Rapid, interpretable, data-driven models of neural dynamics using Recurrent Mechanistic Models

Thiago B. Burghi[1], Maria Ivanova[2], Ekaterina Morozova[2], Huaxinyu Wang[1], Eve Marder[2], and Timothy O'Leary[1]

This manuscript was compiled on December 27, 2024

Obtaining predictive models of a neural system is notoriously challenging. Detailed models suffer from excess model complexity and are difficult to fit efficiently. Simplified models must negotiate a tradeoff between tractability, predictive power and ease of interpretation. We present a new modelling paradigm for estimating predictive mechanistic-like models of neurons and small circuits that navigates these issues using methods from systems theory. The key insight is that membrane currents can be modelled using two scalable system components optimized for learning: linear state space models, and nonlinear artificial neural networks (ANNs). Combining these components, we construct two types of membrane currents: lumped currents, which are flexible, and data-driven conductance-based currents, which are interpretable. The resulting class of models — which we call Recurrent Mechanistic Models (RMMs) — can be trained in a matter of seconds to minutes on intracellular recordings during an electrophysiology experiment, representing a step change in performance over previous approaches. As a proof-of-principle, we use RMMs to learn the dynamics of two groups of neurons, and their synaptic connections, in the Stomatogastric Ganglion (STG), a well-known central pattern generator. We show that RMMs are efficiently trained using teacher forcing and multiple shooting. Due to their reliability, efficiency and interpretability, RMMs enable qualitatively new kinds of experiments using predictive models in closed-loop neurophysiology and online estimation of neural properties in living preparations.

Significance Statement

Our ability to understand the nervous system has been hindered by the difficulty of constructing good predictive models of neurons and circuits. This difficulty persists despite vast accumulated knowledge of how the basic components of the nervous system work, marking a gap in our ability to explain neural dynamics in terms of underlying mechanisms. This work describes a new data-driven modelling approach for neurons and small circuits that combines predictive power with mechanistic interpretation.

Biophysical models | Deep learning | Electrophysiology | Central Pattern Generators | Neural Dynamics

A major obstacle to progress in neuroscience is the difficulty of reconciling models with data. This difficulty is particularly acute in neural circuits that have non-trivial intrinsic dynamics, where the interplay of membrane currents over many timescales dictate circuit-level properties. Biophysically detailed models are fully interpretable, but suffer from excessive complexity and are notoriously difficult to fit to data (1–10). On the other hand, purely data-driven models may fit data very well, but their general purpose structure limits mechanistic insights (11–18), and they can give spurious predictions outside of the strict domain of the data they are trained with. At the heart of these challenges is a paradox: the basic biophysical principles of neurons and networks are very well understood, yet the models we derive from such principles struggle to predict neural activity as well as their phenomenological counterparts (19). On top of this paradox, most existing methods share the common drawback of being costly to constrain in terms of data, computational resources, and time.

In this work we present a new, efficient data-driven modelling approach that navigates the middle ground between mechanistic detail and black-box models from machine learning. Our ultimate goal is to reliably estimate mechanistic neuronal behaviour during the time frame of an experiment. This goal is both pragmatic and scientific: obtaining a predictive model during the lifetime of an experiment opens the door to real-time estimation and closed-loop control of a living neural circuit. Secondly, this goal provides a strong test of our modelling assumptions, and of the efficiency and robustness of model training.

It is important to recognise existing approaches that can successfully fit experimental data and predict neural dynamics in certain settings, and contrast these with our approach and requirements. Existing approaches comprise two broad classes. The first assumes relatively simple neural dynamics and uses phenomenological models to predict spiking, to remarkable effect in some cases (1, 11, 12, 15). However, this approach cannot account for rich intrinsic dynamics seen in circuits such as central pattern generators. Secondly, phenomenological models do not account for mechanistic effects of membrane conductances, limiting interpretability and preventing their use in closed loop, predictive control experiments with dynamic

Author affiliations: [1]Department of Engineering, University of Cambridge, United Kingdom; [2]Department of Biology, Brandeis University, Waltham MA, USA 02543

Author contributions: T.B. Burghi designed the model architecture and corresponding training algorithms, performed theoretical analyses, wrote the computer code implementation of the method, helped with electrophysiology experiments, and wrote the main body of the manuscript. M. Ivanova and E. Morozova performed the electrophysiology experiments, wrote the corresponding Methods subsection, and provided feedback on the manuscript. H. Wang helped with the computer code implementation of the training algorithms. E. Marder contributed to experiment design, experimental facilities, project supervision, and manuscript feedback. T. O'Leary conceptualized the project, contributed to experiment design, and helped writing the main body of the manuscript.

The authors declare no conflicts of interest.

[1]Corresponding Author. Email: tbb29cam.ac.uk

DRAFT

clamp and neuromodulators. The second class of approaches attempts to constrain large, detailed models to data. This requires extremely precise measurements, "clean" preparations, detailed knowledge about a given neuron's ion channel types, and large computational resources. Importantly, such approaches are not yet practical to use during the lifetime of an experiment where neuronal dynamics are uncertain (2–8, 10, 20). Furthermore, it is not clear that these models capture 'true' parameters as opposed to plausible parameter subspaces of mis-specified models (21), which raises questions about their parsimony, interpretability and predictive power.

To address these difficulties, we introduce Recurrent Mechanistic Models (RMMs), a novel modelling framework that leverages the predictive power of machine learning while retaining biophysical, mechanistic intuition. RMMs, which can be framed as class of recurrent neural networks (RNNs) with biophysically motivated constraints, use a combination of linear time-invariant (LTI) state space models and artificial neural networks (ANNs) to capture the dynamics of intrinsic and synaptic ionic currents. Exploiting this model structure, we show that RMMs can be primed for different purposes in which different levels of interpretability are required. RMMs can predict the trajectories of ionic conductances when a detailed model is warranted, effectively acting as a quickly-trainable surrogate for conductance-based models. In more general cases, including most experimental settings where specific ionic currents are unknown or uncertain, we show that fully data-driven lumped currents can be used to predict the dynamics of interconnected neurons and the synaptic current between them; we also show that data-driven lumped and mechanistic currents can be combined to obtain interpretable models in a challenging experimental context.

A key advantage of RMMs is that they can be reliably trained in a matter of seconds to minutes in a consumer-grade desktop computer, enabling their use during an electrophysiology experiment. Rapid training is achieved with teacher forcing (22), which allows RMMs to be trained in a non-recurrent fashion that exploits massive parallelization. We show how to improve on teacher forcing results using a recurrent method, multiple shooting (for a technical exposition, see (23)), that can be tuned so as to trade off speed of training for prediction accuracy, while avoiding the well-known problem of exploding gradients (24).

To provide a realistic experimental test of this methodology, we estimate the dynamics of coupled neurons in the Stomatogastric Ganglion, a well-studied Central Pattern Generator (CPG) in the crab *Cancer borealis*. This circuit provides an ideal testbed for our application, as the dynamics of its neurons are complex and its function is dictated by the action of neuromodulators, which shape membrane conductance properties. We find that RMMs provide excellent predictive power during the lifetime of an experiment, and can recover features of underlying membrane conductances, enabling mechanistic interpretation.

Results

Our results introduce Recurrent Mechanistic Models (RMMs), and demonstrate that they can reliably infer and interpret complex neuronal dynamics, while also being fast enough to generate predictions during electrophysiological experiments.

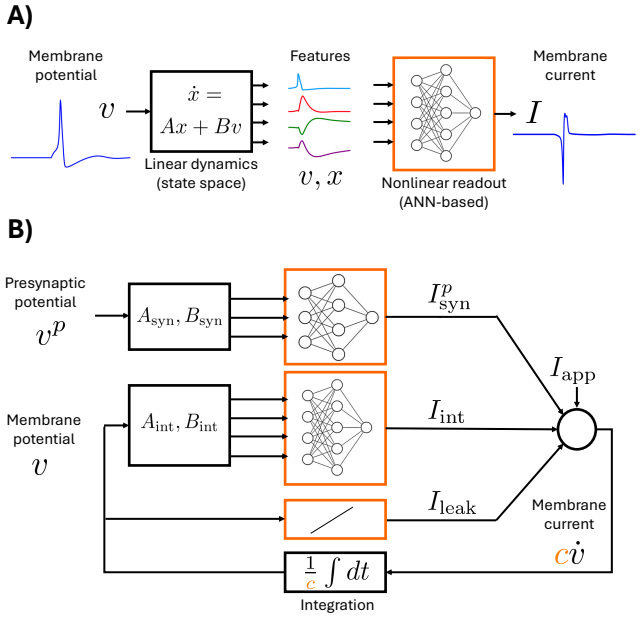

Fig. 1. The basic structure of Recurrent Mechanistic Models. **A)** A state space system in series with an artificial neural network is used to transform membrane voltages into membrane currents. State-space systems are systematically designed so that features are dynamically rich (see Methods). Modelling flexibility can be traded off for interpretability by imposing architectural constraints on the ANN. **B)** A neuronal model is obtained by integrating applied, intrinsic and synaptic membrane currents. The blocks depicted in orange are those containing the main trainable parameters of the model, which also include the membrane capacitance, c. While state space models can in principle be learned, biophysical intuition can be used to construct them and keep them fixed, leading to faster training times (see Methods). Dots indicate time derivatives; in practice, the model is implemented in discrete-time according to Eq. (1)-Eq. (6).

Recurrent Mechanistic Models. RMMs are set apart from other types of neuron models due to the use of linear time-invariant (LTI) state space systems in conjunction with artificial neural networks (ANNs) to model the ionic currents flowing through the neuronal membrane. The basic idea is illustrated in Figure 1 A: a state space system produces a set of dynamical features of the membrane voltage signal, and the features are recombined by a readout function, constructed with ANNs, to yield the membrane current. The state-space systems can be systematically designed to have orthogonal convolution kernels, resulting in dynamically rich features for the ANNs to recombine (see Methods). As the results will show, modelling flexibility can be traded off for interpretability by imposing architectural constraints on the readout function. A schematic of a RMM for a neuron with one synaptic input is shown in Figure 1 B. It is straightforward to recognise that the schematic is equivalent to the circuit diagram of a single-compartment conductance-based model (25), with intrinsic, synaptic, leak and applied currents summed in parallel and integrated to compute membrane voltage.

We now briefly describe the basic mathematical structure of RMMs; further details can be found in Methods. Since RMMs are learned from sampled data, we use discrete-time notation. The discretized derivative voltage derivative is denoted by

$$\Delta v_t := \frac{v_{t+1} - v_t}{\delta} \qquad [1]$$

where δ is the sampling period used when collecting data in an experiment. The dynamics of a single neuron in our modelling paradigm is given by

$$c\Delta v_t = -I_{\text{int},t} - \sum_p I^p_{\text{syn},t} - I_{\text{leak},t} + I_{\text{app},t} \qquad [2]$$

where, at time index t, the current $I_{\text{int},t}$ denotes the total intrinsic (ionic) current flowing through the neuron's membrane, $I^p_{\text{syn},t}$ denotes the total synaptic current due to the pth presynaptic neuron, $I_{\text{leak},t}$ denotes the leak current, and $I_{\text{app},t}$ denotes the applied current.

