Cleo: A testbed for bridging model and experiment by simulating closed-loop stimulation, electrode recording, and optogenetics

Kyle A. Johnsen[a], Nathanael A. Cruzado[b], Adam A. Willats[a], Christopher J. Rozell[b,*]

[a] *Georgia Institute of Technology and Emory University, Coulter Department of Biomedical Engineering, 313 Ferst Drive, Atlanta, USA, 30332*

[b] *Georgia Institute of Technology, School of Electrical and Computer Engineering, 777 Atlantic Drive NW, Atlanta, USA, 30332-0250*

Abstract

Recent advances in neurotechnology enable exciting new experiments, including novel paradigms such as closed-loop optogenetic control that achieve powerful causal interventions. At the same time, improved computational models are capable of reproducing behavior and neural activity with increasing fidelity. However, the complexities of these advances can make it difficult to reap the benefits of bridging model and experiment, such as *in-silico* experiment prototyping or direct comparison of model output to experimental data. We can bridge this gap more effectively by incorporating the simulation of the experimental interface into our models, but no existing tool integrates optogenetics, electrode recording, and flexible closed-loop processing with neural population simulations. To address this need, we have developed the Closed-Loop, Electrophysiology, and Optogenetics experiment simulation testbed (Cleo). Cleo is a Python package built on the Brian 2 simulator enabling injection of recording and stimulation devices as well as closed-loop control with realistic latency into a spiking neural network model. Here we describe the design, features, and use of Cleo, including validation of the individual system components. We further demonstrate its utility in three case studies using a variety of existing models and discuss potential applications for advancing causal neuroscience.

1. Introduction

Mesoscale neuroscience is currently undergoing a revolution fueled by advances in neural manipulation [1, 2, 3, 4, 5, 6, 7, 8] and measurement [9, 10, 11, 12, 13, 14,

*Corresponding author
Email address: crozell@gatech.edu (Christopher J. Rozell)

15, 16] technologies as well as data analysis methods [17, 18, 19, 20, 21, 22]. These have yielded unprecedented datasets [23, 24] and insights into network activity [25, 26, 27, 28, 29], as well as novel experimental paradigms such as direct closed-loop control of neural activity [30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41]. At the same time, models from cognitive, computational, and theoretical neuroscience have grown both in their computational power and their concordance with experimental data [42, 43, 44, 45, 46, 47].

Building bridges between cutting-edge experiments and powerful models could enable these parallel lines of research to better inform and inspire each other. However, modern computational neuroscience models are rarely considered in light of the experimental considerations that would constrain measurement and manipulation of the system being modeled, making it difficult to fully bridge between theory and experiment. Having a widely applicable framework for this type of integrated modeling that is informed by the constraints and idiosyncrasies of experimental interfaces would provide at least two benefits. First, this approach provides a testbed for prototyping complex experiments, accelerating the research process (see Figure 1b). This is especially crucial in closed-loop experiments where the experimental apparatus is reacting to the neural system in real time. Second, this approach facilitates the the direct comparison of a model to experimental data in the context of the experimental system used to collect the data (see Figure 1c). This is especially important with increasingly complex models where indirect comparisons (i.e., comparing explicit measurements to latent model components without the lens of measurement systems, or not considering the effects of experimental perturbations on the model) can be especially difficult to interpret.

The increasing complexity of both experiments and models requires specialized software to meet this goal, creating barriers to access. While multiple existing tools facilitate some degree of stimulation and recording of high-level population simulations [48, 49, 50, 51, 52, 53], existing approaches have crucial limitations. For example, many are oriented towards detailed, multi-compartment neuron models that can be prohibitively expensive for the simulation of large populations, and none offer ready-to-use light and opsin models for optogenetics. Moreover, none support flexible closed-loop control with the important factor of real-time processing latency incorporated into the simulation. Closed-loop feedback control experiments can especially benefit from simulation because the interaction of recording, stimulation, and control parameters makes the impact of design choices less predictable than in passive or open-loop stimulation experiments.

To address this crucial need, the main contribution of this paper is the development of Cleo: the Closed Loop, Electrophysiology, and Optogenetics experiment simulation testbed. Built around the flexible and intuitive Brian simulator [54], Cleo integrates arbitrary closed-loop signal processing, recording, and stimulation devices that can be used in combination with existing network models to simulate passive (i.e., recording only), open-loop stimulation, or closed-loop stimulation experiments (see Figure 1a). Cleo currently implements electrode

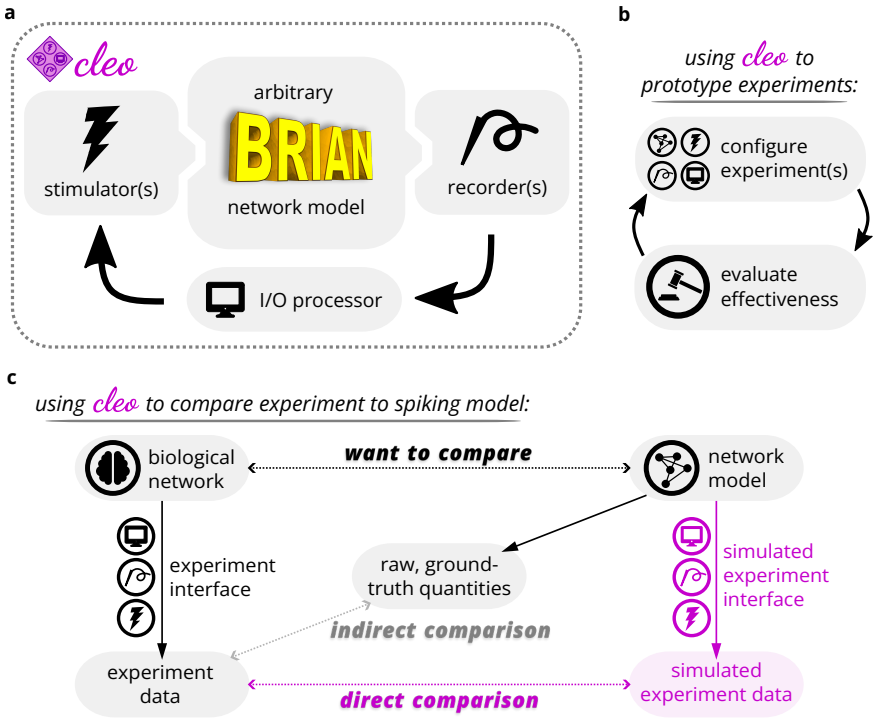

Figure 1: The architecture and potential applications of Cleo. (A) A conceptual diagram of Cleo's functionality. Cleo wraps a Brian network model, injects stimulation and recording devices, and interfaces with the network in real time through a simulated "I/O processor" to capture (potentially closed-loop) processing delays. (B) A diagram of how Cleo can be used to prototype an experiment. Users can configure stimulation, recording, and potentially closed-loop control to identify which scenarios and parameters best answer their scientific question. They might also test variations on the network models to plan experiments that are robust to unknowns or that best distinguish between alternatives. (C) An illustration of Cleo's usefulness for validating a spiking network model against experimental data. By simulating the measurement and manipulation of the underlying neural activity, Cleo produces simulation results that are more directly comparable to electrophysiology experiments.

recording, light, and opsin models for photostimulation, while using a modular design that allows for future addition of other modalities. Here we describe the design, features, and use of Cleo, including validation of the individual system components. We further demonstrate its utility in three case studies using a variety of existing models, and discuss potential applications for advancing causal neuroscience. The three case studies span a variety of use cases, including closed-loop inhibition of a traveling wave in sensory cortex, dynamic clamping of firing rate to disrupt visual cortex plasticity, and sharp wave-ripple evocation in the hippocampus.

2. Materials and Methods

2.1. Architecture and design rationale

A Cleo experiment consists of (1) setting up a Brian spiking neural network model, (2) specifying the recording apparatus, (3) specifying the stimulation apparatus, and (4) configuring an I/O processor to control stimulation devices. Cleo provides modules for integrating the latter three parts with an existing Brian model (see Figure 1a). This integration is performed by a `CLSimulator` object that orchestrates the experiment by injecting devices, running the Brian simulation, and communicating with an `IOProcessor` object at each time step. The `IOProcessor` receives recorder measurements according to a user-specified sampling schedule and returns any updates to stimulator devices. Below, we describe the principles and assumptions that guided our modeling and software choices.

Two factors drove our choice of recording and stimulation models to integrate into Cleo. First, because Cleo's purpose is to simulate experiments, we focused on models at the level of accessible experimental parameters. Because parameters such as electrode location, channel count and density, and optic fiber depth and size are all defined naturally in space, Cleo's electrode and optogenetics modules require a spatial network model where relevant neurons are assigned x, y, and z coordinates. Second, we assumed that highly realistic models of individual neurons are not needed to capture many interesting mesoscale-level phenomena. Accordingly, Cleo was developed with the intention of using point neuron models, rather than multi-compartment neuron models with realistic morphologies. Simulating simpler neuron models is faster in terms of both development and simulation, which is important in the context of prototyping an experiment. This decision had consequences in our software and modeling decisions (see Section 2.2, Section 2.3, Section 2.4).

In addition to our modeling priorities, the goals of usability, flexibility, and extensibility guided our choices in software dependencies and infrastructure. Ease of use is important to make Cleo as accessible as possible, especially to researchers with primarily experimental backgrounds. This usability goal also motivated Cleo's modular design, which allows users to add different recording or stimulation devices with little or no modification to the underlying network

model, easing the burden of testing a variety of experimental configurations. Flexibility in the underlying simulator, in addition to enabling compatibility with a wide variety of models, was a necessity for arbitrarily interacting with the simulation in a closed-loop fashion. Finally, we endeavored to make Cleo extensible so it could be adapted to use cases beyond the optogenetic stimulation and electrode recording modules provided upon release. This motivated the modular "plug-in" pattern inherent in the architecture and enables development of new experimental interfaces (e.g., microstimulation, fluorescence microscopy) in future modules. In the following sections we describe the specific infrastructure and modeling choices we made in accordance with this rationale.

2.2. Software infrastructure decisions

Other tools in the spirit of experiment simulation exist, though none with the collection of goals and functionality of Cleo. One is Mozaik [48], which can manage stimulation and recording parameters as well as data and visualizations, running on the simulator backend-agnostic PyNN interface [55]. It has been used to prototype an experiment [56], but PyNN does not support arbitrary interaction with the simulation while it is running, which is needed to simulate closed-loop control. Three more (BioNet [51, 52], NetPyNE [53], and LFPy [57, 58]) include some of the features we needed, but as front-ends to the NEURON simulator [59] they are oriented towards biophysically detailed, expensive-to-simulate neuron models. The same can be said of VERTEX [49, 50], which is a tool for use in MATLAB. See Table 1 for details.

Table 1: Feature comparison of experiment simulation software

Software	Simulator	Target model type	Closed-loop stimulation	Ready-to-use opsin model	Ready-to-use light model	Extracellular recording
Mozaik [48]	Multiple (through PyNN)	point		X		
LFPy 2.0 [58]	NEURON	morphological		X		X
BioNet [51, 52]	NEURON	morphological	X	X		X
NetPyNE [53]	NEURON	morphological		X		X
VERTEX 2.0 [49, 50]	Custom MATLAB	morphological	X (inflexible)			X
Cleo	Brian 2	point	X	X	X	X

Between the two most widely used spiking neural network simulators optimized for point neurons (Brian 2 [54] and NEST [60]), we chose Brian for its intuitiveness and flexibility. It allows (and even requires) the user to define models mathematically rather than selecting from a pre-defined library of cell types and features, while maintaining the ease of a high-level interface. This keeps model and experiment details together and enabled us to define arbitrary recording and stimulation models that easily interface with the simulation. Moreover, Brian users only need to know Python: a programming language with the advantages of being open-source, intuitive to learn [61], and widely used in computational neuroscience [62, 63]. Also, Brian has the potential to run quickly via compilation to C++ [54] or GPU acceleration [64].

2.3. Optogenetics models

Cleo simulates optogenetic stimulation by combining a model of light propagation with an opsin model relating light to current. The light model is based on Kubelka-Munk light propagation, operating on the assumption that the medium is optically homogeneous and that particles are larger than the light wavelength [65, 66]. Cleo includes absorbance, scattering, and refraction parameters for 473-nm (blue) light as given in [65], but these are easily updated by the user for other wavelengths.

Independent of the light propagation model, Cleo provides two different opsin models. One is a four-state Markov model as presented in [67]. This model captures rise, peak, plateau, and fall dynamics of the photocurrent as opsins are activated and deactivated through a Markov process. By defining conductance rather than current directly, this model is also able to reproduce the photocurrent's dependence on the membrane potential (see Figure 5). While the four-state model fits experimental data fairly well, the code is structured so that three- or six-state models could also be easily implemented. Cleo provides default parameters for ChR2, as given in the PyRhO simulation package [67].

