values values are then used as test statistic, with the original $\alpha = 0.05$ threshold.

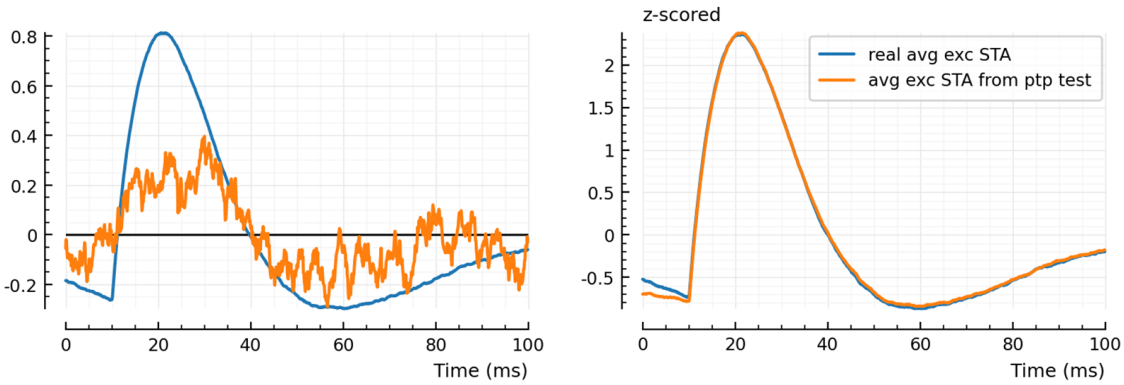

Figure 8: **Correlating the STA with a template.** *Left*: An example STA, and the template it will be correlated with to calculate the connection test statistic. *Right*: Two possible STA templates: in blue, the average of the STAs of all true excitatory connections in the network (or rather, of the excitatory inputs of the 50 neurons that had their voltage recorded). In orange, the average STA of the excitatory connections detected with a strict 'peak-to-peak' test.

Model fitting

Second, we designed a 7-parameter-function f to fit to the STAs.

One part of this STA model function is shaped like a postsynaptic potential bump (PSP, figure 9). It is the impulse response of two linear integrators placed in series. Or, in other words, the convolution of two 'step-and-decay' functions.[13, 14↓] It is the postsynaptic potential in the simplest linear neuron model ($\mathrm{d}v/\mathrm{d}t \cdot \tau_\mathrm{m} = -v + I$) where the synaptic current I is also linear ($\mathrm{d}I/\mathrm{d}t \cdot \tau_\mathrm{s} = -I + s$).[15]

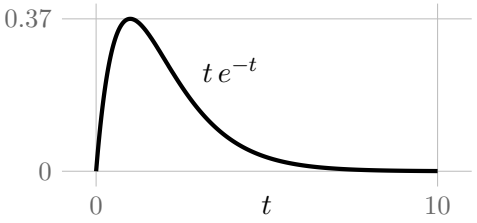

Figure 9: A simple PSP model.

Solving for v and with a single input spike at $t = 0$, we have, for $t \geq 0$:

$$\mathrm{PSP}(t) = \begin{cases} t\,e^{-t/\tau_\mathrm{m}} & \tau_\mathrm{m} = \tau_\mathrm{s} \\ \frac{\tau_\mathrm{m}\tau_\mathrm{s}}{\tau_\mathrm{m}-\tau_\mathrm{s}}\left(e^{-t/\tau_\mathrm{m}} - e^{-t/\tau_\mathrm{s}}\right) & \tau_\mathrm{m} \neq \tau_\mathrm{s} \end{cases} \tag{1}$$

We find that this function approximates the shape of the *simulated* postsynaptic potential in our actual neuron model well – even though our neuron model is not so linear.[16]

But even though equation (1) models our simulated *PSPs* well, it does not resemble our *STAs*[17]; see for example the average STAs in figure 8. First, the bump in the STA does not occur immediately after the `pre`-neuron's spike. This is due to the simulated axonal transmission delay, and it is easily replicated in the model by shifting the PSP function in time. But more interestingly, the STA shows a sort of 'dip', where it dives below baseline just after the PSP bump, and flares up again at the ends (a downwards slant in the initial delay period, and an upwards slant at the end of the STA).