The total intrinsic current $I_{\text{int},t}$ models the sum of all ionic currents flowing through the neuron's membrane. In general, the intrinsic current is given by

$$I_{\text{int},t} = \sum_{i=1}^{m} \varphi_i(x_t, v_t; \theta^{(i)}) \qquad [3a]$$

where each φ_i is an ANN-based nonlinear readout function, parameterized by $\theta^{(i)}$, and x_t is the vector-valued state of a linear time-invariant system given by

$$x_{t+1} = Ax_t + Bv_t \qquad [3b]$$

with A and B matrices of the appropriate dimension. Although A and B can in principle be learned, we have empirically found a systematic choice for A and B such that state space parameters do not need to be learned to achieve high predictive performance. This is described in *constructing linear systems* in Methods.

Each learnable ANN-based readout φ_i in Eq. (3a) models an individual component of the total intrinsic current, whose level of mechanistic detail is determined by the parametric form of φ_i. To model electrical current components whose mechanisms are unknown, uncertain, or too complex to deduce, we employ **lumped currents**. Their models do not single out individual ionic types, and are constructed with a multi-layer perceptron (MLP, i.e, a black-box feed-forward ANN) according to

$$\varphi(x_t, v_t\,; \theta) = \text{mlp}(h(x_t, v_t)\,; \theta) \qquad [4]$$

where $\text{mlp}(\,\cdot\,; \theta)$ is a single-output MLP, and $h(x_t, v_t)$ is a fixed layer used to normalize and possibly reduce the dimension of the state space output vector $(v_t, x_t^\intercal)^\intercal$. To model electrical current components whose ionic mechanisms are better understood, we employ **data-driven mechanistic currents**. Their models make use of mechanistic priors, such as the reversal potential, or the type of gating dynamics of a current (activation, inactivation, or both). A typical data-driven conductance-based current is given by

$$\varphi(x_t, v_t\,; \theta) = g_t(v_t - E) \qquad [5a]$$

with

$$g_t = \bar{g}\,\sigma_+(h_+(x_t)\,; \theta)^\intercal \sigma_-(h_-(x_t)\,; \theta) \qquad [5b]$$

where E is the reversal potential and $\bar{g}$ is a conductance scaling parameter (about which we might have prior knowledge). Here, $\sigma_+(\cdot; \theta)$ and $\sigma_-(\cdot; \theta)$ are single-layer perceptrons whose weight matrices are constrained so as to model activation $(+)$ and inactivation $(-)$ gating variables, respectively. For more details on the models given by Eq. (3b)-Eq. (5), see *constructing lumped currents* and *constructing mechanistic currents* in Methods.

The justification for this particular model structure is based on practical as well as theoretical considerations. From a pragmatic viewpoint, using a single state space system to model the recurrent part of the intrinsic current dynamics greatly facilitates the choice of initial model parameters, which reduce to a sequence of time constants (Methods). State space models also have a smaller number of parameters when compared to equivalent LTI representations such as temporal convolutions. Finally, the structure of RMMs allows exploiting massive paralellization during training, since the readouts φ_i of every intrinsic current component share the same "reservoir" of internal states x_t (see *training algorithms* in Methods).

From the theoretical point of view, we are supported by the fact that any reasonable model of an ionic current (or conductance) must have the so-called Fading Memory property (27). This means that the mapping from voltage to current (or conductance) can be approximated arbitrarily well by a so-called Wiener model: a LTI system with a nonlinear static readout function (27, 28). As pointed out in (13), fading memory can be empirically observed in the input-output behaviour of ionic currents under voltage-clamp. Intuitively, fading memory holds in a population of ionic currents if clamping the membrane voltage to a particular value results in a unique steady-state current, regardless of the value of the voltage prior to clamping. It is the very reason for which neuronal excitability can be studied using steady-state IV curves.

The total synaptic current $I^p_{\text{syn},t}$ caused by a presynaptic neuron with membrane potential v^p_t is modelled analogously to the intrinsic current, the only difference being the use of v^p_t instead of v_t in Eq. (3b). Finally, the leak current is given by a linear-in-the-parameters model

$$I_{\text{leak},t} = \varsigma(\theta_{\text{leak},1})v_t + \theta_{\text{leak},2} \qquad [6]$$

where the softplus function, defined by $\varsigma(\cdot) = \log(1 + \exp(\cdot))$, is used to keep the leak conductance positive throughout training. The leak model Eq. (6) can also be re-parametrized in terms of maximal conductance and reversal potential.

RMMs recover conductance trajectories from current-clamp data. The first question we ask is whether RMMs are able to predict the membrane voltage and conductance trajectories of conductance-based models, based solely on applied current–voltage data. To test this, we employed data-driven conductances (Eq. (5)) to design an RMM whose currents mimic the sodium and the potassium currents of the Hodgkin-Huxley neuron (26). The structure of the resulting total intrinsic current is illustrated in Figure 2 A, and the parameters are described in detail in *Hodgkin-Huxley RMM* in Methods. In constructing this model, we exploit *timescale separation*: the state space system is defined with four "fast" states evolving roughly in the timescales of sodium activation (0.1, 0.3, 0.5, and 0.7 milliseconds), and four "slow" states corresponding to sodium inactivation and potassium activation (1, 3, 5, and 7 milliseconds). While the fast state components are fed to the sodium activation readout, the slow state components are fed to both the sodium inactivation and the potassium inactivation readouts. In addition, the fixed layers at the input of the activation and inactivation readouts are designed

DRAFT

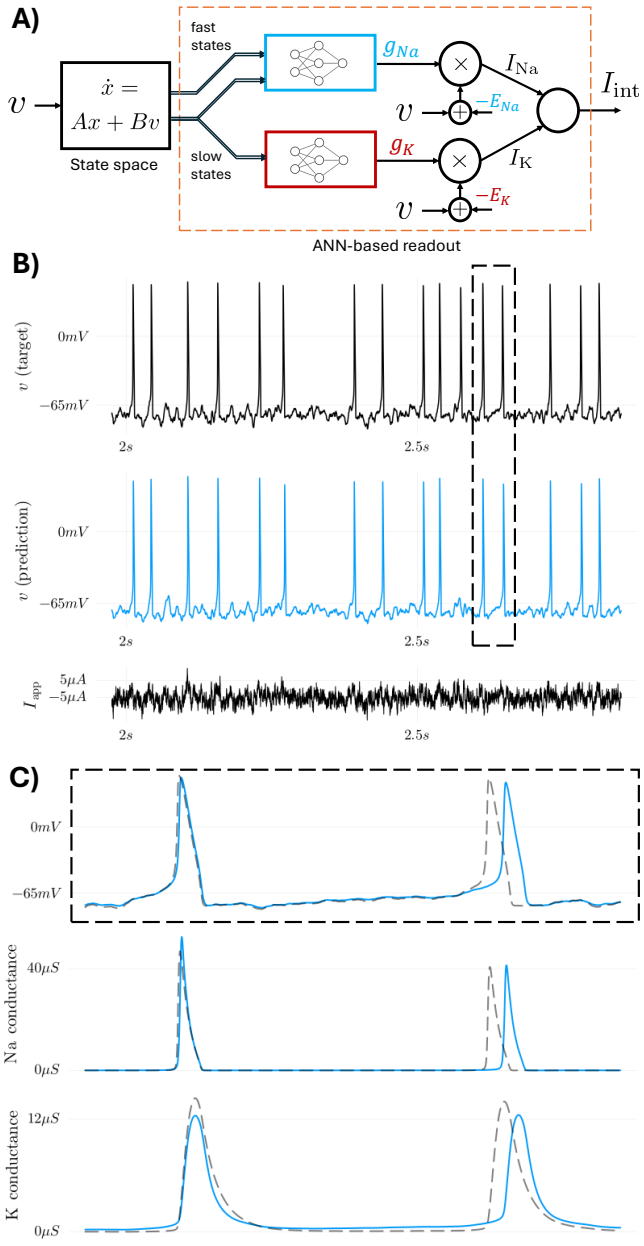

Fig. 2. RMMs recover the classical Hodgkin-Huxley conductances (26) from input-output data. **A)** A state space system with eight internal states is designed so that the states evolve in fast and slow timescales roughly corresponding to those of the spiking dynamics of the original HH model. The ANN-based readout is constrained to promote interpretability (cf. Figure 1): sodium and potassium conductances are obtained as outputs of MLPs, and the corresponding ionic currents are obtained by multiplying those data-driven conductances by a difference in potential $(v - E_{\text{ion}})$, with E_{ion} a prior reversal potential. To ensure the model learns the correct currents, fixed layers selecting fast and slow states are used. **B)** Target voltage trace (top) resulting from simulating the original HH model given a noisy applied current, and corresponding RMM prediction (bottom) on held-out test data. **C)** (top) zoom of the dashed region in (B) shows accuracy of spike prediction. (middle, bottom) The RMM accurately recovers conductance trajectories during spiking. For modelling and training details, see *Hodgkin-Huxley RMM* in Methods.

so that, at randomly initialized weights and biases, both mechanistic currents are roughly inactivated outside of the range of voltages relevant to the HH model's dynamics (see Methods).

After training a randomly initialized Hodgkin-Huxley RMM with regularized teacher forcing (TF) on a current-clamp dataset, we simulated the trained RMM on held-out validation data, obtaining the the predictions shown in Figure 2 B and C (for a description of TF and the training dataset, see Methods). It can be seen that the learned RMM predicts not only the spike times of the target HH model, but also its ionic conductance trajectories. In Figure S1, we reproduce these results for five different RMMs initialized and trained with different random number generator (RNG) seeds chosen arbitrarily; all models are obtained around the same training epoch, and display similar predictive power at the level of spike times and conductance trajectory shapes. Notice that the small difference in the shape of conductance traces is an expected consequence of model mismatch, and of the fact that conductance traces are not used in the cost function of the training problem (which only uses applied current and voltage signals).

Experimental validation of the RMM approach. While data-driven conductances (Eq. (5)) can make RMMs fully interpretable, in most experimental scenarios the precise ion channel makeup of a given neuron is not readily available, providing poor biophysical priors to work with. In addition, quantitatively predicting the response of neuronal circuits *in vitro* and *in vivo* often requires using multi-compartment models whose parameters are difficult if not impossible to estimate based on a single voltage signal. This section shows how single-compartment RMMs can address these problems through judicious use of lumped currents (Eq. (4)).

As a proof-of-principle that RMMs can be used to estimate the behaviour of neurons with complex morphology and uncertain biophysics *solely from applied current and voltage measurements*, we performed experiments to predict neuronal dynamics in a well-studied system: the Stomatogastric Ganglion (STG) of the crab *Cancer borealis*. We use the experimental setup illustrated in Figure 3 A: after pharmacologically blocking glutamatergic synapses of the STG, a simple feed-forward network is obtained where the Lateral Pyloric (LP) neuron is synaptically driven by two Pyloric Dilator (PD) neurons via cholinergic synapses. Both PD neurons are strongly coupled to the Anterior Burster (AB) neuron, a bursting pacemaker. As an approximation, one can think of this as a feedforward circuit where a strongly coupled subcircuit consisting of the two PD neurons and the AB neuron drives the single LP neuron. The only data used for training our models in this section consists of the applied currents and voltage signals from the LP neuron and one of the PD neurons. The 'true' PD-AB dynamics in particular are thus extremely complex, owing to the morphology of the cells as well as the fact that the PD measurement signal accounts for the combined dynamics of three coupled neurons.