However, because the Markov model depends on somewhat realistic membrane potential and resistance values, it is not well suited for many pre-existing models that do not. For example, many commonly used leaky integrate-and-fire (LIF) neurons define the membrane potential as ranging from 0 to 1, rather than -70 mV to -50 mV, rendering both the units and values (relative to the opsin's reversal potential) incompatible. While one could adapt neuron models for compatibility with this Markov opsin model, to minimize user burden we also developed an alternative model that delivers photocurrent proportional to the light intensity at each neuron. Specifically, we offer an optional model of the opsin current described with

$$I_{\text{opto}} = k * \text{Irr} * \rho_{\text{rel}}$$

where k is an arbitrary gain term, Irr is the irradiance of the light source at the location of the neuron with unit mW/mm^2, and $\rho_{\text{rel}} \geq 0$ is the relative opsin

```python
from brian2 import Network, NeuronGroup, mm
import cleo
from cleo import ephys, opto, ioproc

ng = NeuronGroup( # a simple population of 100 LIF neurons
    100,
    """dv/dt = (-v - 70*mV + (500*Mohm)*Iopto) / (20*ms) : volt
    Iopto : amp""",
    threshold='v>-50*mV',
    reset='v=-70*mV'
)
cleo.coords.assign_coords_rand_rect_prism(
    ng, xlim=(-0.2, 0.2), ylim=(-0.2, 0.2), zlim=(0.7, 0.9), unit=mm
)
sim = cleo.CLSimulator(Network(ng))

# configure and inject a 32-channel shank
coords = ephys.linear_shank_coords(
    array_length=1*mm, channel_count=32, start_location=(0, 0, 0.2)*mm
)
mua = ephys.MultiUnitSpiking(
    name="mua",
    perfect_detection_radius=0.05 * mm,
    half_detection_radius=0.1 * mm,
)
probe = ephys.Probe("probe", coords, signals=[mua])
sim.inject_recorder(probe, ng)

# configure and inject optogenetic stimulation
stim = opto.OptogeneticIntervention(
    name="stim",
    opsin_model=opto.FourStateModel(opto.ChR2_four_state),
    light_model_params=opto.default_blue,
    location=(0, 0, 0.4) * mm,
)
sim.inject_stimulator(stim, ng)

# configure closed-loop control with arbitrary code
class MyProcessor(ioproc.LatencyIOProcessor):
    def process(self, state_dict, sample_time_ms):
        # state_dict contains a {'recorder_name': value} dict of network.
        i_spikes, t_ms_spikes, y_spikes = state_dict['probe']['mua']
        # on-off control
        irr0_mW_per_mm2 = 10 if len(i_spikes) < 10 else 0
        # output is a {'stimulator_name': value} dict and output time
        return {'stim': irr0_mW_per_mm2}, sample_time_ms + 3  # (3 ms delay)

sim.set_io_processor(MyProcessor(sample_period_ms=1))

# visualize experimental configuration
cleo.viz.plot(
    ng, colors=['#c500cc'], devices=[probe, (stim, {'n_points': 7e4})], zlim=(0.2, 1)
)
```

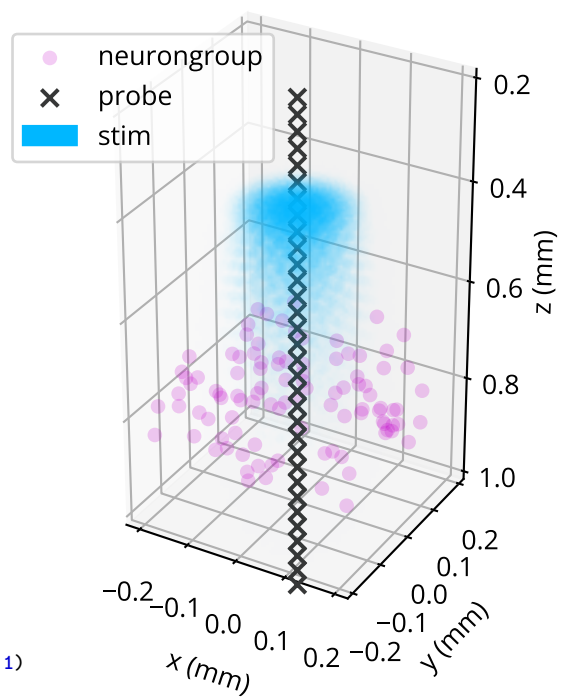

Figure 2: Code sample and plot illustrating basic Cleo experiment configuration. Note how few lines are needed to simulate multi-channel electrode recording, optogenetic stimulation, and delayed closed-loop control starting with a pre-existing Brian `Network` model.

expression level (the default value of 1 corresponding to the standard model fit). Note that k is expressed in $[unitof I_{\text{opto}}] * \text{mm}^2/\text{mW}$, adapting to the units of I_{opto}. This model allows users to retain their neuron model wth parameters and units unchanged, since they can adapt the k term to whatever scale and units are needed. Preliminary experiments show that this simplified opsin model (see Figure 9) can produce responses that are similar in many respects to those of the four-state Markov model.

In addition to options for opsin modeling, Cleo allows the user to specify both the probability that cells of a target population express an opsin and the per-cell expression level (via the afore-mentioned ρ_{rel} parameter). Users can thus study the impact of heterogeneous opsin expression on the outcome of an experiment.

2.4. Electrode recording models

2.4.1. Spiking

Because we have prioritized point neuron simulations, the electrode functionality currently implemented in Cleo does not rely on raw extracellular potentials that could only be computed from multi-compartment neurons [68, 69]]. To approximate spike recording without filtering and thresholding of extracellular potentials, Cleo simply takes ground-truth spikes (returned by the Brian simulator) and stochastically determines which to report as recorded on the electrode. The probability a given spike is detected by an electrode is a function of r, the distance between the neuron and the electrode. This function is parametrized by a perfect detection radius (where all spikes are reported), a half detection radius (where there is a 50% chance a spike will be detected), and a cutoff radius (where all neurons are ignored). The detection probability function is interpolated between the parametrized points with a $1/r$ function [70, 71] (see Figure 3a). The user may refer to studies such as [72] to determine reasonable detection distance parameters.

Cleo provides spike recording functionality in two forms: multi-unit and sorted (see Figure 3c). Multi-unit spiking reports every spike detected by every channel, without regard for the origin of the spike. Thus, each channel can report spikes from multiple neurons and a single spike can be reported on multiple channels. Sorted spiking, on the other hand, reports all spikes detected on at least one channel, where each neuron is identified by a unique index. Because point neurons cannot provide the raw extracellular potential waveforms needed for spike sorting algorithms, we approximate the spike sorting process by assuming perfect sorting. While real-time spike sorting is currently not feasible in practice for large channel counts, this sorted spiking option could be used to emulate a workflow of isolating one or a few neurons to record spikes from in real time.

2.4.2. LFP

To approximate cortical LFP without resorting to morphological neurons and biophysical forward modeling, we implemented what we term the Teleńczuk kernel LFP (TKLFP) approximation [73] (see Figure 3d). This method approximates

the per-spike contribution to LFP with a delayed Gaussian function, where amplitude and delay depend on the position of the neuron relative to the electrode, as well as cell type (excitatory or inhibitory). Default parameters (taken from the original study) were estimated from human temporal cortex experimental data and from hippocampus simulations. As the authors indicate, parameters may need refinement on a per-region basis.

While the original study included reference peak amplitude (A_0) values at just four cortical depths, we inferred these values for arbitrary depths by performing cubic interpolation on reported data (see Figure 5 in [73]) and assumed that this profile dropped to zero at 600 µm below the soma and 1000 µm above. This implementation is available as a standalone Python package [74].

We point out that an alternative method for approximating LFP for point neurons (not yet included in Cleo) is described by [75], using a weighted sum of excitatory and inhibitory synaptic currents as an LFP proxy. The method appears to better capture high-frequency signals, but requires more detailed neuron models.

2.5. Latency model

To simulate the effects of real-time compute latency, Cleo provides a `LatencyIOProcessor` class capable of delivering control signals after arbitrary delays. It does this by storing the outputs calculated for every sample in a buffer along with the time they can be delivered to the network. For example, if a sample is taken at time $t = 20$ ms and the user wishes to simulate a 3-ms delay, the control signal and output time (23 ms) are stored in a buffer which the simulation checks at every time step. As soon as the simulation clock reaches 23 ms, the control signal is taken from the buffer and applied to update the stimulator devices. Because the user has arbitrary control over this latency, they can easily simulate the effects of closed-loop experimental constraints. For example, one could use probabilistic delays to assess the effect closed-loop algorithms with variable round-trip times between measurement and stimulation. By default, `LatencyIOProcessor` samples on a fixed schedule and simulates processing samples in parallel (i.e., the computation time for one sample does not affect that of others). This and the other sampling schemes Cleo provides are illustrated in Figure 4.

2.6. Case study computing environment

Case study 1 was run on Georgia Institute of Technology's Partnership for an Advanced Computing Environment (PACE) Phoenix Cluster with 40 GB RAM and 1 node (and 2 cores per node) allocated to it. In the Phoenix Cluster, each node features dual Intel Xeon Gold 6226 CPUs @ 2.7 GHz (24 cores/node), DDR4-2933 MHz DRAM, and Infiniband 100HDR interconnect. Case study 2 was also run on the PACE Phoenix Cluster with 64 GB RAM and 1 node (and 2 cores per node) allocated to it. Case study 3 was run with Python 3.7 on a Dell laptop with an Intel i9-9980HK CPU @ 2.40GHz processor with 8 cores and 32 GB RAM.

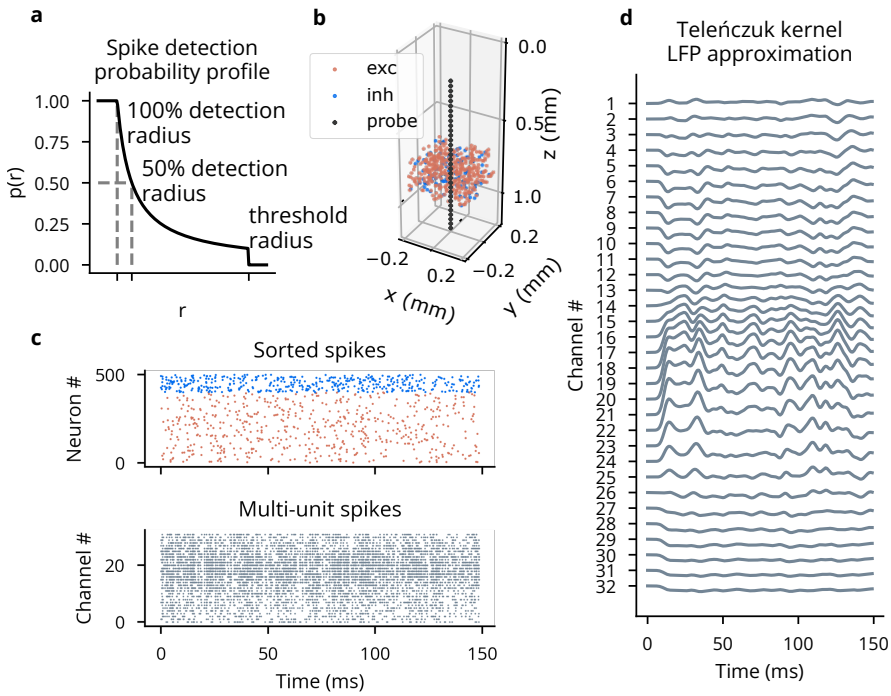

Figure 3: Illustration of LFP and spiking from Cleo's electrophysiology module. (A) The probabilistic spike detection model. All spikes within the 100% detection radius, 50% of spikes at the 50% detection radius, and none of those outside the threshold radius are recorded. The detection probability decays with $1/r$. (B) An example plot generated by Cleo showing the positions of neurons and electrode contacts. (C) Randomly generated spikes for the neurons shown in panel B. Top: the sorted spike signal, which gives the ground truth source neuron for every spike as a perfect proxy for spike sorting. Bottom: multi-unit spikes, where spikes are reported on every channel they are detected on, regardless of the source neuron. (D) The TKLFP approximation of local field potentials generated from the spikes in panel C for each channel.

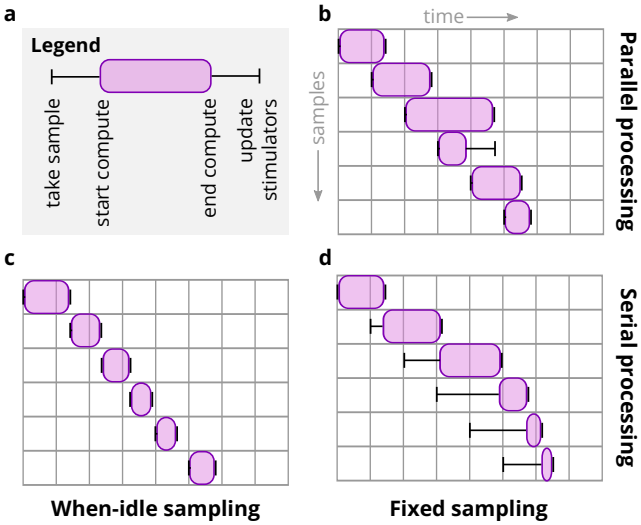

Figure 4: Latency emulation strategy and available configurations. (A) Cleo registers the time a sample is taken from the recording devices, determines the times the computation starts and ends, applies the user-specified delay, and updates stimulation devices when finished. (B) The default parallel processing/fixed sampling mode. Updates are reserved until the previous update is delivered so the sequence of stimulator updates corresponds to the sequence of measurements. (C) The "when-idle" processing mode samples only once the computation for the previous step has terminated. (D) The serial processing/fixed sampling case reflects when computations are not performed in parallel, but sampling continues on a fixed schedule. Samples are taken either as soon as possible after the previous sample time was missed, or on schedule otherwise.

2.7. Feedback control

Case study 2 uses proportional-integral (PI) control and exponential firing rate estimation as described in [35]. Cleo provides implementations of these, which can be found in the `cleo.ioproc` module. Case study 3 used a standard linear quadratic regulator approach as described in [36] and implemented in the ldsCtrlEst C++ library [76]. ldsCtrlEst is part of CLOCTools [77, 78], a larger collection of algorithms and utilities for implementing closed-loop optogenetics in real-time lab experiments.

3. Results

We demonstrate the utility of Cleo with a variety of different results. In particular, we first present a number of results from the Cleo optogenetics and recording modules (LFP simulation in particular) that have direct comparisons to the published literature. These results validate that the modules present in the current Cleo package are suitable for a wide variety of realistic model simulations. We then present results on a three case studies where we use Cleo to perform realistic simulations in a variety of experimental paradigms on models that have been previously reported in the literature.

3.1. Optogenetics model validation

To validate the light and opsin models, we first reproduced within the Cleo system previously reported results on the optic fiber light transmission characteristics. Specifically, transmittance refers to the irradiance Irr as a proportion of the irradiance at the fiber tip Irr_0. Figure 5 demonstrates this transmittance corresponds to previously reported results as a function of radius and axial distance from the optic fiber tip (cf. panel A and Figure 2a of [65]) and distance z straight out from the fiber tip (cf. panel B and Figure 2b of [65]). We also examined the photocurrent in response to the ramping light intensity (cf. panel C and Figure 4c of [67]).

To test how well simplified models produce realistic firing patterns on long timescales, we also compared pulse rate to firing rate for a variety of light intensities and opsin expression levels (similar to previous studies with multi-compartment Hodgkin-Huxley neurons in [65]). We used combinations of leaky integrate-and-fire (LIF) and adaptive exponential integrate-and-fire (AdEx) [80] neuron models, along with proportional current and Markov opsin models. AdEx neurons had parameters as given by [79] for a tonic firing pattern, and irradiance was simulated at 120% of the single-spike threshold. As expected, the different model combinations behave differently and none reproduce exactly more detailed biophysically realistic simulations [65]. Specifically, though both reproduce the linear relationship at lower pulse rates, approximation accuracy is worse at capturing the sublinear relationship at higher pulse rates.