These phenomena thus differentiate an STA from a PSP. We find empirically that they can be parsimonously modelled by subtracting a broad Gaussian curve from the PSP: see figure 10, right.[18]

Our model f is thus the difference of a delayed version of equation (1), and a Gaussian curve. The result is scaled by some factor α, to match the size of the specific STA that is being fit. α can be negative, to model inhibitory connections; f is then flipped upside-down, as in figure 10.

The model function thus has seven parameters: the PSP's delay, τ_m, and τ_s; the Gaussian's location, width, and relative height wrt. the PSP; and the final scaling α. All parameters, except for the Gaussian's, have a ready biophysical interpration.[19]

We fit the model function to individual STAs using nonlinear least-squares optimization; more specifically using the Levenberg-Marquardt

[13] 'Step-and-exponential-decay': $u(t) \cdot e^{-t/\tau}$, with $u(t)$ the Heaviside step function $\mathbf{1}_{t\geq 0}$.

[14] In computational neuroscience, this function is sometimes called the 'alpha function' or 'alpha synapse'. *Synapse*, because the double integrator model can be used to model synaptic conductance (g), with then a fast rise τ_1 (modelling neurotransmitter release), and a slower decay τ_2 (modelling its dissipation). The synaptic conductance model in our simulation is simpler: the 'rise' is instant, i.e. we only have exponential decay.

[15] τ_s and τ_m are the synaptic and membrane time constants, and $s(t) = \sum_i \delta(t - t_i)$ is the train of input spikes i.

[16] Our model for the membrane potential (the Izhikevich neuron – see year 1 report) is *quadratic* in v, and has a component (the 'adaptation current', u) whose differential equation recurrently depends on v. Additionally, the synaptic currents I_j are *conductance-based*, meaning that they also recurrently depend on v: $I_j = g_j\,(v - E_j)$ (with E_j the reversal potential at synapse j, and the synaptic conductance g_j a linear integrator of the input spike train $s_j(t)$: $\mathrm{d}g_j/\mathrm{d}t \cdot \tau_{s,j} = -g_j + s_j$).

[17] This falsifies our initial implicit hypothesis that STAs are merely noisy reflections of PSPs.

[18] Note that in figure 10, we fit an *inhibitory* STA, and the model componentes are thus inverted: the delayed PSP bump (blue) is downwards, and the Gaussian 'dip' (orange) is upwards.

[19] Namely: α is proportional to the connection strength, and indicates excitation (+) or inhibition (−). The PSP's delay, τ_m, and τ_s are estimates of the axonal transmission delay and the synaptic and membrane time constants.

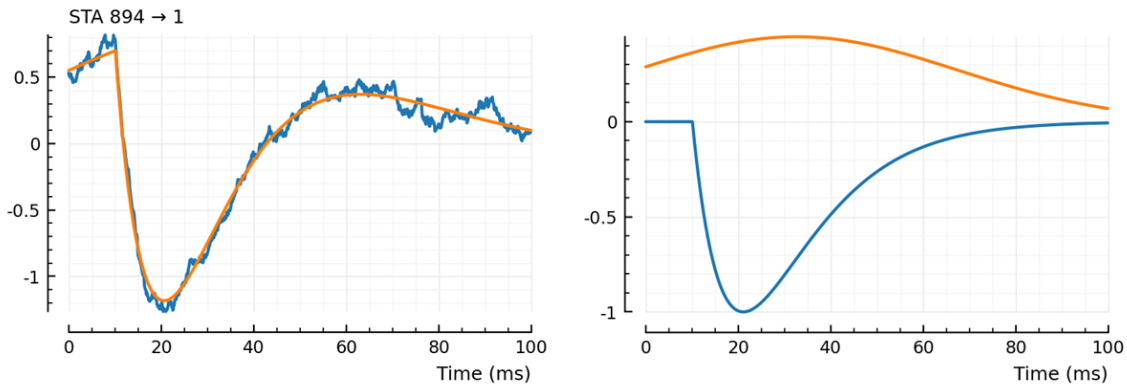

Figure 10: **Fitting a parametric model to the STA.** *Left*: the model f (orange) fit to the STA of an example inhibitory connection from the network simulation (blue). Both signals are z-scored. The components of the orange model function are shown on the *Right*: the delayed simple PSP model (blue) and the Gaussian 'dip' (orange).

algorithm.[20] To make the fitted functions well-behaved and looking like actual STAs, we had to limit the creative freedom of the optimization algorithm, by enforcing box constraints on the parameters.