Rapid prediction of total intrinsic and synaptic currents. To test the predictive power of RMMs constructed exclusively with lumped currents, we trained models to predict the responses of the LP neuron in Figure 3 A. The LP neuron RMM is constructed according to the diagram in Figure 1 B,

DRAFT

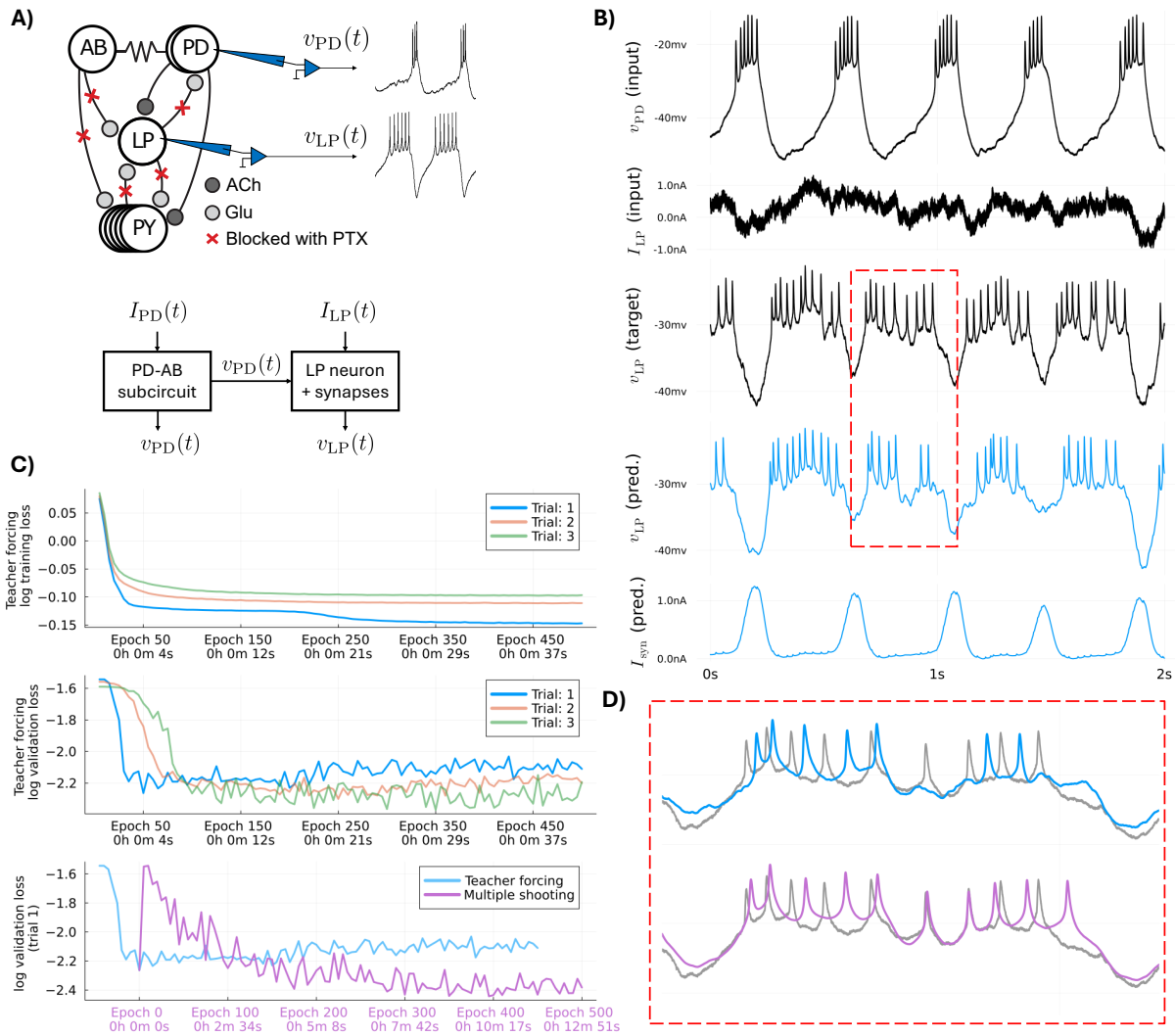

Fig. 3. Estimating a RMM model of a neuron with complex spatiotemporal dynamics. **A)** Experimental setup used to generate data for fitting an RMM: the STG circuit is bathed in PTX, which blocks Glutamate synapses but not Acetylcholine synapses; intracellular recordings are taken of *one* of the PD neurons and the LP neuron; the resulting system is a feedforward interconnection with the measured presynaptic PD neuron voltage driving the LP neuron. **B)** Example voltage traces showing held-out data (three top panels) and the predicted voltage and synaptic currents from an RMM trained with teacher forcing (see Methods). **C)** Evolution of the loss functions during training for three different trial datasets; each dataset consists of around 2 seconds of data recorded with a sampling period of 0.1 ms. The highlighted (blue) Trial 1 yields the model used an example in (B); predictions of the models trained on Trials 2 and 3 are shown in Figure S2. Top: loss function value during training with teacher forcing, evaluated with held-out data; middle: validation loss function value using models trained with teacher forcing; bottom: multiple shooting improves the performance of the best model obtained with teacher forcing. **D)** Example traces demonstrating the improvement obtained with multiple shooting. For a description of teacher forcing and multiple shooting, as well as the loss function used for model validation, see Methods.

with lumped ANN readouts given by Eq. (4) for both intrinsic and synaptic currents (for details, see *LP neuron RMM* in Methods). To generate a dynamically rich, time-varying input to LP, we injected noisy currents given by an Ornstein-Uhlenbeck process, while monitoring both LP and PD membrane potentials. After fitting an RMM we analysed the predictive performance of the model on held-out test data (Figure 3 B) and found good quantitative agreement between predicted LP membrane voltage (blue traces) and measured LP membrane voltage. The model also enabled us to infer the total synaptic current I_{syn}, which varied as expected with PD bursts to generate intermittent pauses in LP spiking.

Over multiple fitting runs with validation on held-out data, we found that good steady-state prediction error is achieved independently of initial model parameters and datasets (Figure 3 C). In Figure 3 C (top), we plotted training losses for three different trials, corresponding to datasets collected from the same preparation at different times. Blue curves correspond to the model of Figure 3 B, while orange and green curves correspond to models whose predictive performance is showcased in Figure S2. Teacher forcing losses decrease and stabilize rapidly, under a minute. These results thus demonstrate a crucial feature of RMMs: the fact that reasonably good prediction accuracy, under large system uncertainty, can be achieved efficiently, with prediction error asymptoting in several tens of seconds on modest hardware (see Methods).

The learning performance of RMMs is a result of using an efficient training method based on teacher forcing (Methods). As with all machine learning methods, the decrease of training

loss during teacher forcing does not guarantee a decrease in predictive power, as shown in Figure 3 (C, middle), where validation losses are plotted for "model snapshots" taken at different epochs during training (see Methods for a description of training vs. validation losses). Thus, while teacher forcing quickly trains the model, it also leads to overfitting, making early stopping or strong regularisation essential.

If necessary, predictive accuracy can be enhanced further using a generalisation of teacher forcing called multiple-shooting (see Methods). The result of this additional training on predictive accuracy is shown in Figure 3 (C, bottom), while Figure 3 D shows how spiking traces from the zoomed in portion of A are improved with this extra step of training. Starting from a model trained with teacher forcing, multiple shooting improves model performance at the cost of longer training times, up to several minutes on this dataset. The tradeoff between predictive accuracy and training time can therefore be dictated by the requirements of the experiment, and can yield very good results in minutes.

Predicting and interpreting endogenous bursting. We next used the RMM framework to learn an interpretable single-compartment model of the bursting PD-AB subcircuit, often called the pacemaker kernel, in the same experimental setup of Figure 3 A. Mechanistically, bursting emerges from a complex combination of sodium, potassium, and calcium ion channels which are unevenly distributed in the system. Despite the complexity, modelling the pacemaker kernel with a single burst-excitable compartment is made possible by the strong electrical coupling between AB and PD neurons, which allows us to assume that the slow component of the bursting wave is roughly the same across those neuron's membranes. However, quickly fitting mechanistic models of the pacemaker kernel remains difficult. In particular, the sodium channels driving intra-burst spikes are located in a region of the PD neuron's axon far from the recording site in the soma, as illustrated in Figure 4 A. As a result, fast intra-burst spikes in the recordings are heavily filtered versions of the PD neuron's spikes (see (29, 30) for details).

To deal with this challenging scenario, we used a combination of lumped and mechanistic data-driven currents to learn the dynamics of the AB-PD pacemaker, as depicted in Figure 4 B, and asked whether the resulting RMM would yield the appropriate intrinsic mechanism for generating endogenous bursting. The RMM is constructed with three mechanistic currents representing the aggregate dynamics of calcium currents (I_{Ca}), potassium (including calcium-activated) currents (I_{K}), and hyperpolarization-induced currents (I_{H}). To model the intra-burst spiking dynamics while constraining the RMM to a single compartment, a lumped current (I_{lumped}) using fast state components is used; this current is more heavily regularized during training than mechanistic currents (see *PD neuron RMM* in Methods).

Figure 4 C, top, shows example voltage traces of a PD neuron recording. The voltage trace shown was held out from training and used to validate a pacemaker RMM, whose predictions are shown in blue in Figure 4 C. The RMM, which was trained in around two minutes (Figure S6), predicts the target dataset well. In addition to predicting the target voltage, the structure of the RMM allows recovering the lumped membrane current, representing the fast axonal current, as well as the data-driven conductances associated to the aggregate calcium, potassium, and H currents. With minimal a priori information, the RMM is able to learn the functional role of the three types of currents: the aggregate potassium current activates following a burst, hyperpolarizing the membrane; the aggregate calcium current drives the upstroke of the slow bursting wave; and the aggregate H current activates following hyperpolarization, helping to initiate the burst.

To evaluate the consistency of the RMM method, we fitted RMMs to four other PD neuron preparations, using the same hyperparameters as those used to obtain Figure 4 — except for the overall regularization constant ρ, which was tuned to deal with differing noise levels. The results are shown in Figure S5, where it can be seen that even though the recordings are done in the same type of cell, the features of the recorded voltage traces are reasonably different, due to the intrinsic variability between individual preparations. In Figure 4 D we plot the steady-state IV curves of the inferred aggregate potassium and calcium currents across all five preparations. The steady-state IV curves all overlap in the range usually associated to the activation of calcium currents in PD neurons of the STG, indicating self-consistency among the preparations, as well as consistency with known voltage-clamp properties of these currents form previous work. We note, that since the models were trained on current-clamp data alone, we would expect that the steady-state IV curves shown in Figure 4 D are subject to greater uncertainty than current-clamp voltage predictions, such as those in Figure 4 C (blue traces). However, these results indicate that RMMs can extract consistent, mechanistically relevant features of membrane physiology without the need for voltage-clamping.

To further investigate whether RMM predictions at the level of individual conductances and currents are dynamically meaningful and generalizable, we probed the behaviour of the RMM after removing the lumped axonal fast current from the model. Figure 4 E shows the result for the same model as Figure 4 C. In the left panel, it can be seen that with both lumped and mechanistic currents, and no external excitation, the RMM generates an endogenous bursting rhythm. This demonstrates generalization with respect to training data, as the model was not trained with a constant current. Removing the fast lumped current results in a slow wave, shown in the right panel, proving the point that the mechanistic currents alone are sufficient to drive the endogenous rhythm.

Discussion

Unlike some areas of physical sciences, where quantitative predictions can determine the validity of a theory, model predictions in neuroscience often have a more equivocal character. A poor fit to data can imply incorrect modelling assumptions, low quality data, inappropriate fitting methods, or any of these issues simultaneously. This ambiguity persists even when modelling reduced preparations consisting of a small number of neurons, and where tight control of experimental conditions and accurate measurements are possible. These challenges have preoccupied neurophysiologists for decades, and the limited success in handling them has hindered the field's ability to apply vast accumulated knowledge of low-level physiology to systems-level questions.