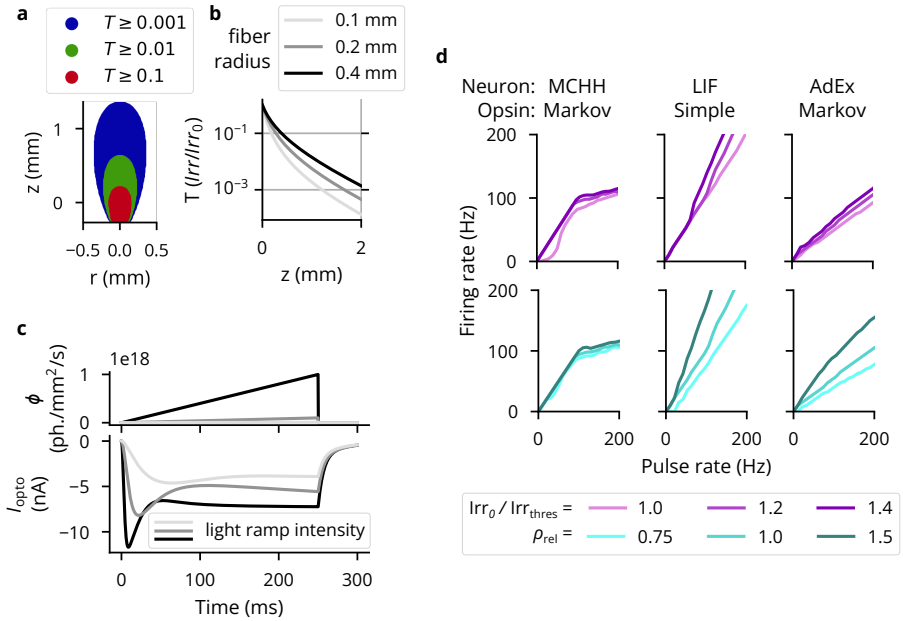

Figure 5: Validation of the optogenetics module. (A) Light transmittance T as a function of radius and axial distance from the optic fiber tip (cf. Figure 2a from [65]). (B) Light transmittance T as a function of distance z straight out from the fiber tip for different optic fiber sizes (cf. Figure 2b from [65]). (C) Photocurrent I_{opto} for ramping light of different intensities (cf. Figure 4c of [67]). (D) Neuron firing rates in response to optical stimulation with 5-ms pulse frequencies ranging from 1 to 200 Hz. The left column re-plots data from [65]. The middle column shows results for an LIF neuron with a simple opsin, and the right column for a tonic AdEx neuron [79] with a Markov opsin model. The top row shows results for different light intensities: 100%, 120%, and 140% of the threshold for producing a single spike with a 5-ms pulse. The bottom row shows results for different expression levels relative to the default, ρ_{rel}.

3.2. LFP Model validation

To test the Teleńczuk kernel LFP approximation module, we compared our results to the demo presented by [73]. We found that the Cleo implementation of TKLFP produced essentially the same result as previously reported (see Figure 10). We also compared the TKLFP output of the hippocampus model to the reference weighted sum method of [75] and found the two to be qualitatively similar (see Section 3.5, Figure 11). Though the TKLFP clearly underrepresents some high-frequency components (as reported in the original publication), this qualitative evaluation demonstrates that the current Cleo LFP simulations are likely suitable for a variety of modeling applications.

3.3. Case study 1: Closed-loop rejection of an S1 traveling wave

We first implemented a Brian 2 simulation of a rodent S1 traveling wave model [81] to demonstrate Cleo's capabilities for simulating an event-triggered closed-loop control experiment. The model of rodent S1 in [81] uses a mix of excitatory and inhibitory neurons (12,500 total) with weak local connections and a sparse sub-network with stronger connections. The neurons lie in a $5\,\mathrm{mm} \times 5\,\mathrm{mm}$ sheet, and we adjusted the initial state and input of the center $1\,\mathrm{mm}^2$ patch to produce a sparse traveling wave of spreading activation in response to an initial stimulus as reported in the original publication.

We configured Cleo to simulate an experiment with an "optrode" (a combined electrode and optic fiber) to trigger inhibitory optogenetic stimulation when recorded multi-unit activity reached 3 or more spikes over the previous 1-ms sampling period (illustrated in Figure 6a). We used the previously described simple opsin model to accommodate the neuron model's normalized, non-biophysical parameters and adjusted the optogenetic stimulus through trial and error to a level sufficient to suppress activity around the optic fiber. This simulation used 12,500 neurons over 15 simulated milliseconds, and required approximately 30 seconds of runtime (with or without Cleo integration) in a cluster computing environment.

As seen in Figure 6b,c, the optogenetic stimulation suppresses neural activity, effectively quenching the traveling wave in the region around the optrode. This demonstrates the use of Cleo in simulating basic "reactive" or "event-triggered" control where a predetermined stimulus is presented in response to a detected feature in the electrophysiology recording. In general, this sort of closed-loop control might be used to either inhibit [82, 39] or amplify said feature. This case study also shows how Cleo can easily record from and stimulate even simplified neuron models as long as they spike and include an input current term.

3.4. Case study 2: Clamping firing rate to disrupt plasticity in V1

We designed and simulated an experiment using a V1 layer 2/3 model of plasticity [83] to demonstrate Cleo's closed-loop feedback control capabilities. A Brian 2 implementation of the model was publicly available on ModelDB [84] and required only the minor modification of assigning coordinates to neurons (random

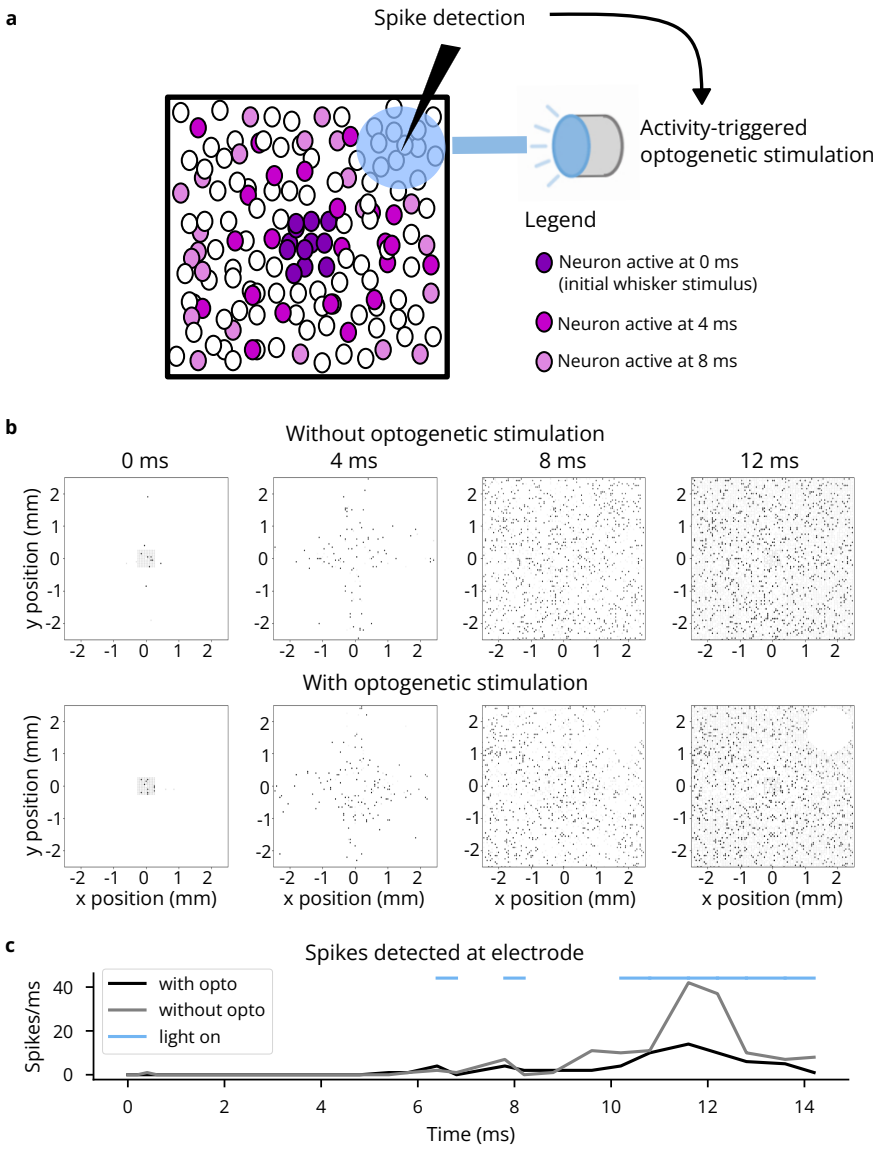

Figure 6: Cleo can simulate reactive control in response to whisker stimulation. (A) Illustration of simulated experimental setup of a 5 mm × 5 mm cortical area and optrode. The center 1 mm^2 patch of neurons is strongly stimulated, initiating a traveling wave of activity radiating outward. When sufficient spiking is detected at the electrode, an optical stimulus activating an inhibitory opsin is triggered. (B) Spatial spiking rasters. Each dot represents a spiking neuron; dark/light dots indicate high/low firing rates. Neural activity is clearly disrupted by the optogenetic stimulus in the neighborhood of the optrode. (C) Spiking activity is disrupted by the optogenetic stimulus. Photostimulation (represented with a blue line) is activated when 3 or more spikes are detected over a 1-ms sampling period.

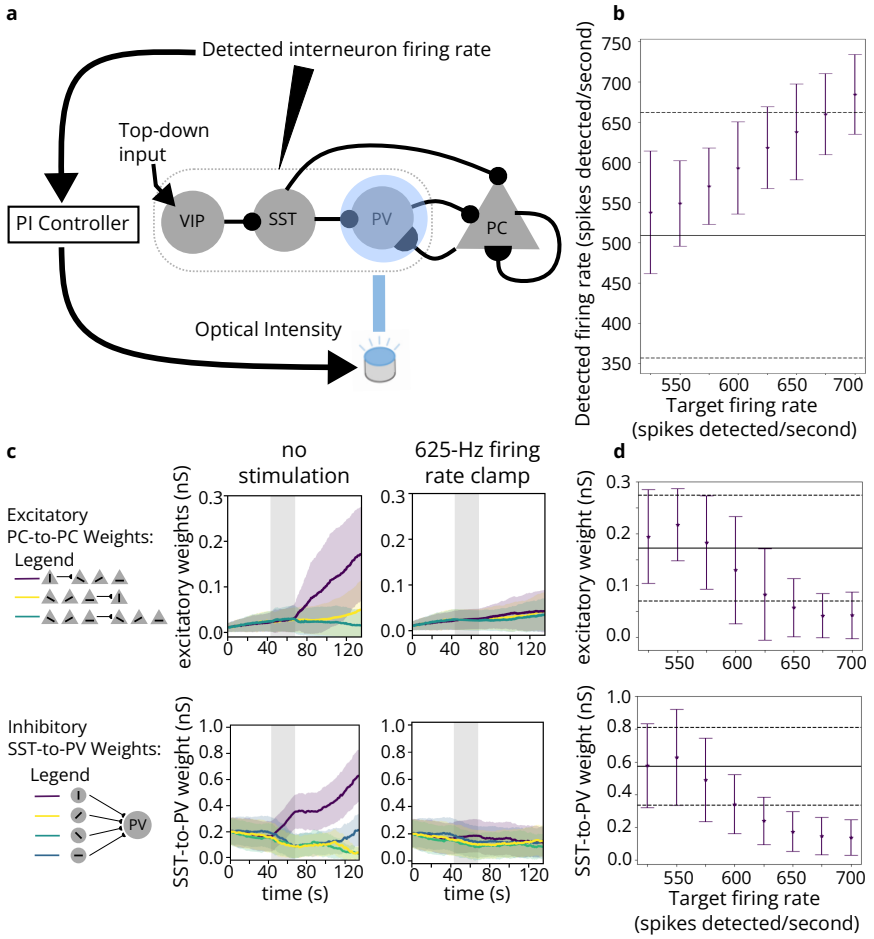

Figure 7: Cleo simulates optogenetic feedback control of interneuron firing rate, disrupting top-down visual plasticity. (A) A model including simulated VIP, SST, PV, and PC neurons [83] was perturbed via optogenetic feedback control. The PI controller set light intensity targeting PV interneurons transfected with channelrhodopsin-2. (B) Actual multi-unit, reward period firing rates for various targets. Dots indicate the mean over time; error bars are one s.d. Solid and dotted black lines indicate mean and s.d. for the unperturbed model. (C) The neural weights across time for PC-PC and SST-PV connections. Neurons are grouped by which stimulus they were selective for, where the vertical stimulus was rewarded. Top: PC-to-PC connection weights for neurons selective (S) to the rewarded stimulus to nonselective neurons (NS) are shown in purple, NS-S in yellow, and NS-NS in green. Bottom: SST-PV weights for S-PV connections are shown in purple, and NS-PV are shown in yellow, green, and blue. Mean weights are shown with solid lines and standard deviations are indicated by shaded regions. Top-down reward period is indicated by gray shading. Weights over time without (with) optogenetic control of firing rate are shown on the left (right). (D) The weights at simulation end ($t = 126\,\mathrm{s}$) for PC-PC, S-NS connections (top) and for reward-selective SST-to-PV connections (bottom). Dots indicate the mean over synapses; error bars are one s.d. Solid and dotted black lines indicate mean and s.d. for the unperturbed model.

locations in a 400 µm × 400 µm × 200 µm volume). This model features a variety of neuron subtypes, including pools of vasoactive intestinal peptide-expressing (VIP), somatostatin (SST), parvalbumin (PV), and pyramidal cell (PC) neurons. The network is defined with inhibitory connections from VIP-to-SST, SST-to-PV, SST-to-PC, and PV-to-PC, as well as excitatory connections from PC-to-PV and PC-to-PC (see Figure 7a). A brief period (24.5 seconds) of top-down reward input to VIP is sufficient to cause substantial changes to neural weights in a longer post-reward period. This is because top-down reward causes SST to inhibit PV, which in turn disinhibits the PC, allowing for plasticity in the PC neurons that continues past the end of the reward period. Thus, we concluded that slightly disrupting PV activity should be sufficient to disrupt plasticity in the PC connections.

We used Cleo to model an electrode recording multi-unit inhibitory activity (spiking from SST and PV neurons), simulating the scenario where the cell type of incoming spikes is identified in real time based on their waveform. To establish a baseline, we observed spiking activity without any optogenetic stimulus, noting the mean and standard deviation of detected firing rate during the reward period to determine target firing rates for closed-loop control in subsequent simulations. Based on these results, we then ran 8 simulations, each with a different reference reward period firing rate ranging from 525 (just over the mean) to 700 (over one standard deviation above the mean) spikes per second. We followed the methods in [35], setting the light intensity in real time via a proportional-integral (PI) controller as implemented in the `cleo.ioproc` module. We used integral and proportional gains $K_i = 0.003\,\mathrm{mW/mm^2/spikes}$ and $K_p = 0.005\,\mathrm{mW/mm^2/Hz}$. Firing rate was estimated via an exponential filter with time constant $\tau = 1\,\mathrm{s}$.