The final test statistic is the mean squared error (MSE) between z-scored versions of the fitted function and the STA. The normalization by z-scoring is necessary to be able to compare goodness of fit between STAs with different ranges[21]

Performance

In a preliminary comparison, we find that the template-correlation approach ('corr' in the below table) performs significantly better at network inference than the simple STA height test ('ptp'):

Test	ptp	corr	model
Recall	**63%**	**84%**	67%
Precision	81%	86%	83%

Table 1: Network inference performance for different test statistics, at false positive rate $\equiv$ 15%. 'ptp' is the simplepeak-to-peak STA height statistic of figure 3. 'corr' is template correlation (with template from an initial, strict 'ptp' step). 'model' is using the STA model fit.

For example, at a false positive rate of 15%, template correlation detects **84%** of all connections (and 86% of its detections are correct), whereas the peak-to-peak test detects only **63%** (with 81% of its detections correct).

The model-fitting approach, on the other hand, does not perform much better than the peak-to-peak test. Though, we must note that its precision for inhibitory connections specifically is significantly higher than that of the peak-to-peak test: 68% of inhibitory detections are correct for model-fitting, versus 47% for peak-to-peak. (Template correlation has 69%). This difference is obscured in the above aggregate measures (which are for both connection types together), as there are 4× as many excitatory as inhibitory connections.

[20] This is the standard method for nonlinear least-squares optimization. For example, in MATLAB's `lsqcurvefit` function, it is the default algorithm when the number of functions (in our case, 1) exceeds the number of datapoints (in our case, 1000: 100 ms of signal post-spike, at $\Delta t = 0.1$ ms).
We used the Levenberg-Marquardt implementation of the Julia package LsqFit.jl.

[21] Most importantly between a real STA and its shuffled versions. Because those shuffled STAs have a smaller range, they will have a lower MSE, even though the fit is visually clearly worse.

5 Plan for completion

Thesis chapters

The **Introduction** chapter will be written near the end, as it signposts and summarizes the rest of the thesis.

I will start with two **Literature Review** chapters, one on voltage imaging and one on network inference methods. These chapters will draw on the literature reviews I have done at the start of the PhD.

Next, three **Methods** chapters. The first one will describe and give examples of the simulations: networks of Izhikevich neurons, from which voltage traces are recorded, which are corrupted by noise to simulate voltage imaging. The second chapter is shorter and describes how network inference methods are evaluated and compared. The third chapter describes the network inference methods (i.e. how they work. Descriptions and illustrations of the algorithms).

Next is a large **Results** chapter, in which the different network inference methods are applied to different network simulations. The inference performance is compared across methods and simulation types, and for varying simulation parameters.

Finally, a **Discussion & Conclusions** chapter, to synthesize the results and put them in a broader context, and to discuss ideas for further work.

Timeline

I plan to submit my thesis by the end of next year. This is a rough sketch of a timeline for the chapters described above:

- *February*: Literature reviews
- *March*: Methods – Simulations
  (Izhikevich neuron, N-to-1 problem, recurrent network)
- *April*: Methods – Performance quantification
- *June*: Methods – Inference algorithms
- *August*: Results
- *September*: Discussion & Conclusions, Introduction
- {editing & integrating feedback}
- *December*: Submission

As my PhD funding ends in February, I plan to work part-time on the thesis, and do software contracting work the other half of the time.

In addition to the longer-term chapter timeline, these are the immediate next steps for the coming weeks:

- There was a bug in the N-to-1 code for big N (see section 2). The specific analysis where the N-to-1 experiment was run for increasing N thus needs to be redone.
- Continue work on new connection tests, using the fitted STA model.
  - Instead of shuffle null-distribution, use a simpler model (like regressing a straight line) as null-model, and compare that with the 'full' 9-parameter functional model.
  - Instead of comparing models or test statistic distributions, use the space of fitted model parameters. Connected versus not-connected STAs will occupy different regions in this space; so we might use a **clustering** algorithm as network inference method.