The approach we have introduced here does not solve all these problems, but it does permit theoretically grounded

DRAFT

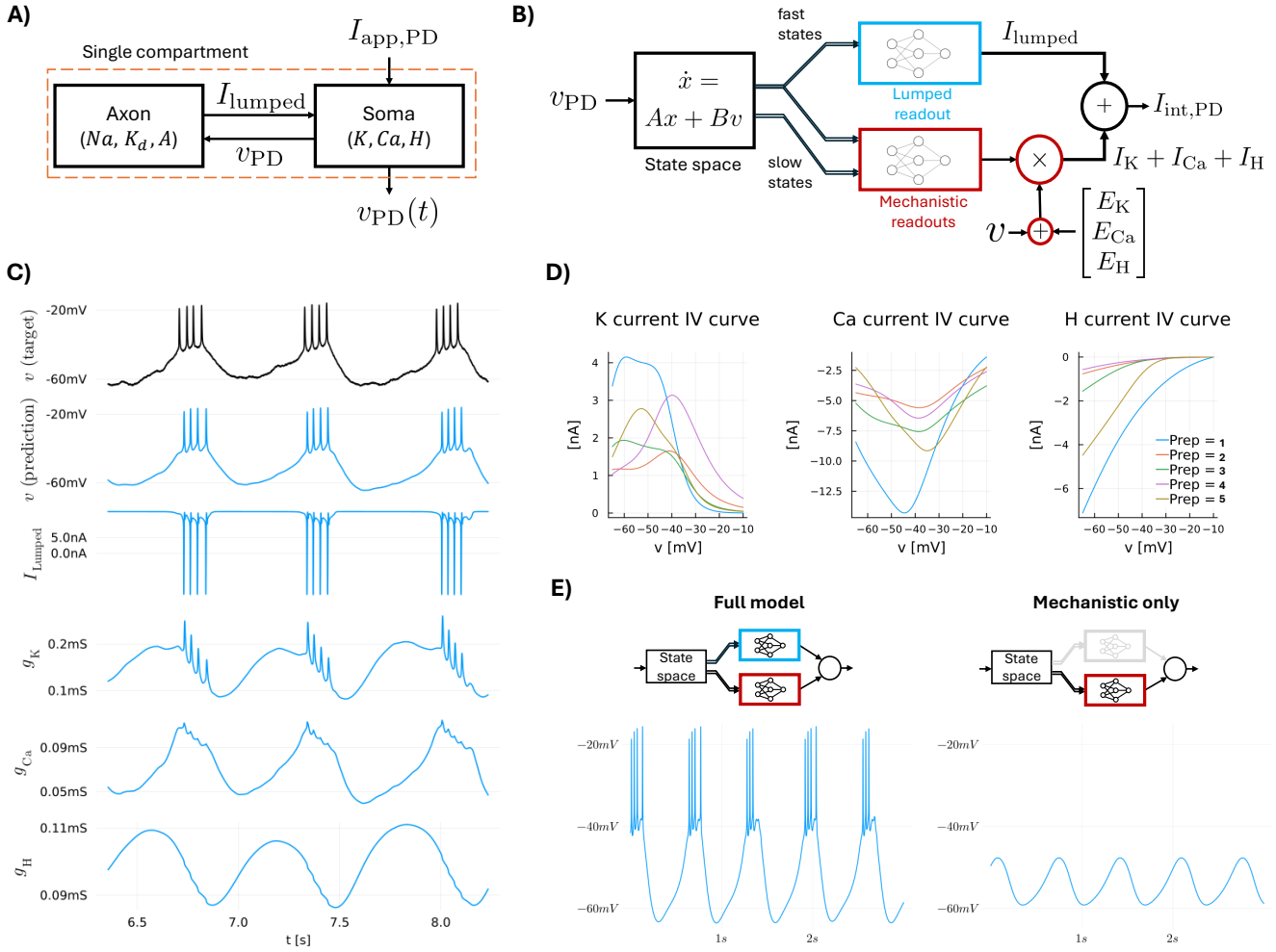

Fig. 4. Training a RMM with lumped and mechanistic branches to learn endogeneous bursting rhythms. **A)** a diagram representing the dynamics of the AB-PD pacemaker, simplified here as a single PD cell (30). Fitting traditional mechanistic models to the system is challenging: a single-compartment conductance-based model cannot quantitatively predict the membrane voltage at the soma (where recordings are made), since fast sodium and potassium currents are located in a distant part of the axon. **B)** To quickly obtain a predictive single-compartment model, we employ an RMM with a mix of lumped and mechanistic currents. The lumped current, which does not distinguish between ion channels and does not require mechanistic structure, is used to predict the fast axonal current. To model the calcium and potassium channels of the soma, we employ three data-driven mechanistic currents to capture the aggregate dynamics of potassium, calcium, and hyperpolarization-induced currents. **C)** Validation of an RMM model on held-out membrane voltage data (black traces). The model, which was trained on current-clamp data (see *PD neuron RMM* in Methods) predicts the membrane voltage well. In addition, the model allows interpreting its internal variables as individual currents and conductances. Figure S5 shows similar results for other PD neuron preparations. **D)** Steady-state IV curves of four RMMs fit to different PD neuron preparations show that RMMs are able to learn the basic mechanisms of bursting based on calcium and potassium currents. **E)** Removing the fast lumped current from the RMM uncovers the slow wave dynamics underlying burst excitability. All models in this figure and Figure S5 were trained with teacher forcing (see methods) over the course of 200 epochs, corresponding to about two minutes of training (see Figure S6 for training curves).

predictive models to be inferred during the course of an experiment, which drastically enhances our ability to solidify theories of neural circuit function, and to use these theories to practical ends. Our goal was to find a model that captures what we know about biophysics without assuming more than can be measured, and to find a means of constraining the model to produce good predictive accuracy on a timescale of minutes.

Achieving this goal forced us to sacrifice a modelling formalism that is often placed among the most successful theoretical developments in all of neuroscience: the detailed Hodgkin-Huxley form of conductance based modelling (26). This may seem tantamount to abandoning any semblance of 'mechanism' in our modelling approach. Most mechanistic models are conductance-based and strive to account for ion channel gating dynamics and even the spatial distribution of channels in complex morphologies. Such models are fundamental for understanding how low level biophysics shapes the dynamic properties of neurons (30–33) and in some cases constitute predictive models that fit data very well (10). However, in general, the empirical success of this modelling paradigm has been modest, especially when applied to more typical neural circuits that possess many tens of ionic conductance types, complex morphologies and physiology that hinders precise experimental measurements. Constraining the parameters of detailed models under such conditions is notoriously difficult (34).

In addition to the practical difficulty of parameter estimation in detailed models, there is a deeper, scientific problem with attempts to capture detail. Regardless of

their detail, models likely omit unknown structural and biophysical properties that are impractical to measure (35). In other words, the strict form of a detailed conductance based model that supposedly describes a given type of neuron is almost certainly wrong, and the parts of it that are right might constrain the model's dynamics in unwanted ways, and even introduce artefacts (36). This simultaneously impacts predictive power and interpretability: inferred parameter values of a misspecified model, however precise, may have no sensible physical meaning.

Our approach acknowledges our ignorance of the extant pool of conductances and synaptic inputs in a preparation, their spatial distribution, phosphorylation state, splice variants and the remaining plethora structural variation that is known to exist but cannot be measured. Instead, we only model what we can measure, representing this information in terms of known biophysics, such as early and late IV relationships, and, where possible, inward and outward current components. Ultimately these are the features of the underlying biophysics that dictate a neuron's dynamics, and in this sense, they are the most parsimonious, mechanistic inference we can be confident of in a typical experimental setting.

Even though the experimental system we chose is well characterised, with a relatively simple connectivity, unobserved inputs and dynamics in the STG abound in the data. Although we blocked fast inhibitory transmission with PTX, there are numerous modulatory inputs coming from the front-end ganglia that are still present in the preparation that affect several intrinsic currents (37) in a manner that depends on the wider state of the preparation (38). Gap junctions between the membranes of different cells introduce further uncertainty (39). PD is tightly electrically coupled to AB, but it is also loosely coupled to LPG, which in turn is connected to VD, and several other cells. LP is connected to IC and multiple PY cells. Furthermore, the strength of gap junctions might be susceptible to modulation (40). These various inputs and perturbations are not accounted for in the model because we are unable to measure them directly, it is therefore far from trivial to be able to predict dynamics in this circuit with any degree of accuracy.

As with most models of the neuronal membrane, RMMs as presented in this paper assume that the applied current is injected at the site of membrane recordings. This assumption is informed by our choice of Discontinuous Current Clamp (DCC) as the method for current injection. DCC is practical, as it allows voltage to be measured simultaneously with the injection of applied current, using a single electrode. It is well-known that DCC results in significantly higher levels of measurement noise in the data. While improved results can in principle be obtained with two-electrode recordings and continuous-current clamp, in that case the effect of filtering of the fast components of the applied current by the neuronal membrane should be taken into account by the model.

Methods

Results in this paper were obtained with code written in Julia and rely on the Flux.jl package (41). The code used to produce the results above can be found in `https://github.com/thiagoburghi/RMM`. In what follows we denote the all-ones and all-zeros vectors by $\mathbf{1}$ and $\mathbf{0}$, respectively.

Constructing linear systems. The state space matrices A and B in the total intrinsic current dynamics Eq. (3) are constructed systematically. First, we choose a sequence of n positive time constants $\tau_1, \tau_2, \ldots, \tau_n$ believed to play a role in the collective dynamics of the intrinsic currents; second, we use the sequence $\{\tau_j\}$ to construct a discrete-time state space system. The sequence $\{\tau_j\}$ is most easily constructed by sampling the time constant functions of an existing "prior" conductance-based model[1]. Letting $\tau_w(v)$ denote the time constant function of a gating variable w in such a prior model, we set the smallest and largest time constants as

$$\tau_1 \approx \min_{v \in V_{\text{obs}}, w} \tau_w(v), \quad \tau_n \approx \max_{v \in V_{\text{obs}}, w} \tau_w(v),$$

where $V_{\text{obs}} = [v_{\min}, v_{\max}]$ is the observed membrane voltage range of the data. Intermediate time constants are chosen incrementally between τ_1 and τ_n so as to roughly represent the range of values observed in all the $\tau_w(v)$. As an illustration of such time constant functions, in Figure S3 we plot the $\tau_w(v)$ of the LP neuron model of (42), which were used to inform the choices of $\{\tau_j\}$ in this paper.

Once $\{\tau_j\}$ is given, we compute a sequence $\{\lambda_j\}$ of eigenvalues for A (i.e., discrete-time system poles) by applying the zero-order-hold transformation

$$\lambda_j = \exp(-\delta/\tau_j), \quad j = 1, \ldots, n \qquad [7]$$

where δ is the sampling period of the data. This transformation leads to a stable state space system; see (43) for sampled-data system theory. The eigenvalues in Eq. (7) are then used to construct a *state space realization*: a choice of A and B such that the eigenvalues of A are given by λ_j. This paper considers two types of realizations:

1. A *diagonal* realization where $A = \text{diag}\{\lambda_j\}$ is a diagonal matrix, and $B = (\mathbf{1} - \text{vec}\{\lambda_j\})$.