This model included a total of 694 neurons simulated over 137 seconds. Without Cleo integration, runtime was approximately 45 minutes in a cluster computing environment. Using Cleo to inject one electrode and one optogenetic stimulus, runtime was approximately one hour in the same environment.

The resulting detected reward period firing rates for each target rate is shown in Figure 7b. PI control modulated firing rates in predictable ways that concorded with the goals of the experiment. Specifically, the reward period firing rate clamp had clear effects on the weights of the neural connections, both for PC-to-PC connections and SST-to-PV connections (shown in Figure 7c). The open-loop alternative to attain a given reference firing rate would be the careful and potentially time-consuming titration of stimulation levels. In this way, Cleo has demonstrated a nominal prototype of an experiment where closed-loop optogenetic control can potentially be used to draw a more compelling causal connection between components of a network. This also demonstrates Cleo's built-in PI control algorithms which provide users with an easy point of entry to feedback control.

3.5. Case study 3: Evoking SWRs in the hippocampus

To demonstrate Cleo's capabilities to simulate optimal feedback control and approximate LFP, we interfaced Cleo with an anatomically informed model of the hippocampus previously reported [85, 86]. When a sustained external current is delivered to the entorhinal cortex of this model to simulate the slow waves of non-REM sleep, the model produces a sharp wave-ripple (SWR)-like pattern of LFPs [87], as approximated by the reference weighted sum method of [75]. Our goal was to evoke a SWR using optogenetics in the absence of this strong square-wave input, illustrating how feedback control can reproduce a signal of interest at arbitrary times.

In the Cleo simulation, we placed simulated electrode contacts at the same locations as in the original model and used them to record LFP using the TKLFP approximation [73]. We then inserted optic fibers along the 15-mm model length and injected Gaussian process noise into the external current driving the model to create trial-to-trial variability. We illustrate three stimulation paradigms: naïve open-loop (consisting of a mirrored, rectified version of the reference signal), model-based open-loop (computed using an LQR optimal controller run on a model fit to previously collected data) and feedback control (using this same LQR controller online to deliver input after a 3-ms delay). The results demonstrate that Cleo can be used to simulate complex experimental scenarios with multiple recording and stimulation interfaces, and a variety of experimental stimulation protocols can be prototyped on the same model with relative ease. In this case, the simulated response to stimulation is stereotypical, creating little meaningful variation between the three scenarios (see Figure 8b). We expect differences to become more pronounced for a less predictable system and when using more sophisticated control schemes.

The required computation time was not prohibitively high for consumer hardware. Using Brian's Cython code generation, the 400-millisecond simulation took roughly 9 minutes for the stimulus trials, as compared to 4 minutes with LFP recording but without optogenetics. Part of the slowdown is due to the fact that a separate opsin model was simulated for every optic fiber. Near-future plans for Cleo's improvement include efficient support for many-to-many light-opsin relationships, which should mitigate this.

4. Discussion

Here we have presented Cleo, a Python package built around Brian 2 designed as a testbed for bridging computational models and experiments for mesoscale neuroscience. Specifically, by simulating closed-loop control, electrophysiology, and optogenetics, Cleo can bridge model and experiment in two ways: prototyping potential experiments, and making a simulation more directly comparable to experimental data. These can be thought of as model informing experiment and experiment informing model, respectively.

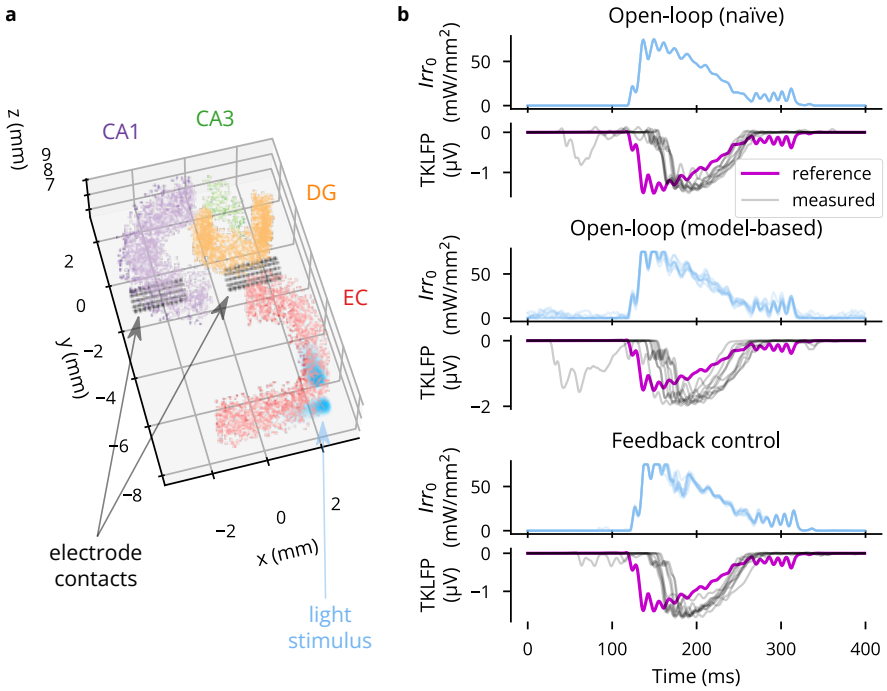

Figure 8: An example application of Cleo to the anatomical hippocampus model by Aussel *et al.* [85, 86]. (A) A 2.5 mm-thick slice of the 15 mm-tall model is shown. The model consists of four regions, entorhinal cortex (EC), dentate gyrus (DG), CA3, and CA1. Electrode contacts are represented as black dots and are in the same location as in the original model. Two light sources are shown in EC. Nine other such pairs (for a total of 20 light sources) not pictured here were spaced regularly parallel to the z axis. (B) Results are shown for ten trials each of three stimulation paradigms: naïve open-loop, where the input is simply a mirrored, rectified version of the reference signal; model-based open-loop, where a controller is run offline on a simulated model; and feedback control, where a controller is run using online measurements. Input Irr_0 is the light intensity at the tip of each optic fiber. The system output TKLFP refers to the Teleńczuk kernel LFP approximation.

Using Cleo to prototype experiments is especially important when considering advanced methods that require considerable time, resources, or risks to implement (especially true in closed-loop experiments). Thus, the researcher can answer beforehand questions such as whether an experiment is feasible [56], which opsin(s) to use, what cells to target, where to record, or what closed-loop control algorithms perform adequately and with tolerable latency. When a sufficiently realistic model for the studied system does not exist, multiple models representing possible variations in connectivity, parameters, or mechanisms could be used to cast light on which experimental configurations work best across hypothetical models. Indeed, the desired experiment in this case could be one that best adjudicates between these hypotheses [88].

Using Cleo to compare the results of a simulation more directly with experimental data allows a more useful bridge between theory and experiment than the typical practice of using artificial inputs and ground-truth model outputs instead of inputs and measurements that more closely mimic an electrophysiology experiment. Subsequent comparison of simulation to experiment would allow the user to evaluate the model in the spirit of NeuronUnit [89], NetworkUnit [90, 91], and other such tools [92].

As mentioned previously, a primary motivation for developing Cleo was to prototype closed-loop optogenetic control (CLOC), which may enable stronger causal hypothesis testing. Neuroscientists have identified many network- and circuit-level variables and phenomena in search of interpretable explanations of brain activity. A natural application of CLOC is to control these features to enable stronger inference of their relationship to downstream variables. Examples of these potential targets for control include the activity of different cell types [93, 94, 95, 96, 97, 98]; the type [99, 100, 101], frequency [102], amplitude [102], spike coherence [103, 104] and interactions [105, 106] of different oscillatory patterns [107]; discrete phenomena such as bursts [79], sharp wave ripples [87], oscillatory bursts [108, 109, 110, 111, 112], traveling waves [113, 114, 115, 37, 81, 116], or sleep spindles [117]; and latent states describing neural dynamics [118, 119, 120, 121, 122, 123, 124, 125], including those most relevant to behavior [20, 126, 127].

While some of these targets lend themselves easily to CLOC, others require continued innovation in interfacing technology. Specifically, stimulation technologies have been much more limited in their degrees of freedom than modern recording technology [9, 10, 11, 12, 13, 14, 15, 16], and thus unlikely to sufficiently control many variables of interest. For this reason, the development of multi-channel micro-LED/optrode devices [128, 129, 130, 131, 132, 133, 134, 135, 56, 136, 137, 8] and holographic optogenetic stimulation [138, 139, 140, 37, 3] are of particular interest. Crucially, Cleo will enable rigorous investigation of both proposed specific technologies as well as general technological capabilities to guide new interface design.

While Cleo was designed to facilitate and accelerate the simulation of complex experiments as much as possible, it has several limitations. First, while Brian and

Cleo have the flexibility to accommodate a wide variety of models, alternative tools and methods—adapted as necessary to simulate the experimental interface—may be better suited for larger spatiotemporal scales [141, 142, 52], higher levels of abstraction [143, 42, 44], and greater biophysical detail [59, 51, 52, 53]. A second limitation is that the flexibility that enables arbitrary closed-loop stimulation can slow down what might otherwise be a fast, purely compiled simulation. A third limitation is that the user must work to interface their model with Cleo, which could be simple (e.g., assigning neuron coordinates) or complicated (e.g., re-implementing the model in Brian either from scratch or using NeuroML [144] import functionality). Finally, users wanting to take advantage of state-of-the-art optogenetic innovations such as improved channel rhodopsins [145, 146, 147, 148, 149, 3, 150], chloride pumps [151, 152, 153] and channels [154, 155], and other innovative opsins [153, 4] will need to provide additional light and opsin model parameters. Thankfully, parameters for many such opsins are available in published literature [156, 157, 158, 159, 160].

Cleo is open-source and can be installed from the Python Package Index under the name "cleosim". The code can is hosted on GitHub at https://github.com/Sensory-Information-Processing-Lab/Cleo. Documentation, including an overview, tutorials, and API reference, can be found at https://cleosim.readthedocs.io. Future development of Cleo is relatively straightforward given Cleo's modular structure. We anticipate future development to meet community needs may include simulation of different levels of abstraction (e.g., forward modeling of extracellular potentials [68, 58] for multi-compartment models or more realistic light propagation profiles [161, 162, 163]), additional/improved recording and stimulation modalities (e.g., calcium and voltage imaging, electrical micro-stimulation, or multi-light-source/multi-wavelength optogenetics), or support for heterogeneous sampling rates to capture scenarios such as when imaging is slower than electrode recording (similar to the approach taken by the real-time processing software Bonsai [164]).

5. Author contributions

KJ designed and developed the software, executed the third example, and wrote most of the manuscript and the documentation. NC identified, simulated, and wrote the results for the first two examples, as well as helped maintain code. CR and AW provided direction and consultation and CR additionally edited the manuscript.

6. Supplemental methods

6.1. Case study 3

The reference signal was generated by delivering a 1-nA, square wave input from 100 to 300 ms to entorhinal cortex using the original model's I_{ext} term, without noise added. Training data was generated by running the system for 13 seconds with alternating on and off periods of length $T \sim \mathcal{N}(200,\ 50)\,\mu\text{s}$. During "on" intervals, $I_{\text{ext}} \sim |\mathcal{N}\left(0, Irr_{\text{max}}/3\right)|$. Irr_{max} was $75\,\text{mw/mm}^2$, which is described as an upper safety limit for 473-nm light delivery to the brain [165]. Gaussian process noise was generated with mean $\mu = 0.167\,\text{nA}$ and using the exponentiated quadratic kernel

$$k\left(t_1, t_2\right) = \sigma^2 e^{\frac{-(t_2 - t_1)^2}{2l^2}}$$

with $\sigma = 0.083\,\text{nA}$, $l = 30\,\text{ms}$ and was added to input current I_{ext} in each control scenario. The training data was fit using the ldsCtrlEst library's SSID and EM fitting methods with a hidden dimensionality $n_x = 4$ [76]. ldsCtrlEst's Gaussian controller was used with an LQR gain with $Q = C^T C$, $R = 0.001$. The parameters of light delivery were altered from the 473-nm optic fiber defaults to allow for greater propagation—K and S were both divided by 10.

7. Supplemental figures

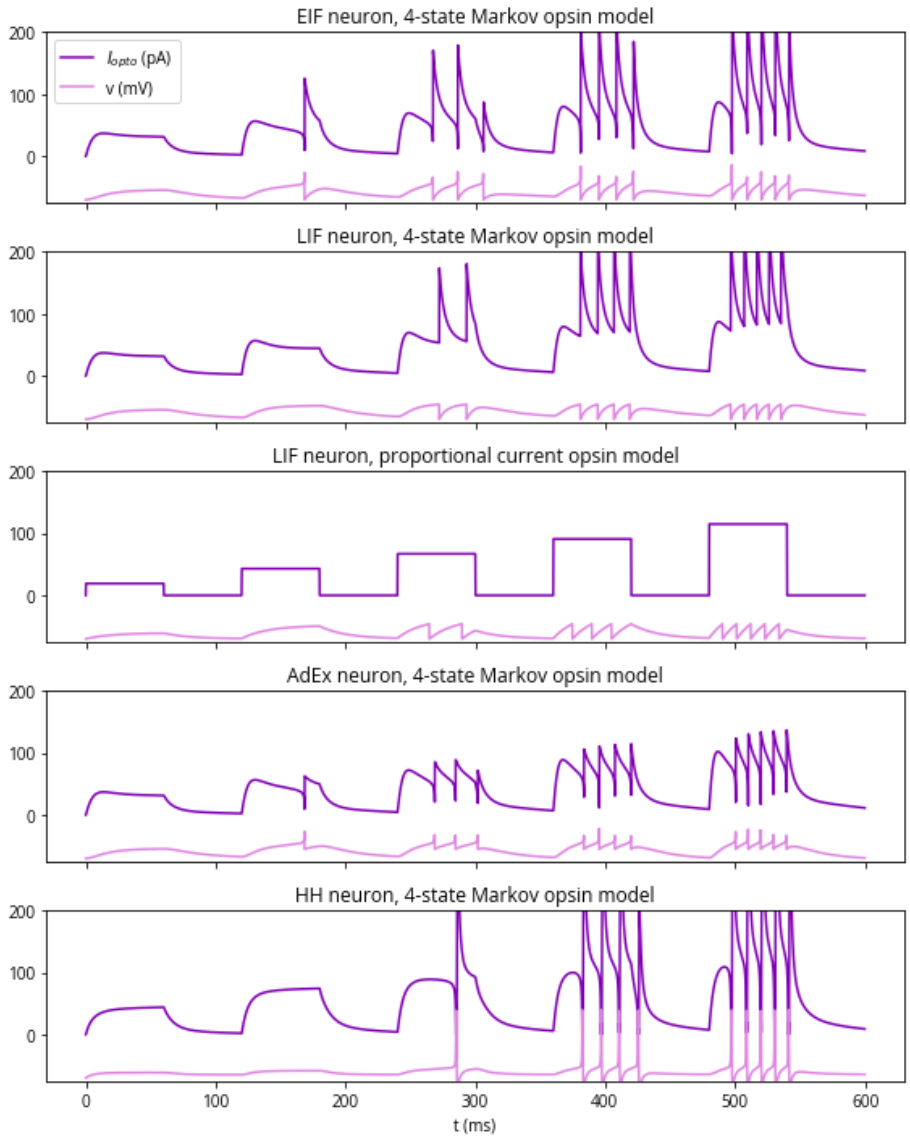

Figure 9: Comparison of different opsin and neuron model combinations, illustrating that similar light-firing rate relationships can be achieved multiple ways.