2. A realization with *orthogonal impulse responses*[2] (44). The SI Appendix describes how such a realization can be constructed from $\{\lambda_j\}$.

An important fact about the two realizations above is that their DC (i.e., steady-state) gains $(I-A)^{-1}B$ are elementwise positive. Different realizations of A and B affect the learning algorithms by changing the trajectories of the inputs to ann-based readouts. For illustrative purposes, in Figure S4 we compare the "spike responses" of a diagonal state space system with those of a system with identical A matrix eigenvalues, but orthogonal impulse responses; the figure shows how the states of the latter overlap less in time, an effect we have found to be beneficial when training lumped currents. Intuitively, a system with orthogonal impulse responses generates a rich set of signals for ANNs to recombine.

Constructing lumped currents. The MLPs used to model lumped readouts in Eq. (4) are given by the chained composition of mappings

$$\text{mlp}(\cdot\,;\theta) := W_{L+1} \bigcirc_{\ell=1}^{L} \tanh(W_\ell \cdot + b_\ell) \qquad [8]$$

[1] When such a model is not available, time constants can alternatively be based on the frequency content of the voltage recording

[2] The impulse response of a LTI system is the kernel of that system's convolution representation.

DRAFT

where L is the number of MLP layers, W_ℓ are weight matrices, b_ℓ are bias vectors, and $\tanh(\cdot)$ applies the hyperbolic tangent elementwise to its inputs. In this work, the weight matrices are initialized with the Nguyen-Widrow heuristic (45), which is optimized for MLP inputs that lie in the interval $[-1, 1]$ (see SI Appendix). The fixed layer $h(x_t, v_t)$ of a lumped current is given by a non-learnable affine mapping used to normalize v_t, as well as to select and normalize a subset of states in x_t which is passed as input to the MLP. In our results we have used min-max normalization, which maps the minimum and maximum values of the inputs of the MLP seen during training to -1 and $+1$, respectively (see Appendix SI for details). Lumped currents have empirically been found to work better with state space models with orthogonal impulse responses, see *LP neuron RMM* and *PD neuron RMM* below.

Constructing mechanistic currents. The $\sigma_+(\cdot;\theta)$ and $\sigma_-(\cdot;\theta)$ mappings of mechanistic readouts in Eq. (5) are given by

$$\sigma_\pm(\,\cdot\,;\theta) = W_2^\pm \sigma(W_1^\pm \cdot + b_1^\pm) \qquad [9]$$

where $\sigma(\cdot)$ applies the logistic function $(1 + \exp(-\cdot))^{-1}$ elementwise to its inputs. The sign of the elements of the weight matrices in Eq. (9) are constrained so as to implement monotonically increasing (σ_+) and decreasing (σ_-) mappings, in the sense that if $u_2 \geq u_1$ elementwise, then $\sigma_+(u_2) \geq \sigma_+(u_1)$ and $\sigma_-(u_2) \leq \sigma_-(u_1)$, also elementwise. This is done to model the activation and the inactivation of a current, respectively. The weight constraints are implemented by setting $W_1^\pm = \pm\varsigma(\tilde{W}_1^+)$ and $W_2^\pm = \varsigma(\tilde{W}_2^\pm)$, where $\varsigma(\cdot)$ applies the softplus function $\log(1+\exp(\cdot))$ elementwise to its inputs. The matrices $\tilde{W}_i^\pm$ are unconstrained, and contain the trainable weights of a mechanistic readout; they are initialized randomly, but in such a way that

$$\begin{aligned} &\mathbb{E}\{\sigma_+(u;\theta)\} \approx \mathbf{0}, \quad \mathbb{E}\{\sigma_-(u;\theta)\} \approx \mathbf{1}, \quad u = -\mathbf{1} \\ &\mathbb{E}\{\sigma_+(u;\theta)\} \approx \mathbf{1}, \quad \mathbb{E}\{\sigma_-(u;\theta)\} \approx \mathbf{0}, \quad u = +\mathbf{1} \end{aligned} \qquad [10]$$

See the SI appendix for the detailed initialization heuristic.

The fixed layer $h(x_t)$ is a non-learnable affine normalisation mapping. It is used to select a subset of states of x_t, and normalize those states. Normalisation implements a biophysical prior: based on a prior activation or inactivation voltage interval $[v_{\text{low}}, v_{\text{high}}]$, $h(x_t)$ maps the steady-state values of $x_{i,t}$ at v_{low} and at v_{high} to -1 and $+1$, respectively (SI Appendix). Notice that as a consequence of Eq. (10), this ensures the network is initialized in such a way that currents respect the prior activation/inactivation range $[v_{\text{low}}, v_{\text{high}}]$.

Mechanistic readouts can read from either diagonal state space systems or systems with orthogonal impulse responses. But because of the constraints placed on the weight matrices of $\sigma_\pm(\cdot;\theta)$, the choice of state space realisation is critical to the those currents. Reading from diagonal realisations results in "simple" currents where a monotonically increasing voltage trajectory leads to monotonically increasing activations σ_+ and monotonically decreasing inactivations σ_-. This is a straightforward fact from *positive systems theory* (46) (see details in the Appendix SI). Reading from realisations with orthogonal impulse responses results in more flexible (but also more degenerate) models, since in this case the preservation of the monotonicity of voltage trajectories is not guaranteed. Such realisations can be used in conjunction with mechanistic readouts to model complex ionic currents, such as calcium-activated potassium currents. Notice however the DC gain $(I - A)^{-1}B$ of such realisations is still elementwise positive. Hence, monotonically increasing/decreasing *steady-state voltages* lead to monotonically increasing/decreasing steady-state activations σ_+, and monotonically decreasing/increasing steady-state inactivations σ_-. This results in a biophysical constraint in the IV curves of the corresponding ionic currents.

In mechanistic currents, it is possible to gain further modelling flexibility by modifying Eq. (5) as

$$\begin{aligned} \varphi(x_t, v_t\,;\theta) = \bar{g}\,(&\sigma_+(h_+(x_t)\,;\theta) \odot \\ &\sigma_-(h_-(x_t)\,;\theta))^\intercal\,(\mathbf{1}v_t - E) \end{aligned} \qquad [11]$$

where $\odot$ denotes the elementwise (Hadamard) product, and now E is a vector of Nernst potentials. This can be used when the Nernst potential of a current is uncertain, as is often the case with Calcium currents; the Nernst vector E acts as extra parameters of the data-driven mechanistic current.

Training RMMs. Rapid training is achieved with two related training methods that avoid well-known difficulties commonly encountered when training mechanistic models (47). We consider a dataset given by $\{I_{\text{app},t}, \bar{v}_t\}_{t\in[0,T]}$ consisting of measured applied current $I_{\text{app},t}$ and voltage targets $\bar{v}_t$. The main difficulty is that of so-called *exploding gradients* (24, 48). When dealing with spiking signals, a naive attempt to minimize the mean-squared error $\frac{1}{T}\sum_{t=1}^{T}\|\bar{v}_t - v_t\|^2$ with gradient-based methods leads to exploding cost function gradients as the dataset horizon T increases. This occurs due to the high sensitivity of spiking models to changes in their parameters (47), which in turn is due to the existence of bifurcations in neural dynamics (31). The two methods outlined below adapt the techniques of teacher forcing (22) and multiple-shooting (23) to the context of RMMs, eliminating exploding gradients altogether.

Teacher forcing (TF) with backpropagation. Teacher forcing (TF) (22) adapted to RMMs is formulated as the supervised learning problem

$$\mathcal{C}(\theta, T) = \frac{1}{T}\sum_{t=0}^{T-1}\|I_{\text{app},t} - \hat{I}_{\text{app},t}(\theta)\|^2 + \sum_i \rho_i(\theta_i - \bar{\theta}_i)^2 \qquad [12]$$

where $I_{\text{app},t}$ is the measured applied current, $\hat{I}_{\text{app},t}$ is an applied current estimate, ρ_i are regularization constants, and $\bar{\theta}_i$ are parameter priors. We impose $\rho_i = 0$ whenever θ_i is either a layer bias or an element of the $\tilde{W}_2^\pm$ matrices of a mechanistic current. We also impose $\bar{\theta}_i = 0$ whenever θ_i is not a Nernst potential. TF, which has been applied to conductance-based models in (20), uses voltage measurements to estimate the applied current. Manipulating Eq. (2), we find that the current estimate is

$$\hat{I}_{\text{app},t} = -c\Delta\bar{v}_{t+1} + \hat{I}_{\text{int},t} + \sum_{p\in\mathcal{P}} \hat{I}^p_{\text{syn},t} + \hat{I}_{\text{leak},t} \qquad [13]$$

where

$$\hat{I}_{\text{int},t} = \sum_i \varphi_i(\hat{x}_t, \bar{v}_t; \theta^{(i)}_{\text{int}}) \qquad [14a]$$

$$\hat{x}_{t+1} = A_{\text{int}}\hat{x}_t + B_{\text{int}}\bar{v}_t \qquad [14b]$$

DRAFT

is used to compute the internal current estimate, with $\hat{I}^p_{\text{syn},t}$ and $\hat{I}_{\text{leak},t}$ being computed analogously using the measured $\bar{v}^p_t$ and $\bar{v}_t$, respectively. Eq. (12) implements TF because training is carried out by replacing v_t by the measured (teacher) signal $\bar{v}_t$ in the model dynamics. The formulation has roots in the *equation-error* method of system identification (49).

In the SI Appendix we show that when applied to RMMs with fixed A and B matrices, TF allows massively parallelized GPU computations during training, which is the reason why RMMs can be trained so quickly. In short, when A and B need not be trained, the estimated applied current $\hat{I}_{\text{app},t}$ in Eq. (13) effectively becomes a feed-forward neural network, whose inputs are given by the the voltage signal $\bar{v}_t$ and the corresponding pre-simulated internal states. Since those inputs can be concatenated into a large matrix, this implies that Eq. (12) can be solved using one's favourite backpropagation-based gradient descent algorithm, while exploiting efficient matrix-based computations in the GPU.

Multiple shooting with backpropagation through time. Teacher forcing (TF) comes with the downside that since the model is trained to predict the applied current, it might not predict voltage satisfactorily. Although in our results we have shown that TF yields very good results on experimental data, better models can be found by exploiting multiple shooting (23), which we adapt to RMMs, and which generalizes the TF method above. In multiple shooting, a dataset of length T is divided into N smaller intervals of length T_s, with N the largest integer such that $NT_s \leq T$. To construct a training cost function, the model being fit is simulated N times, with the initial conditions of each simulation defined independently. Mathematically, for each $n = 0, 1, \ldots, N-1$ and each $t = 0, 1, \ldots, T_s - 1$, we define the inputs $I^{(n)}_{\text{app},t} = I_{\text{app},t+nT_s}$ and simulate the system to obtain N state vectors, which we denote by

$$z^{(n)}_t := (v^{(n)}_t \,;\, x^{(n)}_{\text{int},t} \,;\, x^{(n)}_{\text{syn}_1,t} \,;\, x^{(n)}_{\text{syn}_2,t} \,;\, \ldots)$$

and which are initialized according to $z^{(n)}_0 = \zeta^{(n)}$, where the $\zeta^{(n)}$ are arbitrary initial conditions. In practice, good initial conditions are given by the measurements and the teacher-forced states, i.e.

$$\zeta^{(n)} = (\bar{v}_{nT_s} \,;\, \hat{x}_{\text{int},nT_s} \,;\, \hat{x}_{\text{syn}_1,nT_s} \,;\, \hat{x}_{\text{syn}_2,nT_s} \,;\, \ldots) \qquad [15]$$

with $\hat{x}_{\text{int},nT_s}$ obtained from Eq. (14).

The key feature of multiple shooting is that the $\zeta^{(n)}$ are trained alongside θ so as to minimize the discrepancy between the state at the end of the nth simulation, $z^{(n)}_{T_s}$, and the initial condition of the $(n+1)^{\text{th}}$ simulation, $\zeta^{(n+1)}$. This is accomplished by solving the optimization problem given by

$$\min_{\theta,\zeta} \quad \sum_{n=0}^{N-1} C_n(\theta, \zeta^{(n)}) + \|z^{(n)}_{T_s} - \zeta^{(n+1)}\|^2_D + r(\theta;\rho) \qquad [16a]$$

where $\|u\|^2_D := u^\intercal D u$ is a D-weighted 2-norm,

$$C_n(\theta, \zeta^{(n)}) = \frac{1}{T_s} \sum_{t=0}^{T_s-1} \|\bar{v}_{t+1+nT_s} - v^{(n)}_{t+1}\|^2 \qquad [16b]$$

are the cost functions associated to each of the N simulations, and $r(\theta;\rho)$ is the same parameter regularization term as in Eq. (12). The problem Eq. (16) is an unconstrained optimization problem on the parameters θ and initial conditions $\zeta^{(n)}$; it can be solved efficiently with gradient descent. When training with multiple shooting, computing the gradient of the cost function in Eq. (16a) requires performing backpropagation through time over the T_s timesteps of each parallel simulation; to avoid exploding gradients one must keep T_s small enough. The size of the shot size T_s is a hyperparameter of the problem; larger shot sizes may yield better predictions at the cost of slower learning[3]. Multiple shooting can be sped up by exploiting parallelization, as explained in the SI Appendix.

Validating RMMs. All models in this paper were validated on held-out training data as follows: a "snapshot" of models during training was saved so that each training run yielded around 100 snapshot models; after training, snapshot models were simulated using Eq. (2) and the validation I_{app} (this was done using parallel computing to speed up the validation process), and the best model was chosen as the result of the training procedure. This was done as a means to prevent overfitting, and to recover loss curves. Validation of all 100 snapshot models did not exceed two minutes for any of the results in this paper. The best snapshot model was determined by comparing its predicted spike times (including intra-burst spikes) to the validation voltage spike times. This was done by computing a validation cost given by

$$\mathcal{V}(\theta, T) = \frac{1}{T} \sum_{t=0}^{T-1} (F(\bar{v}_t) - F(v_t))^2$$

where v_t is the predicted voltage trace, $\bar{v}_t$ is the target validation voltage trace, and $F(\cdot)$ is a nonlinear filter designed to extract a smoothed spike train out of the voltage traces. The nonlinear filter F applies, in succession: a bandpass 6th order Butterworth filter with cutoff frequencies at 1/50 and 1/2 kHz (which filters out most of the slow components of the voltage traces); a relu nonlinearity with threshold set at 2 mV (which transforms the filtered spike train into a sequence of approximate impulse functions); and a Gaussian smoother with standard deviation of 50 (HH neuron RMM), 200 (LP neuron RMM) and 500 mV (PD neuron RMM) which smooths the approximate impulse train. Notice comparing smoothed spike trains is an established methodology to validate statistical integrate-and-fire models (12).

Hogkin-Huxley RMM. In Figure 2, training data was generated by simulating the Hodgkin-Huxley model (26, 31) with the forward-euler discretization scheme and simulation timestep of $\delta_{\text{sim}} = 0.01$ ms. A diagonal state space realization was constructed with the time constants mentioned in Results; mechanistic readouts are given by Eq. (5) and Eq. (9), with a single output and 20 ANN units; Nernst potentials were fixed to those of the HH model. The training I_{app} consisted in three realizations of a stochastic process defined by

$$\begin{aligned} I_{\text{app},t} &= q_t + s_t + I_{\text{ramp},t} \\ q_{t+1} &= -\bar{\alpha} q_t + \bar{\sigma} w_t \\ s_t &= \{S_k, \quad 0.1k \leq \delta_{\text{sim}} t \leq 0.1(k+1) \end{aligned} \qquad [17]$$

where $I_{\text{ramp},t}$ is a deterministic ramp of applied current such that $I_{\text{ramp},0} = -15\mu A$ and $I_{\text{ramp},T} = 0\mu A$; q_t is the output

[3] It can be seen in Eq. (16) that when initial conditions are chosen according to Eq. (15), multiple shooting generalizes TF, which is recovered by choosing $N = T$ and $T_s = 1$.

DRAFT

of a discretized Ornstein-Uhlenbeck process as described above, with $\bar{\alpha} = 0.2$, and $\bar{\sigma} = 1$; and s_t is a random process given by a sequence of steps of random uniformly distributed amplitude $S_k \sim \mathcal{U}[0, 10]$. For learning, the resulting simulated data was downsampled to $\delta = 0.05$ ms, and the voltage was corrupted by white Gaussian noise of standard deviation equal to 0.5 mV. One realisation of the training dataset is illustrated in Figure S1. The validation dataset was obtained by stimulating the HH model with Eq. (17) containing only the Ornstein-Uhlenbeck component, with $\bar{\alpha} = 0.2$ and $\bar{\sigma} = 1$ ($I_{\text{ramp},t} = 0$ and $s_t = 0$). The Hodgkin-Huxley RMM was trained with TF and a regularization constant of $\rho_i = 10^{-6}$ for all weights in the model not belonging to $W_2^{\pm}$. The cost Eq. (12) was minimized with the ADAM routine with step size 0.001 and momentum parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$. Training curves can be found in the SI Appendix.

LP neuron RMM. To construct the LP neuron RMM, the intrinsic and synaptic currents were defined based on the sequence of continuous time constants $\tau_i^{\text{int}} = 0.05 \times 1.3^i$, $i = 0, 1, \ldots, 35$, and $\tau^{\text{syn}} = 0.05 \times 1.3^i$, $\quad i = 14, 15, \ldots, 29$, which were chosen pragmatically so as to ensure that all timescales usually found in LP model are represented; see Figure S4 for representative time constant functions found in an LP conductance-based model (42). A state space realization with orthogonal impulse responses was used. The model is constructed with two lumped readouts corresponding to the total intrinsic and total synaptic currents, with the corresponding MLPs given by Eq. (8). The MLPs were chosen with $L_{\text{int}} = 3$ layers and $L_{\text{syn}} = 2$ layers, respectively, with ten units in any given layer. In Figure 3, training and validation data for the LP neuron consisted in voltage $\bar{v}_t$ and applied current $I_{\text{app},t}$ obtained experimentally (see *electrophysiology* in Methods). The applied current used to train and validate the RMM was given by a discretized OU process, that is $I_{\text{app},t} = q_t$, with q_t as in Eq. (17), with $\bar{\alpha} = 0.015$ and $\bar{\sigma} = 0.015$. To train the model, we first used TF with $\rho_i = 0.0015$ for all weights in the model, and minimized the cost function with the ADAM gradient descent routine with step size 0.003, $\beta_1 = \beta_2 = 0.9$, and a mini-batch size of 484000 datapoints. In Figure 3 B we shifted the synaptic current prediction so that its minimum value is at zero. In Figure 3 we used multiple shooting with $T_s = 30$ and the same ρ_i, mini-batch size and and ADAM hyperparameters as TF; final states were regularized with $D = 0.1\,\text{diag}(1, D_{\text{int}}^2, D_{\text{syn}}^2)$ where D_{int} and D_{syn} are diagonal matrices whose diagonals are given by the inverse DC gains $[(I - A)^{-1}B]^{-1}$.

PD neuron RMM. To construct the PD neuron RMM, the intrinsic current matrices A and B were defined based on the sequence of continuous time constants $\tau_i^{\text{int}} = 0.5 \times 1.26^i$, $i = 0, 1, \ldots, 31$ (this results in logarithmically spread out time constants from 0.5 ms to around 643 ms). This was again chosen pragmatically based on the range of time constants from (42) (see Figure S4). The PD neuron RMM contained four currents: a lumped current I_{lumped}, given by Eq. (4), and three mechanistic data-driven currents I_{K}, I_{Ca}, and I_{H}, given by Eq. (11). To provide states for the current readouts, a block-diagonal A matrix was constructed using a mixed realization with $A = \text{diag}(A_d, A_o^{\text{lumped}}, A_o^{\text{K}+}, A_o^{\text{K}-})$, where A_d is a diagonal realization constructed with $\{\tau_i\}_{i=6}^{i=31}$, and A_o^{lumped}, $A_o^{\text{K}+}$, $A_o^{\text{K}-}$ are realizations with orthogonal impulse responses constructed with $\{\tau_i\}_{i=0}^{i=13}$, $\{\tau_i\}_{i=6}^{i=27}$, and $\{\tau_i\}_{i=20}^{i=27}$, respectively. The readouts of I_{Ca} and I_{H} read from the diagonal part of the state space system, while the readouts of I_{lumped} and I_{K} read from the part with orthogonal impulse responses. Each of the activation (σ_+) and inactivation (σ_-) mappings in the mechanistic currents contained two layers, with 20 and 10 units in each layer, respectively; the H current had no activation mapping. Conductance scaling parameters were initialized at $\bar{g}_{\text{K}} = 10.0$, $\bar{g}_{\text{Ca}} = 1.0$, $\bar{g}_{\text{H}} = 1.0$ and were left fixed throughout training. Nernst potentials of the mechanistic currents were initialized to $E_{\text{K}} = -80$, $E_{\text{K}} = 130 + \epsilon$, and $E_{\text{H}} = -10 + \epsilon$, with ϵ a vector of zero-mean normally distributed random variables with standard deviation of 10 mV. Nernst potential vectors of the I_{H} and I_{Ca} currents were allowed to be trained, while the Nernst potential of I_{K} was fixed. In the mechanistic currents, the normalization layers $h_{\pm}(\cdot)$ at the input of $\sigma_{\pm}(\cdot)$ were defined with biophysical priors for the voltage range given by $[v_{\min}^{h_+,K}, v_{\max}^{h_+,K}] = [v_{\min}^{h_+,Ca}, v_{\max}^{h_+,Ca}] = [-80.0, -20.0]$ mV, and $[v_{\min}^{h_-,K}, v_{\max}^{h_-,K}] = [v_{\min}^{h_-,Ca}, v_{\max}^{h_-,Ca}] = [v_{\min}^{h_-,H}, v_{\max}^{h_-,H}] = [-80.0, -30.0]$ mV (this choice was based on the activation functions of the model from (42)).

In Figure 4, training and validation data for the AB-PD subsystem consisted in voltage $\bar{v}_t$ and applied current $I_{\text{app},t}$ obtained experimentally from a single PD neuron following the experimental procotol described in the *electrophysiology* section. The dataset used to train and validate the RMM consisted of a combination of four different 42-second long trials where the PD neuron was excited with the discretized OU process Eq. (17); in two of the trials $\bar{\sigma} = 0.015$ was fixed and $\bar{\alpha} = 0.001, 0.005$ was varied, and in two of the trials $\bar{\sigma} = 0.03$ was fixed and $\bar{\alpha} = 0.01, 0.015$ was varied. To train the model, we used TF with $\rho_i = 10^{-5}$ for all weights belonging to hidden ANN layers; for weights in $\tilde{W}_2^{\pm}$ we used $\rho_i = 0$, while for weights in W_2^{lumped} we used $\rho_i = 0.01$. The cost was minimized with the ADAM gradient descent routine with step size 0.001, momentum parameters of $\beta_1 = 0.9$, $\beta_2 = 0.99$, and a mini-batch size of 484000 datapoints.