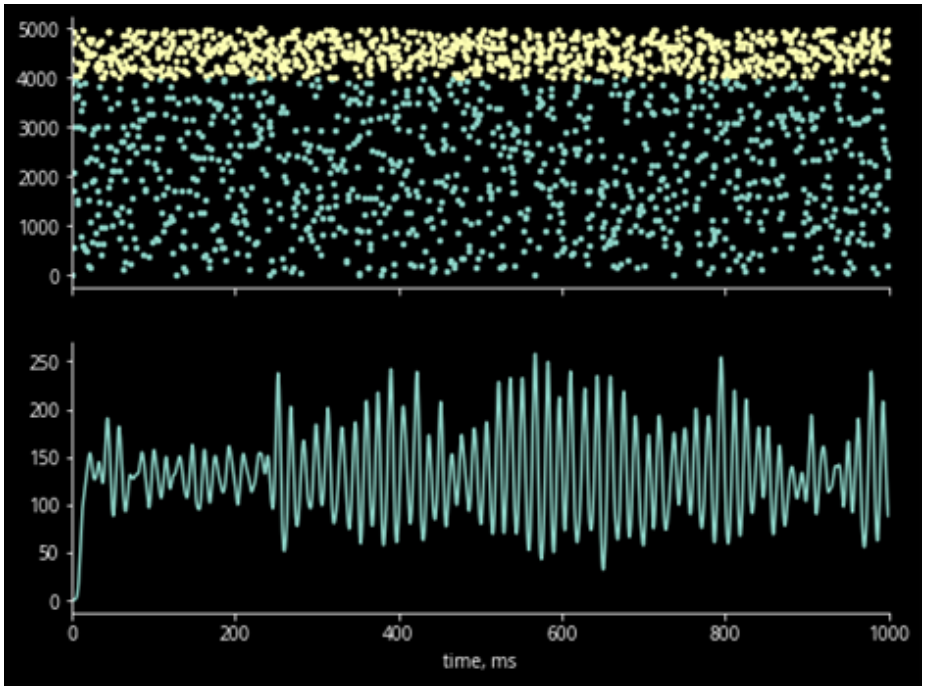

Figure 10: Replication of the Telenczuk kernel LFP demo, showing that the LFP signal produced by the given spikes is practically identical.

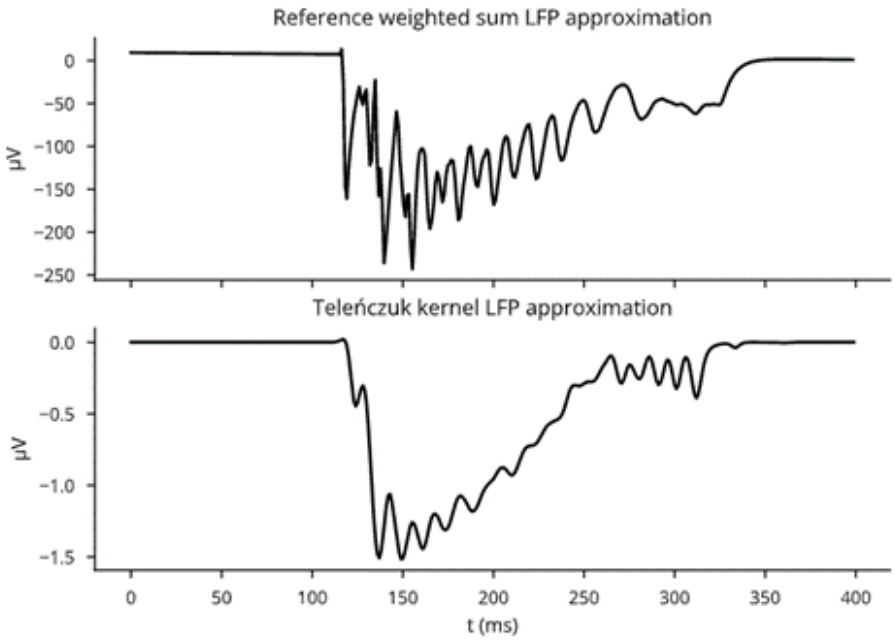

Figure 11: Comparison of LFP approximation methods for the hippocampus SWR case study. The reference weighted sum method is described in [75] and is implemented in [85]. This implementation had a sign error, which is corrected here. The Gaussian kernel approximation method is as described in [73] and computed by Cleo (see Section 2.4.2). We see here that TKLFP captures less high-frequency content than the weighted sum method, which is as reported by Teleńczuk et al.

References

[1] L. Fenno, O. Yizhar, K. Deisseroth, The development and application of optogenetics, Annual Review of Neuroscience 34 (2011) 389–412. doi:10.1146/annurev-neuro-061010-113817. 1

[2] J. S. Wiegert, M. Mahn, M. Prigge, Y. Printz, O. Yizhar, Silencing Neurons: Tools, Applications, and Experimental Constraints, Neuron 95 (3) (2017) 504–529. doi:10.1016/j.neuron.2017.06.050. 1

[3] S. Sridharan, M. A. Gajowa, M. B. Ogando, U. K. Jagadisan, L. Abdeladim, M. Sadahiro, H. A. Bounds, W. D. Hendricks, T. S. Turney, I. Tayler, K. Gopakumar, I. A. Oldenburg, S. G. Brohawn, H. Adesnik, High-performance microbial opsins for spatially and temporally precise perturbations of large neuronal networks, Neuron 110 (7) (2022) 1139–1155.e6. doi:10.1016/j.neuron.2022.01.008. 1, 20, 21

[4] J. Vierock, S. Rodriguez-Rozada, A. Dieter, F. Pieper, R. Sims, F. Tenedini, A. C. Bergs, I. Bendifallah, F. Zhou, N. Zeitzschel, J. Ahlbeck, S. Augustin, K. Sauter, E. Papagiakoumou, A. Gottschalk, P. Soba, V. Emiliani, A. K. Engel, P. Hegemann, J. S. Wiegert, BiPOLES is an optogenetic tool developed for bidirectional dual-color control of neurons, Nature Communications 12 (1) (2021) 1–20. doi:10.1038/s41467-021-24759-5. 1, 21

[5] H. Adesnik, L. Abdeladim, Probing neural codes with two-photon holographic optogenetics, Nature Neuroscience 24 (10) (2021) 1356–1366. doi:10.1038/s41593-021-00902-9. 1

[6] G. Faini, C. Molinier, C. Telliez, C. Tourain, B. C. Forget, E. Ronzitti, V. Emiliani, Ultrafast Light Targeting for High-Throughput Precise Control of Neuronal Networks, bioRxiv (2021) 2021.06.14.448315doi:10.1101/2021.06.14.448315. 1

[7] B. L. Roth, DREADDs for Neuroscientists, Neuron 89 (4) (2016) 683–694. doi:10.1016/j.neuron.2016.01.040. 1

[8] D. Eriksson, A. Schneider, A. Thirumalai, M. Alyahyay, B. de la Crompe, K. Sharma, P. Ruther, I. Diester, Multichannel optogenetics combined with laminar recordings for ultra-controlled neuronal interrogation, Nature Communications 13 (1) (2022) 985. doi:10.1038/s41467-022-28629-6. 1, 20

[9] N. A. Steinmetz, C. Aydin, A. Lebedeva, M. Okun, M. Pachitariu, M. Bauza, M. Beau, J. Bhagat, C. Böhm, M. Broux, S. Chen, J. Colonell, R. J. Gardner, B. Karsh, F. Kloosterman, D. Kostadinov, C. Mora-Lopez, J. O'Callaghan, J. Park, J. Putzeys, B. Sauerbrei, R. J. van Daal, A. Z. Vollan, S. Wang, M. Welkenhuysen, Z. Ye, J. T. Dudman, B. Dutta, A. W. Hantman, K. D. Harris, A. K. Lee, E. I. Moser,

J. O'Keefe, A. Renart, K. Svoboda, M. Häusser, S. Haesler, M. Carandini, T. D. Harris, Neuropixels 2.0: A miniaturized high-density probe for stable, long-term brain recordings, Science 372 (6539) (Apr. 2021). doi:10.1126/science.abf4588. 1, 20

[10] J. H. Siegle, A. C. López, Y. A. Patel, K. Abramov, S. Ohayon, J. Voigts, Open Ephys: An open-source, plugin-based platform for multichannel electrophysiology, Journal of Neural Engineering 14 (4) (2017) 045003. doi:10.1088/1741-2552/aa5eea. 1, 20

[11] P. Gutruf, J. A. Rogers, Implantable, wireless device platforms for neuroscience research, Current Opinion in Neurobiology 50 (2018) 42–49. doi:10.1016/j.conb.2017.12.007. 1, 20

[12] W. Göbel, F. Helmchen, In Vivo Calcium Imaging of Neural Network Function, Physiology 22 (6) (2007) 358–365. doi:10.1152/physiol.00032.2007. 1, 20

[13] T. Knöpfel, C. Song, Optical voltage imaging in neurons: Moving from technology development to practical tool, Nature Reviews Neuroscience 20 (12) (2019) 719–727. doi:10.1038/s41583-019-0231-4. 1, 20

[14] K. Svoboda, R. Yasuda, Principles of Two-Photon Excitation Microscopy and Its Applications to Neuroscience, Neuron 50 (6) (2006) 823–839. doi:10.1016/j.neuron.2006.05.019. 1, 20

[15] A. Kazemipour, O. Novak, D. Flickinger, J. S. Marvin, A. S. Abdelfattah, J. King, P. M. Borden, J. J. Kim, S. H. Al-Abdullatif, P. E. Deal, E. W. Miller, E. R. Schreiter, S. Druckmann, K. Svoboda, L. L. Looger, K. Podgorski, Kilohertz frame-rate two-photon tomography, Nature Methods 16 (8) (2019) 778–786. doi:10.1038/s41592-019-0493-9. 2, 20

[16] J. Wu, Y. Liang, S. Chen, C. L. Hsu, M. Chavarha, S. W. Evans, D. Shi, M. Z. Lin, K. K. Tsia, N. Ji, Kilohertz two-photon fluorescence microscopy imaging of neural activity in vivo, Nature Methods 17 (3) (2020) 287–290. doi:10.1038/s41592-020-0762-7. 2, 20

[17] L. van der Maaten, G. Hinton, Visualizing Data using t-SNE, Journal of Machine Learning Research 9 (86) (2008) 2579–2605. 2

[18] G. J. Berman, D. M. Choi, W. Bialek, J. W. Shaevitz, Mapping the stereotyped behaviour of freely moving fruit flies, Journal of The Royal Society Interface 11 (99) (2014) 20140672. doi:10.1098/rsif.2014.0672. 2

[19] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, M. Bethge, DeepLabCut: Markerless pose estimation of user-defined body parts with deep learning, Nature Neuroscience 21 (9) (2018) 1281–1289. doi:10.1038/s41593-018-0209-y. 2

[20] O. G. Sani, H. Abbaspourazad, Y. T. Wong, B. Pesaran, M. M. Shanechi, Modeling behaviorally relevant neural dynamics enabled by preferential subspace identification, Nature Neuroscience 24 (1) (2021) 140–149. doi:10.1038/s41593-020-00733-0. 2, 20

[21] O. Sporns, Graph theory methods: Applications in brain networks, Dialogues in Clinical Neuroscience 20 (2) (2018) 111–121. doi:10.31887/DCNS.2018.20.2/osporns. 2

[22] S. Schneider, J. H. Lee, M. W. Mathis, Learnable latent embeddings for joint behavioral and neural analysis (Apr. 2022). arXiv:2204.00673, doi:10.48550/arXiv.2204.00673. 2

[23] A. L. Juavinett, G. Bekheet, A. K. Churchland, Chronically implanted Neuropixels probes enable high-yield recordings in freely moving mice, eLife 8 (2019) e47188. doi:10.7554/eLife.47188. 2

[24] L. K. Scheffer, C. S. Xu, M. Januszewski, Z. Lu, S.-y. Takemura, K. J. Hayworth, G. B. Huang, K. Shinomiya, J. Maitlin-Shepard, S. Berg, J. Clements, P. M. Hubbard, W. T. Katz, L. Umayam, T. Zhao, D. Ackerman, T. Blakely, J. Bogovic, T. Dolafi, D. Kainmueller, T. Kawase, K. A. Khairy, L. Leavitt, P. H. Li, L. Lindsey, N. Neubarth, D. J. Olbris, H. Otsuna, E. T. Trautman, M. Ito, A. S. Bates, J. Goldammer, T. Wolff, R. Svirskas, P. Schlegel, E. Neace, C. J. Knecht, C. X. Alvarado, D. A. Bailey, S. Ballinger, J. A. Borycz, B. S. Canino, N. Cheatham, M. Cook, M. Dreher, O. Duclos, B. Eubanks, K. Fairbanks, S. Finley, N. Forknall, A. Francis, G. P. Hopkins, E. M. Joyce, S. Kim, N. A. Kirk, J. Kovalyak, S. A. Lauchie, A. Lohff, C. Maldonado, E. A. Manley, S. McLin, C. Mooney, M. Ndama, O. Ogundeyi, N. Okeoma, C. Ordish, N. Padilla, C. M. Patrick, T. Paterson, E. E. Phillips, E. M. Phillips, N. Rampally, C. Ribeiro, M. K. Robertson, J. T. Rymer, S. M. Ryan, M. Sammons, A. K. Scott, A. L. Scott, A. Shinomiya, C. Smith, K. Smith, N. L. Smith, M. A. Sobeski, A. Suleiman, J. Swift, S. Takemura, I. Talebi, D. Tarnogorska, E. Tenshaw, T. Tokhi, J. J. Walsh, T. Yang, J. A. Horne, F. Li, R. Parekh, P. K. Rivlin, V. Jayaraman, M. Costa, G. S. Jefferis, K. Ito, S. Saalfeld, R. George, I. A. Meinertzhagen, G. M. Rubin, H. F. Hess, V. Jain, S. M. Plaza, A connectome and analysis of the adult Drosophila central brain, eLife 9 (2020) e57443. doi:10.7554/eLife.57443. 2