Electrophysiology.

Animals. Adult male Jonah Crabs, *Cancer borealis*, were acquired from Commercial Lobster (Boston, MA) between August 2022 and July 2023. Animals were held in groups up to 5 in 75 Gallon tanks filled with artificial seawater at 11-13°C with a 12 hr light/dark cycle. The crabs were put on ice for at least 30 min before the dissection. Dissections were performed as described in (50). In brief, the crab stomach was removed, the stomatogastric nervous system (STNS) was dissected from the stomach and pinned down in a petri dish (10 ml) lined with Sylgard (The Dow Chemical Company). The STNS includes the stomatogastric ganglion (STG) containing the cells of interest, modulatory inputs descending into the STG from the esophageal ganglion and the two bilateral commissural ganglia, and the motor nerves coming from STG. The STNS was held at 4°C overnight in solution (saline) containing 440 mM NaCl, 11 mM KCl, 26 mM $MgCl_2$, 13 mM $CaCl_2$, 11.2 mM Trizma base, 5.1 mM maleic acid, pH 7.45, and used for the experiment next day.

Recordings. The STG was desheathed and intracellular recordings from somata of the lateral pyloric (LP) and pyloric dilator

DRAFT

(PD) neurons were obtained. Sharp glass microelectrodes (10-20 MΩ) filled with 0.6 M K_2SO_4 and 20 mM KCl were used to record the intracellular activity. During the experiment the STNS-containing dish was pump-perfused with 11°C saline at a rate of 5ml/min. The temperature was controlled by Peltier device connected to the model CL-100 temperature controller (Warner Instruments). The cells were identified by their activity (monitored by extracellular electrodes) on the motor nerves: PD activity on the pyloric dilator nerve and LP activity on the lateral ventricular nerve. Extracellular recordings were done with a stainless-steel electrode in a well (Vaseline and 10% mineral oil mixture) around the motor nerve of interest. An Axoclamp 900A amplifier (Molecular Devices, San Jose) was used to amplify intracellular signals. Extracellular signals were amplified by an A-M Systems 1700 extracellular amplifiers. Data were obtained with pClamp data acquisition software (Molecular Devices, San Jose, version 10.5), Digidata 1440A digitizer (Molecular Devices, San Jose) and Real-Time eXperiment Interface (RTXI) software version 2.2. Recordings were acquired at 10 kHz sampling frequency. A custom RTXI module was used for the Ornstein-Uhlenbeck noise current injection. After the cells were identified, the solution was switched to saline containing 10^{-5}M picrotoxin (PTX), Sigma-Aldrich, to block the inhibitory synaptic connections between cells. Recording started 30 min after PTX was applied. The membrane voltage was recorded (in PD and LP) and current was injected (in LP or in PD, depending on the experiment) with a single electrode in Discontinuous Current Clamp (DCC) mode. To filter out the inherent noise of DCC recordings, PD neuron data was filtered with a 4th order Butterworth filter with cutoff frequency given $f_c = 4$ kHz.

Hardware. Training times in this paper were obtained with a desktop computer set up with an AMD Ryzen 7 7800X3D 8-Core Processor, 64 Gigabites of RAM, and a NVIDIA GeForce RTX 4090 graphics card with 24 GB of VRAM.

Acknowledgements. The research leading to these results has been funded by the Kavli Foundation and the National Institutes of Health (R35 NS 097343). Their support is greatly appreciated. The authors thank Rodolphe Sepulchre, Fulvio Forni, Dhruva Raman, Alessio Franci and Kyra Schapiro for their insightful comments on this work.

1. L Paninski, J Pillow, E Simoncelli, Comparing integrate-and-fire models estimated using intracellular and extracellular data. *Neurocomputing* **65**, 379–385 (2005).
2. H Abarbanel, D Creveling, R Farsian, M Kostuk, Dynamical State and Parameter Estimation. *SIAM J. on Appl. Dyn. Syst.* **8**, 1341–1381 (2009).
3. CD Meliza, et al., Estimating parameters and predicting membrane voltages with conductance-based neuron models. *Biol. Cybern.* **108**, 495–516 (2014).
4. JM Lueckmann, et al., Flexible statistical inference for mechanistic models of neural dynamics in *31st Conference on Neural Information Processing Systems (NIPS 2017)*. (Long Beach, CA, USA), p. 18 (2017).
5. K Abu-Hassan, et al., Optimal solid state neurons. *Nat. Commun.* **10**, 5309 (2019).
6. LM Alonso, E Marder, Visualization of currents in neural models with similar behavior and different conductance densities. *eLife* **8**, e42722 (2019).
7. JD Taylor, S Winnall, A Nogaret, Estimation of neuron parameters from imperfect observations. *PLOS Comput. Biol.* **16**, e1008053 (2020).
8. PJ Gonçalves, et al., Training deep neural density estimators to identify mechanistic models of neural dynamics. *eLife* **9**, e56261 (2020).
9. TB Burghi, R Sepulchre, Adaptive observers for biophysical neuronal circuits. *IEEE Transactions on Autom. Control.* **69**, 5020–5033 (2024).
10. SA Wells, JD Taylor, PG Morris, A Nogaret, Inferring the Dynamics of Ionic Currents from Recursive Piecewise Data Assimilation of Approximate Neuron Models. *PRX Life* **2**, 023007 (2024).
11. J Pillow, M Park, Adaptive Bayesian Methods for Closed-Loop Neurophysiology in *Closed Loop Neuroscience*. (Academic Press, Cambridge, MA), pp. 3–18 (2016).
12. W Gerstner, WM Kistler, R Naud, L Paninski, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. (Cambridge University Press, Cambridge, UK), (2014).
13. TB Burghi, M Schoukens, R Sepulchre, System identification of biophysical neuronal models in *59th IEEE Conference on Decision and Control*. (Jeju Island, South Korea), pp. 6180–6185 (2020).
14. D Beniaguev, I Segev, M London, Single cortical neurons as deep artificial neural networks. *Neuron* **109**, 2727–2739.e3 (2021).
15. KW Latimer, F Rieke, JW Pillow, Inferring synaptic inputs from spikes with a conductance-based neural encoding model. *eLife* **8**, e47012 (2019).
16. D Durstewitz, G Koppe, MI Thurm, Reconstructing computational system dynamics from neural data with recurrent neural networks. *Nat. Rev. Neurosci.* **24**, 693–710 (2023).
17. M Aguiar, A Das, KH Johansson, Learning flow functions of spiking systems in *Proceedings of the 6th Annual Learning for Dynamics & Control Conference*. (PMLR), pp. 591–602 (2024).
18. M Brenner, et al., Tractable Dendritic RNNs for Reconstructing Nonlinear Dynamical Systems in *Proceedings of the 39th International Conference on Machine Learning*. (PMLR), pp. 2292–2320 (2022).
19. R Naud, W Gerstner, Can we predict every spike? in *Spike Timing: Mechanisms and Function*. (CRC Press), (2013).
20. QJM Huys, MB Ahrens, L Paninski, Efficient Estimation of Detailed Single-Neuron Models. *J. Neurophysiol.* **96**, 872–890 (2006).
21. AA Prinz, D Bucher, E Marder, Similar network activity from disparate circuit parameters. *Nat. Neurosci.* **7**, 1345–1352 (2004).
22. K Doya, Bifurcations in the learning of recurrent neural networks in *[Proceedings] 1992 IEEE International Symposium on Circuits and Systems*. Vol. 6, pp. 2777–2780 vol.6 (1992).
23. AH Ribeiro, K Tiels, J Umenberger, TB Schön, LA Aguirre, On the smoothness of nonlinear system identification. *Automatica* **121**, 109158 (2020).
24. R Pascanu, T Mikolov, Y Bengio, On the difficulty of training recurrent neural networks in *Proceedings of the 30th International Conference on Machine Learning*, ICML'13. (Atlanta, GA, USA), Vol. 28, pp. 1310–1318 (2013).
25. J Keener, J Sneyd, *Mathematical Physiology*. (Springer, New York, NY) Vol. 8/1, 2 edition, (2009).
26. AL Hodgkin, AF Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve. *The J. Physiol.* **117**, 500–544 (1952).
27. S Boyd, L Chua, Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits Syst.* **32**, 1150–1161 (1985).
28. IW Sandberg, Uniform approximation and the circle criterion. *IEEE Transactions on Autom. Control.* **38**, 1450–1458 (1993).
29. J Golowasch, E Marder, Ionic currents of the lateral pyloric neuron of the stomatogastric ganglion of the crab. *J. Neurophysiol.* **67**, 318–331 (1992).
30. AL Taylor, JM Goaillard, E Marder, How Multiple Conductances Determine Electrophysiological Properties in a Multicompartment Model. *J. Neurosci.* **29**, 5573–5586 (2009).
31. EM Izhikevich, *Dynamical Systems in Neuroscience*. (MIT Press, Cambridge, MA), (2007).
32. GB Ermentrout, DH Terman, *Mathematical Foundations of Neuroscience*. (Springer, New York), (2010).
33. G Drion, A Franci, J Dethier, R Sepulchre, Dynamic Input Conductances Shape Neuronal Spiking. *eneuro* **2**, ENEURO.0031–14.2015 (2015).
34. WV Geit, ED Schutter, P Achard, Automated neuron model optimization techniques: A review. *Biol. Cybern.* **99**, 241–251 (2008).
35. M Almog, A Korngreen, Is realistic neuronal modeling realistic? *J. Neurophysiol.* **116**, 2180–2209 (2016).
36. DA McCormick, Y Shu, Y Yu, Hodgkin and Huxley model — still standing? *Nature* **445**, E1–E2 (2007).
37. E Marder, D Bucher, Understanding Circuit Dynamics Using the Stomatogastric Nervous System of Lobsters and Crabs. *Annu. Rev. Physiol.* **69**, 291–316 (2007).
38. DM Blitz, MP Nusbaum, State-Dependent Presynaptic Inhibition Regulates Central Pattern Generator Feedback to Descending Inputs. *J. Neurosci.* **28**, 9564–9574 (2008).
39. E Marder, GJ Gutierrez, MP Nusbaum, Complicating connectomes: Electrical coupling creates parallel pathways and degenerate circuit mechanisms. *Dev. Neurobiol.* **77**, 597–609 (2017).
40. B Johnson, J Peck, R Harris-Warrick, Differential modulation of chemical and electrical components of mixed synapses in the lobster stomatogastric ganglion. *J. Comp. Physiol. A* **175**, 233–249 (1994).
41. M Innes, Flux: Elegant machine learning with julia. *J. Open Source Softw.* (2018).
42. Z Liu, J Golowasch, E Marder, LF Abbott, A Model Neuron with Activity-Dependent Conductances Regulated by Multiple Calcium Sensors. *J. Neurosci.* **18**, 2309–2320 (1998).
43. KJ Aström, RM Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. (Princeton University Press, Princeton), (2008).
44. PSC Heuberger, PMJ van den Hof, B Wahlberg, *Modelling and Identification with Rational Orthogonal Basis Functions*. (Springer, London), (2005).
45. D Nguyen, B Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights in *1990 IJCNN International Joint Conference on Neural Networks*. pp. 21–26 vol.3 (1990).
46. L Farina, S Rinaldi, *Positive Linear Systems: Theory and Applications*. (John Wiley & Sons), (2000).
47. HDI Abarbanel, DR Creveling, JM Jeanne, Estimation of parameters in nonlinear systems using balanced synchronization. *Phys. Rev. E* **77** (2008).
48. AH Ribeiro, K Tiels, LA Aguirre, T Schön, Beyond exploding and vanishing gradients: Analysing RNN training using attractors and smoothness in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. (PMLR), pp. 2370–2380 (2020).
49. L Ljung, *System Identification: Theory for the User*. (Prentice Hall PTR, Upper Saddle River, NJ), (1999).
50. GJ Gutierrez, RG Grashow, Cancer Borealis Stomatogastric Nervous System Dissection. *J. Vis. Exp. : JoVE* p. 1207 (2009).