[25] E. R. Oby, M. D. Golub, J. A. Hennig, A. D. Degenhart, E. C. Tyler-Kabara, B. M. Yu, S. M. Chase, A. P. Batista, New neural activity patterns emerge with long-term learning, Proceedings of the National Academy of Sciences 116 (30) (2019) 15210–15215. doi:10.1073/pnas.1820296116. 2

[26] Y. Yang, S. Qiao, O. G. Sani, J. I. Sedillo, B. Ferrentino, B. Pesaran, M. M. Shanechi, Modelling and prediction of the dynamic responses of large-scale

brain networks during direct electrical stimulation, Nature Biomedical Engineering 5 (4) (2021) 324–345. doi:10.1038/s41551-020-00666-w. 2

[27] B. R. Cowley, A. C. Snyder, K. Acar, R. C. Williamson, B. M. Yu, M. A. Smith, Slow Drift of Neural Activity as a Signature of Impulsivity in Macaque Visual and Prefrontal Cortex, Neuron 108 (3) (2020) 551–567.e8. doi:10.1016/j.neuron.2020.07.021. 2

[28] L. Avitan, C. Stringer, Not so spontaneous: Multi-dimensional representations of behaviors and context in sensory areas, Neuron (Jul. 2022). doi:10.1016/j.neuron.2022.06.019. 2

[29] M. Jazayeri, S. Ostojic, Interpreting neural computations by examining intrinsic and embedding dimensionality of neural activity, Current Opinion in Neurobiology 70 (2021) 113–120. arXiv:2107.04084, doi:10.1016/j.conb.2021.08.002. 2

[30] L. Grosenick, J. H. Marshel, K. Deisseroth, Review Closed-Loop and Activity-Guided Optogenetic Control, Neuron 86 (2015) 106–139. doi:10.1016/j.neuron.2015.03.034. 2

[31] A. Kumar, I. Vlachos, A. Aertsen, C. Boucsein, Challenges of understanding brain function by selective modulation of neuronal subpopulations, Trends in Neurosciences 36 (10) (2013) 579–586. doi:10.1016/j.tins.2013.06.005. 2

[32] S. M. Potter, A. El Hady, E. E. Fetz, Closed-loop neuroscience and neuroengineering, Frontiers in Neural Circuits 0 (2014) 115. doi:10.3389/FNCIR.2014.00115. 2

[33] G. Acharya, S. F. Ruf, E. Nozari, Brain Modeling for Control: A Review (Oct. 2022). arXiv:2210.15957, doi:10.48550/arXiv.2210.15957. 2

[34] J. P. Newman, M. F. Fong, D. C. Millard, C. J. Whitmire, G. B. Stanley, S. M. Potter, Optogenetic feedback control of neural activity, eLife (2015). doi:10.7554/eLife.07192. 2

[35] M. F. Bolus, A. A. Willats, C. J. Whitmire, C. J. Rozell, G. B. Stanley, Design strategies for dynamic closed-loop optogenetic neurocontrol in vivo, Journal of Neural Engineering 15 (2) (2018) 026011. doi:10.1088/1741-2552/aaa506. 2, 12, 17

[36] M. F. Bolus, A. A. Willats, C. J. Rozell, G. B. Stanley, State-space optimal feedback control of optogenetically driven neural activity, Journal of neural engineering 18 (3) (2021) 036006. doi:10.1101/2020.06.25.171785. 2, 12

[37] Z. Zhang, L. E. Russell, A. M. Packer, O. M. Gauld, M. Häusser, Closed-loop all-optical interrogation of neural circuits in vivo, Nature Methods 15 (12) (2018) 1037–1040. doi:10.1038/s41592-018-0183-z. 2, 20

[38] A. C. F. Bergs, J. F. Liewald, S. Rodriguez-Rozada, Q. Liu, C. Wirt, A. Bessel, N. Zeitzschel, H. Durmaz, A. Nozownik, J. Vierock, C. I. Bargmann, P. Hegemann, J. S. Wiegert, A. Gottschalk, All-optical closed-loop voltage clamp for precise control of muscles and neurons in live animals (Jun. 2022). doi:10.1101/2022.06.03.494532. 2

[39] E. Krook-Magnuson, C. Armstrong, M. Oijala, I. Soltesz, On-demand optogenetic control of spontaneous seizures in temporal lobe epilepsy, Nature Communications 4 (1) (2013) 1–8. doi:10.1038/ncomms2376. 2, 14

[40] A. Witt, A. Palmigiano, A. Neef, A. El Hady, F. Wolf, D. Battaglia, Controlling the oscillation phase through precisely timed closed-loop optogenetic stimulation: A computational study, Frontiers in Neural Circuits 7 (April) (2013) 1–17. doi:10.3389/fncir.2013.00049. 2

[41] S. Dutta, E. Ackermann, C. Kemere, Analysis of an open source, closed-loop, realtime system for hippocampal sharp-wave ripple disruption, Journal of Neural Engineering 16 (1) (2019) 016009. doi:10.1088/1741-2552/aae90e. 2

[42] D. L. K. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, J. J. DiCarlo, Performance-optimized hierarchical models predict neural responses in higher visual cortex, Proceedings of the National Academy of Sciences 111 (23) (2014) 8619–8624. doi:10.1073/pnas.1403112111. 2, 21

[43] S.-M. Khaligh-Razavi, N. Kriegeskorte, Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation, PLOS Computational Biology 10 (11) (2014) e1003915. doi:10.1371/journal.pcbi.1003915. 2

[44] C. Pandarinath, D. J. O'Shea, J. Collins, R. Jozefowicz, S. D. Stavisky, J. C. Kao, E. M. Trautmann, M. T. Kaufman, S. I. Ryu, L. R. Hochberg, J. M. Henderson, K. V. Shenoy, L. F. Abbott, D. Sussillo, Inferring single-trial neural population dynamics using sequential auto-encoders, Nature Methods 15 (10) (2018) 805–815. doi:10.1038/s41592-018-0109-9. 2, 21

[45] A. S. Andalman, V. M. Burns, M. Lovett-Barron, M. Broxton, B. Poole, S. J. Yang, L. Grosenick, T. N. Lerner, R. Chen, T. Benster, P. Mourrain, M. Levoy, K. Rajan, K. Deisseroth, Neuronal Dynamics Regulating Brain and Behavioral State Transitions, Cell 177 (4) (2019) 970–985.e20. doi:10.1016/j.cell.2019.02.037. 2

[46] N. A. Cruzado, Z. Tiganj, S. L. Brincat, E. K. Miller, M. W. Howard, Conjunctive representation of what and when in monkey hippocampus and lateral prefrontal cortex during an associative memory task, Hippocampus 30 (12) (2020) 1332–1346. doi:10.1002/hipo.23282. 2

[47] N. Kriegeskorte, P. K. Douglas, Cognitive computational neuroscience, Nature Neuroscience 21 (9) (2018) 1148–1160. arXiv:1807.11819, doi:10.1038/s41593-018-0210-5. 2

[48] J. Antolík, A. P. Davison, Integrated workflows for spiking neuronal network simulations, Frontiers in Neuroinformatics 7 (2013) 34. doi:10.3389/fninf.2013.00034. 2, 5

[49] R. J. Tomsett, M. Ainsworth, A. Thiele, M. Sanayei, X. Chen, M. A. Gieselmann, M. A. Whittington, M. O. Cunningham, M. Kaiser, Virtual Electrode Recording Tool for EXtracellular potentials (VERTEX): Comparing multi-electrode recordings from simulated and biological mammalian cortical tissue, Brain Structure and Function 220 (4) (2015) 2333–2353. doi:10.1007/s00429-014-0793-x. 2, 5

[50] C. Thornton, F. Hutchings, M. Kaiser, The virtual electrode recording tool for extracellular potentials (VERTEX) Version 2.0: Modelling in vitro electrical stimulation of brain tissue, Wellcome Open Research 4 (2019). doi:10.12688/wellcomeopenres.15058.1. 2, 5

[51] S. L. Gratiy, Y. N. Billeh, K. Dai, C. Mitelut, D. Feng, N. W. Gouwens, N. Cain, C. Koch, C. A. Anastassiou, A. Arkhipov, BioNet: A Python interface to NEURON for modeling large-scale networks, PLoS ONE 13 (8) (2018) e0201630. doi:10.1371/journal.pone.0201630. 2, 5, 21

[52] K. Dai, S. L. Gratiy, Y. N. Billeh, R. Xu, B. Cai, N. Cain, A. E. Rimehaug, A. J. Stasik, G. T. Einevoll, S. Mihalas, C. Koch, A. Arkhipov, Brain Modeling ToolKit: An open source software suite for multiscale modeling of brain circuits, PLoS Computational Biology 16 (11) (2020) e1008386. doi:10.1371/journal.pcbi.1008386. 2, 5, 21

[53] S. Dura-Bernal, B. A. Suter, P. Gleeson, M. Cantarelli, A. Quintana, F. Rodriguez, D. J. Kedziora, G. L. Chadderdon, C. C. Kerr, S. A. Neymotin, R. A. McDougal, M. Hines, G. M. Shepherd, W. W. Lytton, NetpyNE, a tool for data-driven multiscale modeling of brain circuits, eLife 8 (Apr. 2019). doi:10.7554/eLife.44494. 2, 5, 21

[54] M. Stimberg, R. Brette, D. F. Goodman, Brian 2, an intuitive and efficient neural simulator, eLife 8 (Aug. 2019). doi:10.7554/eLife.47314. 2, 6

[55] A. P. Davison, D. Brüderle, J. Eppler, J. Kremkow, E. Muller, D. Pecevski, L. Perrinet, P. Yger, PyNN: A common interface for neuronal network simulators, Frontiers in Neuroinformatics 2 (JAN) (2009) 11. doi:10.3389/neuro.11.011.2008. 5

[56] J. Antolik, Q. Sabatier, C. Galle, Y. Frégnac, R. Benosman, Assessment of optogenetically-driven strategies for prosthetic restoration of cortical vision in large-scale neural simulation of V1, Scientific Reports 11 (1) (2021) 1–18. doi:10.1038/s41598-021-88960-8. 5, 20

[57] H. Lindén, E. Hagen, S. Leski, E. Norheim, K. Pettersen, G. Einevoll, LFPy: A tool for biophysical simulation of extracellular potentials generated by detailed model neurons, Frontiers in Neuroinformatics 7 (2014). 5

[58] E. Hagen, S. Næss, T. V. Ness, G. T. Einevoll, Multimodal modeling of neural network activity: Computing LFP, ECoG, EEG, and MEG signals with LFPy 2.0, Frontiers in Neuroinformatics 12 (2018) 92. doi:10.3389/fninf.2018.00092. 5, 21

[59] M. L. Hines, N. T. Carnevale, The NEURON Simulation Environment, Neural Computation 9 (6) (1997) 1179–1209. doi:10.1162/neco.1997.9.6.1179. 5, 21

[60] M.-O. Gewaltig, M. Diesmann, NEST (NEural Simulation Tool), Scholarpedia 2 (4) (2007) 1430. doi:10.4249/scholarpedia.1430. 6

[61] A. Bogdanchikov, M. Zhaparov, R. Suliyev, Python to learn programming, in: Journal of Physics: Conference Series, Vol. 423, IOP Publishing, 2013, p. 012027. doi:10.1088/1742-6596/423/1/012027. 6

[62] A. P. Davison, M. L. Hines, E. Muller, Trends in programming languages for neuroscience simulations, Frontiers in Neuroscience 3 (DEC) (2009) 374–380. doi:10.3389/neuro.01.036.2009. 6

[63] E. Muller, J. A. Bednar, M. Diesmann, M. O. Gewaltig, M. Hines, A. P. Davison, Python in neuroscience, Frontiers in Neuroinformatics 9 (APR) (2015) 11. doi:10.3389/fninf.2015.00011. 6

[64] M. Stimberg, D. F. Goodman, T. Nowotny, Brian2GeNN: Accelerating spiking neural network simulations with graphics hardware, Scientific Reports 10 (1) (Dec. 2020). doi:10.1038/s41598-019-54957-7. 6

[65] T. J. Foutz, R. L. Arlow, C. C. Mcintyre, Theoretical principles underlying optical stimulation of a channelrhodopsin-2 positive pyramidal neuron, J Neurophysiol 107 (2012) 3235–3245. doi:10.1152/jn.00501.2011.-Optogenetics. 6, 12, 13

[66] T. Vo-Dinh, Biomedical Photonics: Handbook, CRC Press, 2003. doi:10.1001/archfaci.3.3.207. 6

[67] B. D. Evans, S. Jarvis, S. R. Schultz, K. Nikolic, PyRhO: A Multiscale Optogenetics Simulation Platform, Frontiers in Neuroinformatics 10 (MAR) (2016) 8. doi:10.3389/fninf.2016.00008. 6, 12, 13

[68] K. H. Pettersen, H. Lindén, A. M. Dale, G. T. Einevoll, Extracellular spikes and CSD, Handbook of neural activity measurement 1 (2012) 92–135. 8, 21

[69] G. Buzsáki, C. A. Anastassiou, C. Koch, The origin of extracellular fields and currents — EEG, ECoG, LFP and spikes, Nature Reviews Neuroscience 2012 13:6 13 (6) (2012) 407–420. doi:10.1038/nrn3241. 8

[70] G. R. Holt, C. Koch, Electrical interactions via the extracellular potential near cell bodies, Journal of Computational Neuroscience 6 (2) (1999) 169–184. doi:10.1023/A:1008832702585. 8

[71] S. R. Nason, A. K. Vaskov, M. S. Willsey, E. J. Welle, H. An, P. P. Vu, A. J. Bullard, C. S. Nu, J. C. Kao, K. V. Shenoy, T. Jang, H.-S. Kim, D. Blaauw, P. G. Patil, C. A. Chestek, A low-power band of neuronal spiking activity dominated by local single units improves the performance of brain–machine interfaces, Nature Biomedical Engineering 2020 (2020) 1–11doi:10.1038/s41551-020-0591-0. 8

[72] V. Viswam, M. E. J. Obien, F. Franke, U. Frey, A. Hierlemann, Optimal electrode size for multi-scale extracellular-potential recording from neuronal assemblies, Frontiers in Neuroscience 13 (APR) (2019) 385. doi:10.3389/fnins.2019.00385. 8

[73] B. Telenczuk, M. Telenczuk, A. Destexhe, A kernel-based method to calculate local field potentials from networks of spiking neurons, Journal of Neuroscience Methods 344 (2020) 108871. doi:10.1016/j.jneumeth.2020.108871. 8, 9, 14, 18, 25

[74] K. Johnsen, Kjohnsen/tklfp: V0.2.0, Zenodo (Jul. 2022). doi:10.5281/zenodo.6787979. 9

[75] A. Mazzoni, H. Lindén, H. Cuntz, A. Lansner, S. Panzeri, G. T. Einevoll, Computing the Local Field Potential (LFP) from Integrate-and-Fire Network Models, PLOS Computational Biology 11 (12) (2015) e1004584. doi:10.1371/JOURNAL.PCBI.1004584. 9, 14, 18, 25

[76] M. F. Bolus, A. A. Willats, K. A. Johnsen, LDS Control & Estimation (Aug. 2022). 12, 22

[77] A. A. Willats, M. F. Bolus, K. A. Johnsen, N. A. Cruzado, G. B. Stanley, C. J. Rozell, CLOCTools, https://cloctools.github.io/ (2022). 12

[78] K. A. Johnsen, CLOCTools: A library of tools for closed-loop neuroscience (Nov. 2022). 12

[79] W. Gerstner, W. M. Kistler, R. Naud, L. Paninski, Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition, Cambridge University Press, 2014. doi:10.1017/CBO9781107447615. 12, 13, 20

[80] N. Fourcaud-Trocmé, D. Hansel, C. Van Vreeswijk, N. Brunel, How Spike Generation Mechanisms Determine the Neuronal Response to Fluctuating Inputs, Journal of Neuroscience 23 (37) (2003) 11628–11640. doi:10.1523/jneurosci.23-37-11628.2003. 12

[81] S. Moldakarimov, M. Bazhenov, D. E. Feldman, T. J. Sejnowski, Structured networks support sparse traveling waves in rodent somatosensory cortex, Proceedings of the National Academy of Sciences of the United States of

America 115 (20) (2018) 5277–5282. doi:10.1073/pnas.1710202115. 14, 20

[82] V. Ego-Stengel, M. A. Wilson, Disruption of ripple-associated hippocampal activity during rest impairs spatial learning in the rat, Hippocampus 20 (1) (2010) 1–10. doi:10.1002/hipo.20707. 14

[83] K. A. Wilmes, C. Clopath, Inhibitory microcircuits for top-down plasticity of sensory representations, Nature Communications 10 (1) (2019) 5055. doi:10.1038/s41467-019-12972-2. 14, 16

[84] R. A. McDougal, T. M. Morse, T. Carnevale, L. Marenco, R. Wang, M. Migliore, P. L. Miller, G. M. Shepherd, M. L. Hines, Twenty years of ModelDB and beyond: Building essential modeling tools for the future of neuroscience, Journal of Computational Neuroscience 42 (1) (2017) 1–10. doi:10.1007/s10827-016-0623-7. 14

[85] A. Aussel, L. Buhry, L. Tyvaert, R. Ranta, A detailed anatomical and mathematical model of the hippocampal formation for the generation of sharp-wave ripples and theta-nested gamma oscillations, Journal of Computational Neuroscience 45 (3) (2018) 207–221. doi:10.1007/s10827-018-0704-x. 18, 19, 25

[86] A. Aussel, R. Ranta, O. Aron, S. Colnat-Coulbois, L. Maillard, L. Buhry, Cell to network computational model of the epileptic human hippocampus suggests specific roles of network and channel dysfunctions in the ictal and interictal oscillations, Journal of Computational Neuroscience (Aug. 2022). doi:10.1007/s10827-022-00829-5. 18, 19

[87] G. Buzsáki, Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning, Hippocampus 25 (10) (2015) 1073–1188. doi:10.1002/hipo.22488. 18, 20

[88] A. A. Willats, Closed-loop identifiability in neural circuits (Nov. 2022). 20

[89] R. C. Gerkin, J. Birgiolas, R. J. Jarvis, C. Omar, S. M. Crook, NeuronUnit: A package for data-driven validation of neuron models using SciUnit, bioRxiv (2019) 665331doi:10.1101/665331. 20

[90] R. Gutzen, M. von Papen, G. Trensch, P. Quaglio, S. Grün, M. Denker, Reproducible neural network simulations: Statistical methods for model validation on the level of network activity data, Frontiers in Neuroinformatics 12 (2018) 90. doi:10.3389/fninf.2018.00090. 20

[91] G. Trensch, R. Gutzen, I. Blundell, M. Denker, A. Morrison, Rigorous Neural Network Simulations: A Model Substantiation Methodology for Increasing the Correctness of Simulation Results in the Absence of Experimental Validation Data, Frontiers in Neuroinformatics 12 (2018). 20

[92] S. Appukuttan, L. Sharma, P. Garcia-Rodriguez, A. Davison, A Software Framework for Validating Neuroscience Models (2022). 20

[93] E. A. Mukamel, J. Ngai, Perspectives on defining cell types in the brain, Current Opinion in Neurobiology 56 (2019) 61–68. doi:10.1016/j.conb.2018.11.007. 20

[94] P. Němec, P. Osten, The evolution of brain structure captured in stereotyped cell count and cell type distributions, Current Opinion in Neurobiology 60 (2020) 176–183. doi:10.1016/j.conb.2019.12.005. 20

[95] R. I. Martinez-Garcia, B. Voelcker, J. B. Zaltsman, S. L. Patrick, T. R. Stevens, B. W. Connors, S. J. Cruikshank, Two dynamically distinct circuits drive inhibition in the sensory thalamus, Nature 583 (7818) (2020) 813–818. doi:10.1038/s41586-020-2512-5. 20

[96] C. Lee, A. Lavoie, J. Liu, S. X. Chen, B.-h. Liu, Light Up the Brain: The Application of Optogenetics in Cell-Type Specific Dissection of Mouse Brain Circuits, Frontiers in Neural Circuits 14 (2020). 20

[97] A. Joglekar, A. Prjibelski, A. Mahfouz, P. Collier, S. Lin, A. K. Schlusche, J. Marrocco, S. R. Williams, B. Haase, A. Hayes, J. G. Chew, N. I. Weisenfeld, M. Y. Wong, A. N. Stein, S. A. Hardwick, T. Hunt, Q. Wang, C. Dieterich, Z. Bent, O. Fedrigo, S. A. Sloan, D. Risso, E. D. Jarvis, P. Flicek, W. Luo, G. S. Pitt, A. Frankish, A. B. Smit, M. E. Ross, H. U. Tilgner, A spatially resolved brain region- and cell type-specific isoform atlas of the postnatal mouse brain, Nature Communications 12 (1) (2021) 463. doi:10.1038/s41467-020-20343-5. 20

[98] H. Zeng, What is a cell type and how to define it?, Cell 185 (15) (2022) 2739–2755. doi:10.1016/j.cell.2022.06.031. 20

[99] S. R. Cole, B. Voytek, Brain Oscillations and the Importance of Waveform Shape, Trends in Cognitive Sciences 21 (2) (2017) 137–149. doi:10.1016/j.tics.2016.12.008. 20

[100] S. Cole, B. Voytek, Cycle-by-cycle analysis of neural oscillations, Journal of Neurophysiology 122 (2) (2019) 849–861. doi:10.1152/JN.00273.2019. 20

[101] M. S. Fabus, A. J. Quinn, C. E. Warnaby, M. W. Woolrich, Automatic decomposition of electrophysiological data into distinct nonsinusoidal oscillatory modes, Journal of Neurophysiology 126 (5) (2021) 1670–1684. doi:10.1152/jn.00315.2021. 20

[102] A. B. Saleem, A. D. Lien, M. Krumin, B. Haider, M. R. Rosón, A. Ayaz, K. Reinhold, L. Busse, M. Carandini, K. D. Harris, M. Carandini, Subcortical Source and Modulation of the Narrowband Gamma Oscillation in Mouse Visual Cortex, Neuron 93 (2) (2017) 315–322. doi:10.1016/j.neuron.2016.12.028. 20

[103] E. A. Buffalo, P. Fries, R. Landman, T. J. Buschman, R. Desimone, Laminar differences in gamma and alpha coherence in the ventral stream, Proceedings of the National Academy of Sciences of the United States of America 108 (27) (2011) 11262–11267. doi:10.1073/pnas.1011284108. 20

[104] T. J. Buschman, E. L. Denovellis, C. Diogo, D. Bullock, E. K. Miller, Synchronous Oscillatory Neural Ensembles for Rules in the Prefrontal Cortex, Neuron 76 (4) (2012) 838–846. doi:10.1016/j.neuron.2012.09.029. 20

[105] J. Aru, J. Aru, V. Priesemann, M. Wibral, L. Lana, G. Pipa, W. Singer, R. Vicente, Untangling cross-frequency coupling in neuroscience, Current Opinion in Neurobiology 31 (2015) 51–61. doi:10.1016/j.conb.2014.08.002. 20

[106] L. Zhang, J. Lee, C. Rozell, A. C. Singer, Sub-second dynamics of theta-gamma coupling in hippocampal CA1, eLife 8 (Jul. 2019). doi:10.7554/eLife.44320. 20

[107] G. Buzsáki, A. Draguhn, Neuronal oscillations in cortical networks, Science 304 (5679) (2004) 1926–1929. doi:10.1126/science.1099745. 20

[108] T. Akam, D. M. Kullmann, Oscillatory multiplexing of population codes for selective communication in the mammalian brain, Nature Reviews Neuroscience 15 (2) (2014) 111–122. doi:10.1038/nrn3668. 20

[109] I. Tal, S. Neymotin, S. Bickel, P. Lakatos, C. E. Schroeder, Oscillatory Bursting as a Mechanism for Temporal Coupling and Information Coding, Frontiers in Computational Neuroscience 14 (2020). 20

[110] M. Lundqvist, J. Rose, P. Herman, S. L. L. Brincat, T. J. J. Buschman, E. K. K. Miller, Gamma and Beta Bursts Underlie Working Memory, Neuron 90 (1) (2016) 152–164. doi:10.1016/j.neuron.2016.02.028. 20

[111] M. Lundqvist, J. Rose, S. L. Brincat, M. R. Warden, T. J. Buschman, P. Herman, E. K. Miller, Reduced variability of bursting activity during working memory, Scientific Reports 12 (1) (2022) 15050. doi:10.1038/s41598-022-18577-y. 20

[112] G. Karvat, A. Schneider, M. Alyahyay, F. Steenbergen, M. Tangermann, I. Diester, Real-time detection of neural oscillation bursts allows behaviourally relevant neurofeedback, Communications Biology 3 (1) (2020) 1–10. doi:10.1038/s42003-020-0801-z. 20

[113] T. K. Sato, I. Nauhaus, M. Carandini, Traveling Waves in Visual Cortex, Neuron 75 (2) (2012) 218–229. doi:10.1016/j.neuron.2012.06.029. 20

[114] L. Muller, F. Chavane, J. Reynolds, T. J. Sejnowski, Cortical travelling waves: Mechanisms and computational principles, Nature Reviews Neuroscience 19 (5) (2018) 255–268. doi:10.1038/nrn.2018.20. 20

[115] M. E. Rule, C. Vargas-Irwin, J. P. Donoghue, W. Truccolo, Phase reorganization leads to transient β-LFP spatial wave patterns in motor cortex during steady-state movement preparation, Journal of Neurophysiology 119 (6) (2018) 2212–2228. doi:10.1152/jn.00525.2017. 20

[116] Z. W. Davis, L. Muller, J. Martinez-Trujillo, T. Sejnowski, J. H. Reynolds, Spontaneous travelling cortical waves gate perception in behaving primates, Nature 587 (7834) (2020) 432–436. doi:10.1038/s41586-020-2802-y. 20

[117] L. M. J. Fernandez, A. Lüthi, Sleep Spindles: Mechanisms and Functions, Physiological Reviews 100 (2) (2020) 805–868. doi:10.1152/physrev.00042.2018. 20

[118] M. M. Churchland, J. P. Cunningham, M. T. Kaufman, J. D. Foster, P. Nuyujukian, S. I. Ryu, K. V. Shenoy, K. V. Shenoy, Neural population dynamics during reaching, Nature 487 (7405) (2012) 51–56. doi:10.1038/nature11129. 20

[119] K. V. Shenoy, M. Sahani, M. M. Churchland, Cortical Control of Arm Movements: A Dynamical Systems Perspective, Annual Review of Neuroscience 36 (2013) 337–359. doi:10.1146/annurev-neuro-062111-150509. 20

[120] J. P. Cunningham, B. M. Yu, Dimensionality reduction for large-scale neural recordings., Nature neuroscience 17 (11) (2014) 1500–9. doi:10.1038/nn.3776. 20

[121] M. T. Kaufman, M. M. Churchland, S. I. Ryu, K. V. Shenoy, Cortical activity in the null space: Permitting preparation without movement, Nature Neuroscience 17 (3) (2014) 440–448. doi:10.1038/nn.3643. 20

[122] J. A. Gallego, M. G. Perich, L. E. Miller, S. A. Solla, Neural Manifolds for the Control of Movement., Neuron 94 (5) (2017) 978–984. doi:10.1016/j.neuron.2017.05.025. 20

[123] S. Vyas, M. D. Golub, D. Sussillo, K. V. Shenoy, Computation through Neural Population Dynamics, Annual Review of Neuroscience 43 (2020) 249–275. doi:10.1146/annurev-neuro-092619-094115. 20

[124] K. V. Shenoy, J. C. Kao, Measurement, manipulation and modeling of brain-wide neural population dynamics, Nature Communications 12 (1) (2021) 1–5. doi:10.1038/s41467-020-20371-1. 20

[125] D. Peixoto, J. R. Verhein, R. Kiani, J. C. Kao, P. Nuyujukian, C. Chandrasekaran, J. Brown, S. Fong, S. I. Ryu, K. V. Shenoy, W. T. Newsome, Decoding and perturbing decision states in real time, Nature 591 (7851) (2021) 604–609. doi:10.1038/s41586-020-03181-9. 20

[126] O. G. Sani, B. Pesaran, M. M. Shanechi, M. Hsieh, Where is all the nonlinearity: Flexible nonlinear modeling of behaviorally relevant neural dynamics using recurrent neural networks, bioRxiv (2021) 2021.09.03.458628doi:10.1101/2021.09.03.458628. 20

[127] C. Hurwitz, A. Srivastava, K. Xu, J. Jude, M. G. Perich, L. E. Miller, M. H. Hennig, Targeted Neural Dynamical Modeling, in: Advances in Neural Information Processing Systems, Vol. 35, 2021, pp. 29379–29392. arXiv:2110.14853. 20

[128] S. Dufour, Y. D. Koninck, Optrodes for combined optogenetics and electrophysiology in live animals, Neurophotonics 2 (3) (2015) 031205. doi:10.1117/1.NPh.2.3.031205. 20

[129] K. Y. Kwon, H.-M. Lee, M. Ghovanloo, A. Weber, W. Li, Design, fabrication, and packaging of an integrated, wirelessly-powered optrode array for optogenetics application, Frontiers in Systems Neuroscience 9 (2015). 20

[130] M. Welkenhuysen, L. Hoffman, Z. Luo, A. De Proft, C. Van den Haute, V. Baekelandt, Z. Debyser, G. Gielen, R. Puers, D. Braeken, An integrated multi-electrode-optrode array for in vitro optogenetics, Scientific Reports 6 (1) (2016) 20353. doi:10.1038/srep20353. 20

[131] L. Wang, K. Huang, C. Zhong, L. Wang, Y. Lu, Fabrication and modification of implantable optrode arrays for in vivo optogenetic applications, Biophysics Reports 4 (2) (2018) 82–93. doi:10.1007/s41048-018-0052-4. 20

[132] N. McAlinden, Y. Cheng, R. Scharf, E. Xie, E. Gu, C. F. Reiche, R. Sharma, P. Tathireddy, P. Tathireddy, L. Rieth, S. Blair, K. Mathieson, Multisite microLED optrode array for neural interfacing, Neurophotonics 6 (3) (2019) 035010. doi:10.1117/1.NPh.6.3.035010. 20

[133] D. Mao, N. Li, Z. Xiong, Y. Sun, G. Xu, Single-Cell Optogenetic Control of Calcium Signaling with a High-Density Micro-LED Array, iScience 21 (2019) 403–412. doi:10.1016/j.isci.2019.10.024. 20

[134] D. Mao, Z. Xiong, M. Donnelly, G. Xu, Brushing-Assisted Two-Color Quantum-Dot Micro-LED Array Towards Bi-Directional Optogenetics, IEEE Electron Device Letters 42 (10) (2021) 1504–1507. doi:10.1109/LED.2021.3108554. 20

[135] Y. Ohta, M. C. Guinto, T. Tokuda, M. Kawahara, M. Haruta, H. Takehara, H. Tashiro, K. Sasagawa, H. Onoe, R. Yamaguchi, Y. Koshimizu, K. Isa, T. Isa, K. Kobayashi, Y. M. Akay, M. Akay, J. Ohta, Micro-LED Array-Based Photo-Stimulation Devices for Optogenetics in Rat and Macaque Monkey Brains, IEEE Access 9 (2021) 127937–127949. doi:10.1109/ACCESS.2021.3111666. 20

[136] S. Jeon, Y. Lee, D. Ryu, Y. K. Cho, Y. Lee, S. B. Jun, C.-H. Ji, Implantable Optrode Array for Optogenetic Modulation and Electrical Neural Recording, Micromachines 12 (6) (2021) 725. doi:10.3390/mi12060725. 20

[137] C. Kathe, F. Michoud, P. Schönle, A. Rowald, N. Brun, J. Ravier, I. Furfaro, V. Paggi, K. Kim, S. Soloukey, L. Asboth, T. H. Hutson, I. Jelescu, A. Philippides, N. Alwahab, J. Gandar, D. Huber, C. I. De Zeeuw, Q. Barraud, Q. Huang, S. P. Lacour, G. Courtine, Wireless closed-loop optogenetics across the entire dorsoventral spinal cord in mice, Nature Biotechnology 40 (2) (2022) 198–208. doi:10.1038/s41587-021-01019-x. 20

[138] A. M. Packer, L. E. Russell, H. W. P. Dalgleish, M. Häusser, Simultaneous all-optical manipulation and recording of neural circuit activity with cellular resolution in vivo, Nature Methods 12 (2) (2015) 140–146. doi:10.1038/nmeth.3217. 20

[139] E. Ronzitti, R. Conti, V. Zampini, D. Tanese, A. J. Foust, N. Klapoetke, E. S. Boyden, E. Papagiakoumou, V. Emiliani, Submillisecond Optogenetic Control of Neuronal Firing with Two-Photon Holographic Photoactivation of Chronos, Journal of Neuroscience 37 (44) (2017) 10679–10689. doi:10.1523/JNEUROSCI.1246-17.2017. 20

[140] I.-W. Chen, E. Papagiakoumou, V. Emiliani, Towards circuit optogenetics, Current Opinion in Neurobiology 50 (2018) 179–189. doi:10.1016/j.conb.2018.03.008. 20

[141] P. Sanzleon, S. A. Knock, M. M. Woodman, L. Domide, J. Mersmann, A. R. Mcintosh, V. Jirsa, The virtual brain: A simulator of primate brain network dynamics, Frontiers in Neuroinformatics 7 (MAY) (2013) 10. doi:10.3389/fninf.2013.00010. 21

[142] S. A. Neymotin, D. S. Daniels, B. Caldwell, R. A. McDougal, N. T. Carnevale, M. Jas, C. I. Moore, M. L. Hines, M. Hämäläinen, S. R. Jones, Human neocortical neurosolver (HNN), a new software tool for interpreting the cellular and network origin of human MEG/EEG data, eLife 9 (Jan. 2020). doi:10.7554/eLife.51214. 21

[143] J. H. Macke, L. Buesing, J. P. Cunningham, B. M. Yu, K. V. Shenoy, M. Sahani, Empirical models of spiking in neural populations, in: Advances in Neural Information Processing Systems, Vol. 24, Curran Associates, Inc., 2011. 21

[144] R. C. Cannon, P. Gleeson, S. Crook, G. Ganapathy, B. Marin, E. Piasini, R. A. Silver, LEMS: A language for expressing complex biological models in concise and hierarchical form and its use in underpinning NeuroML 2, Frontiers in Neuroinformatics 8 (SEP) (2014) 79. doi:10.3389/fninf.2014.00079. 21

[145] L. A. Gunaydin, O. Yizhar, A. Berndt, V. S. Sohal, K. Deisseroth, P. Hegemann, Ultrafast optogenetic control, Nature Neuroscience 13 (3) (2010) 387–392. doi:10.1038/nn.2495. 21

[146] J. Y. Lin, P. M. Knutsen, A. Muller, D. Kleinfeld, R. Y. Tsien, ReaChR: A red-shifted variant of channelrhodopsin enables deep transcranial optogenetic excitation, Nature Neuroscience 16 (10) (2013) 1499–1508. doi:10.1038/nn.3502. 21

[147] N. C. Klapoetke, Y. Murata, S. S. Kim, S. R. Pulver, A. Birdsey-Benson, Y. K. Cho, T. K. Morimoto, A. S. Chuong, E. J. Carpenter, Z. Tian, J. Wang, Y. Xie, Z. Yan, Y. Zhang, B. Y. Chow, B. Surek, M. Melkonian, V. Jayaraman, M. Constantine-Paton, G. K. S. Wong, E. S. Boyden, Independent optical excitation of distinct neural populations, Nature Methods 11 (3) (2014) 338–346. doi:10.1038/nmeth.2836. 21

[148] D. R. Hochbaum, Y. Zhao, S. L. Farhi, N. Klapoetke, C. A. Werley, V. Kapoor, P. Zou, J. M. Kralj, D. MacLaurin, N. Smedemark-Margulies, J. L. Saulnier, G. L. Boulting, C. Straub, Y. K. Cho, M. Melkonian, G. K. S. Wong, D. J. Harrison, V. N. Murthy, B. L. Sabatini, E. S. Boyden, R. E. Campbell, A. E. Cohen, All-optical electrophysiology in mammalian neurons using engineered microbial rhodopsins, Nature Methods 11 (8) (2014) 825–833. doi:10.1038/NMETH.3000. 21

[149] T. Mager, D. L. D. L. Morena, V. Senn, J. Schlotte, A. Derrico, K. Feldbauer, C. Wrobel, S. Jung, K. Bodensiek, V. Rankovic, L. Browne, A. Huet, J. Jüttner, P. G. Wood, J. J. Letzkus, T. Moser, E. Bamberg, High frequency neural spiking and auditory signaling by ultrafast red-shifted optogenetics, Nature Communications 9 (1) (2018) 1–14. doi:10.1038/s41467-018-04146-3. 21

[150] K. E. Kishi, Y. S. Kim, M. Fukuda, M. Inoue, T. Kusakizako, P. Y. Wang, C. Ramakrishnan, E. F. Byrne, E. Thadhani, J. M. Paggi, T. E. Matsui, K. Yamashita, T. Nagata, M. Konno, S. Quirin, M. Lo, T. Benster, T. Uemura, K. Liu, M. Shibata, N. Nomura, S. Iwata, O. Nureki, R. O. Dror, K. Inoue, K. Deisseroth, H. E. Kato, Structural basis for channel conduction in the pump-like channelrhodopsin ChRmine, Cell 185 (4) (2022) 672–689.e23. doi:10.1016/j.cell.2022.01.007. 21

[151] V. Gradinaru, F. Zhang, C. Ramakrishnan, J. Mattis, R. Prakash, I. Diester, I. Goshen, K. R. Thompson, K. Deisseroth, Molecular and Cellular Approaches for Diversifying and Extending Optogenetics, Cell 141 (1) (2010) 154–165. doi:10.1016/j.cell.2010.02.037. 21

[152] A. S. Chuong, M. L. Miri, V. Busskamp, G. A. Matthews, L. C. Acker, A. T. Sørensen, A. Young, N. C. Klapoetke, M. A. Henninger, S. B. Kodandaramaiah, M. Ogawa, S. B. Ramanlal, R. C. Bandler, B. D. Allen, C. R. Forest, B. Y. Chow, X. Han, Y. Lin, K. M. Tye, B. Roska, J. A.

Cardin, E. S. Boyden, Noninvasive optical inhibition with a red-shifted microbial rhodopsin, Nature Neuroscience 17 (8) (2014) 1123–1129. doi:10.1038/nn.3752. 21

[153] A. Berndt, S. Y. Lee, J. Wietek, C. Ramakrishnan, E. E. Steinberg, A. J. Rashid, H. Kim, S. Park, A. Santoro, P. W. Frankland, S. M. Iyer, S. Pak, S. Ährlund-Richter, S. L. Delp, R. C. Malenka, S. A. Josselyn, M. Carlén, P. Hegemann, K. Deisseroth, Structural foundations of optogenetics: Determinants of channelrhodopsin ion selectivity, Proceedings of the National Academy of Sciences of the United States of America 113 (4) (2016) 822–829. doi:10.1073/pnas.1523341113. 21

[154] E. G. Govorunova, O. A. Sineshchekov, R. Janz, X. Liu, J. L. Spudich, Natural light-gated anion channels: A family of microbial rhodopsins for advanced optogenetics, Science 349 (6248) (2015) 647–650. doi:10.1126/science.aaa7484. 21

[155] E. G. Govorunova, O. A. Sineshchekov, E. M. Rodarte, R. Janz, O. Morelle, M. Melkonian, G. K. Wong, J. L. Spudich, The Expanding Family of Natural Anion Channelrhodopsins Reveals Large Variations in Kinetics, Conductance, and Spectral Sensitivity, Scientific Reports 7 (1) (2017) 1–10. doi:10.1038/srep43358. 21

[156] S. Saran, N. Gupta, S. Roy, Theoretical analysis of low-power fast optogenetic control of firing of Chronos-expressing neurons, Neurophotonics 5 (02) (2018) 1. doi:10.1117/1.nph.5.2.025009. 21

[157] H. Bansal, N. Gupta, S. Roy, Theoretical Analysis of Low-power Bidirectional Optogenetic Control of High-frequency Neural Codes with Single Spike Resolution, Neuroscience 449 (2020) 165–188. doi:10.1016/j.neuroscience.2020.09.022. 21

[158] H. Bansal, N. Gupta, S. Roy, Comparison of low-power, high-frequency and temporally precise optogenetic inhibition of spiking in NpHR, eNpHR3.0 and Jaws-expressing neurons, Biomedical Physics and Engineering Express 6 (4) (2020) 045011. doi:10.1088/2057-1976/ab90a1. 21

[159] N. Gupta, H. Bansal, S. Roy, Theoretical optimization of high-frequency optogenetic spiking of red-shifted very fast-Chrimson-expressing neurons, Neurophotonics 6 (02) (2019) 1. doi:10.1117/1.nph.6.2.025002. 21

[160] H. Bansal, N. Gupta, S. Roy, Theoretical analysis of optogenetic spiking with ChRmine, bReaChES and CsChrimson-expressing neurons for retinal prostheses, Journal of Neural Engineering 18 (4) (2021) 0460b8. doi:10.1088/1741-2552/ac1175. 21

[161] Y. Liu, S. L. Jacques, M. Azimipour, J. D. Rogers, R. Pashaie, K. W. Eliceiri, OptogenSIM: A 3D Monte Carlo simulation platform for light delivery design in optogenetics, Biomedical Optics Express 6 (12) (2015) 4859. doi:10.1364/boe.6.004859. 21

[162] B. Gysbrechts, L. Wang, N. N. D. Trong, H. Cabral, Z. Navratilova, F. Battaglia, W. Saeys, C. Bartic, Light distribution and thermal effects in the rat brain under optogenetic stimulation, Journal of Biophotonics 9 (6) (2016) 576–585. doi:10.1002/jbio.201500106. 21

[163] J. W. Luo, K. Nikolic, P. Degenaar, Modelling Optogenetic Subthreshold Effects, in: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 6136–6140. doi:10.1109/EMBC.2019.8856664. 21

[164] G. Lopes, N. Bonacchi, J. Frazão, J. P. Neto, B. V. Atallah, S. Soares, L. Moreira, S. Matias, P. M. Itskov, P. A. Correia, R. E. Medina, L. Calcaterra, E. Dreosti, J. J. Paton, A. R. Kampff, Bonsai: An event-based framework for processing and controlling data streams, Frontiers in Neuroinformatics 9 (2015). 21

[165] J. A. Cardin, M. Carlén, K. Meletis, U. Knoblich, F. Zhang, K. Deisseroth, L. H. Tsai, C. I. Moore, Targeted optogenetic stimulation and recording of neurons in vivo using cell-type-specific expression of Channelrhodopsin-2, Nature Protocols 5 (2) (2010) 247–254. doi:10.1038/nprot.2009.228. 22