DRAFT

Supporting Information for

Rapid, interpretable, data-driven models of neural dynamics using Recurrent Mechanistic Models

Thiago B. Burghi*, Maria Ivanova, Ekaterina Morozova, Huaxinyu Wang, Eve Marder and Timothy O'Leary

***Corresponding author.**
E-mail: tbb29@cam.ac.uk

This PDF file includes:

Supporting text
Figs. S1 to S6
SI References

Supporting Information Text

Methods

Weight initialization. In lumped and mechanistic currents, the purpose of weight initialization is to obtain a random set of ANN weights and biases which is empirically optimized for the ANN to learn efficiently from inputs that lie mostly on the interval $[-1, +1]$. Initialization for lumped and mechanistic currents differ due to the weight constraints mentioned in the main text.

Lumped currents. The W matrices in each layer of the lumped readouts are initialized with the scaled Nguyen-Widrow heuristic (1). Letting $W[k]$ denote the weight matrix at iteration k of the learning algorithm, at each layer ℓ we set

$$[W_\ell]_{ij}[0] = 2 \times 0.7 \times (n_\ell)^{\frac{1}{n_\ell - 1}} \frac{X_{ij}}{\sqrt{\sum_i X_{ij}^2}}, \qquad X_{ij} \sim \mathcal{U}[-1, 1]$$

where n_ℓ is the number of units in the layer ℓ (the number of inputs of the MLP when $\ell = 0$). The $2\times$ factor is included in the heuristic to optimize the weights for use with tanh, whose effective activation range is the interval $[-2, 2]$. The biases b in each layer are spread evenly according to

$$[b_\ell]_i[0] = 2 \times \left(-1 + \frac{2(i-1)}{(n_\ell - 1)}\right), \qquad i = 1, 2, \ldots, n_\ell$$

if $n_\ell > 1$, and $[b_\ell]_1 = 0$ if $n_\ell = 1$.

Mechanistic currents. The weight initialization heuristic changes in mechanistic currents due to the elementwise positivity constraints on the weight matrices. The weight matrices $W_1^\pm[k]$ and $W_2^\pm[k]$ at iteration k of the learning algorithm are given by

$$[W_1^\pm]_{ij}[k] = \pm\varsigma([\tilde{W}_1^\pm]_{ij}[k]), \qquad [W_2^\pm]_{ij}[k] = \varsigma([\tilde{W}_2^\pm]_{ij}[k]),$$

where $\varsigma(\cdot) = \log(1 + \exp(\cdot))$ is the softplus function. The matrices $\tilde{W}_1^\pm$ and $\tilde{W}_2^\pm$ contain the parameters being learned.

We initialize $\tilde{W}_1^\pm$ according to

$$[\tilde{W}_1^\pm]_{ij}[0] = \varsigma^{-1}\left(\frac{16}{n_0}(0.5 + 0.25\, Y_{ij})\right), \qquad Y_{ij} \sim \mathcal{U}[-1, 1]$$

where n_0 is the number of outputs of the mechanistic normalization layer (inputs of σ_+), and $\varsigma^{-1}(\cdot) = \cdot + \log(1 - \exp(-\cdot)$ is the inverse of the softplus function. The corresponding biases are given by

$$[b_1^\pm]_i[0] = 4 \times \left(-1 + \frac{2(i-1)}{(n_0 - 1)}\right), \quad i = 1, \ldots, n_0$$

The initialization heuristic above ensures that

$$\begin{aligned} u = \mathbf{1} \implies & \mathbb{E}\left\{W_1^+[0]u + b_1^+[0]\right\} \\ & = 8 \times \mathbf{1} + b_1^+[0] \\ & \geq 4 \times \mathbf{1} \quad \text{(elementwise)} \end{aligned}$$

which implies that at initialization, $\mathbb{E}\{\sigma(W_1^+[0]u + b_1^+[0])\} \approx \mathbf{1}$ whenever $u = \mathbf{1}$, since the effective activation range of the logistic nonlinearity $\sigma(\cdot)$ is the interval $[-4, 4]$. Similarly, we have

$$u = -\mathbf{1} \implies \mathbb{E}\left\{W_1^+[0]u + b_1^+[0]\right\} \leq -4 \times \mathbf{1} \quad \text{(elementwise)}$$

and thus $\mathbb{E}\{\sigma(W_1^+[0]u + b_1^+[0])\} \approx \mathbf{0}$ whenever $u = -\mathbf{1}$. A completely analogous consideration applies to W_1^-.

We initialize $\tilde{W}_2^\pm$ according to

$$[\tilde{W}_2^\pm]_{ij}[0] = \varsigma^{-1}\left(\frac{0.5 + 0.25 Z_{ij}}{\sum_j (0.5 + 0.25 Z_{ij})}\right), \quad Z_{ij} \sim \mathcal{U}[-1, 1]$$

which implies that at initialization, $\mathbb{E}\{\sigma_+(u)\} \approx \mathbf{1}$ whenever $u = \mathbf{1}$ and $\mathbb{E}\{\sigma_+(u)\} \approx \mathbf{0}$ whenever $u = -\mathbf{1}$. An analogous reasoning applies to W_2^- and $\sigma_-(\cdot)$.

The above initialization heuristic guarantees that mechanistic currents are initially shut down for steady-states corresponding to voltages at the boundaries of a prior activation range; see *Normalization layers* below.

Normalization layers. The normalization layers of lumped currents and mechanistic currents differ in the interval of readout inputs which is mapped to $[-1, +1]$.

Lumped currents. In lumped currents, the fixed mappings $h(v_t, x_t)$ are given by

$$h(v_t, x_t) = \begin{bmatrix} C_v v_t + \beta_v \\ C_x x_t + \beta_x \end{bmatrix}$$

where C_v and β_v are used to perform min-max normalization of v_t, and C_x and β_x are used to select a subset of states of x_t and perform min-max normalization on those states. Min-max normalization is accomplished as follows: given the data $\bar{v}_t$ used for training, we first obtain a state trajectory $\bar{x}_t$ by simulating Eq. 3b (main text) with the voltage data $\bar{v}_t$ (transients not used during training are discarded). Then the C matrices and β shift vectors are calculated to ensure that the interval $[\min_t \bar{v}_t, \max_t \bar{v}_t]$, as well as each of the intervals $[\min_t \bar{x}_{i,t}, \max_t \bar{x}_{i,t}]$, are mapped onto $[-1, 1]$. Mathematically, we have

$$C_v = \frac{2}{(\max_t \bar{v}_t) - (\min_t \bar{v}_t)}, \qquad \beta_v = -1 - 2\frac{(\min_t \bar{v}_t)}{(\max_t \bar{v}_t) - (\min_t \bar{v}_t)}$$

for voltage, and

$$C_x = \begin{bmatrix} 0 & \dots & 0 & \begin{matrix} \dots \\ C_{x_i} \\ \dots \end{matrix} & 0 & \dots & 0 \end{bmatrix}, \qquad \beta_x = \begin{bmatrix} \dots & \beta_{x_i} & \dots \end{bmatrix}^\intercal$$

with C_{x_i} and β_{x_i} computed analogously to C_v and β_v above (here, i is the index of a state selected to be an input to the MLP).

Mechanistic currents. In mechanistic currents, for each activation or inactivation readout, $\sigma_+(h_+(x_t))$ and $\sigma_-(h_-(x_t))$ respectively, the fixed mapping

$$h(x_t) = C_x x_t + \beta_x$$

selects a subset of states of x_t and normalizes them. However, instead of min-max normalization, we employ normalization based on a prior voltage interval over which we initially believe a current activates or inactivates. Given such an interval $[v_{\text{low}}, v_{\text{high}}]$, we first compute the steady-states

$$x_{\text{low}} = (I - A)^{-1} B\, v_{\text{low}}, \qquad x_{\text{high}} = (I - A)^{-1} B\, v_{\text{high}}$$

at the endpoints of $[v_{\text{low}}, v_{\text{high}}]$. Then, C_x and β_x are found by mapping each of the intervals $[x_{i,\text{low}}, x_{i,\text{high}}]$ onto $[-1, 1]$, where i is the index of a state selected to be an input to $\sigma(\cdot)$. Mathematically, the expressions for C_x and β_x for mechanistic currents are analogous to those of lumped currents.

Notice that because we employ state space realisations with elementwise positive steady-state gain $(I - A)^{-1}B$, the C matrices found in this way are also elementwise positive.

Monotonicity of mechanistic currents with diagonal state space realisations. Consider the function $\sigma_+(h_+(x))$ modelling the activation of a mechanistic readout which reads from a diagonal state space realisation with $A = \text{diag}\{\lambda_j\}$ and $B = (\mathbf{1} - \text{vec}(\lambda_j))$. The input-output system realised by $(A, B, W_1^+ C_x)$ is said to be *positive* (2), because we are dealing with a discrete time system whose state space matrices are all elementwise positive. As a consequence, monotonically increasing/decreasing input (voltage) trajectories yield monotonically increasing/decreasing trajectories at the input of the logistic function nonlinearities (3). The elementwise positivity of $W_2^\pm$ coupled with the monotonicity of the logistic nonlinearities in $\sigma_+(h_+(x))$ then ensures that monotonicity of those trajectories is preserved at the output of the readout. The argument is analogous for $\sigma_-(h_-(x))$, with the difference that since W_1^- is elementwise negative, the realisation $(A, B, W_1^- C_x)$ is a *negative system*. In that case monotonically increasing/decreasing voltage trajectories yield monotonically decreasing/increasing trajectories at the input of the logistic nonlinearities; this is again preserved by the monotonicity of the logistic function and elementwise positivity of W_2^-.

State space system with orthogonal impulse responses. The discrete-time poles λ_j are can be used to A and B so as to realize a linear system with orthogonal impulse responses (also known as convolution kernels), pointed out in the main text. To construct a system with no timescale separation between states, this is accomplished by defining $A = A_n$ and $B = B_n$, where A_n and B_n are given by the iterative procedure

$$\begin{aligned} &A_1 = \lambda_1, \quad B_1 = \sqrt{1 - \lambda_1^2}, \quad C_1 = B_1, \quad D_1 = -\lambda_1 \\ &A_i = \begin{bmatrix} A_{i-1} & 0 \\ \sqrt{1 - \lambda_i^2} C_{i-1} & \lambda_i \end{bmatrix}, \quad B_i = \begin{bmatrix} B_{i-1} \\ \sqrt{1 - \lambda_i^2} D_{i-1} \end{bmatrix} \\ &C_i = \begin{bmatrix} -\lambda_i C_{i-1} & \sqrt{1 - \lambda_i^2} \end{bmatrix}, \quad D_i = -\lambda_i D_{i-1} \end{aligned} \tag{1}$$

for $i = 1, \dots, n$ (see (4, Chapter 2) for details). The above procedure results in a system whose state components (the elements of x_t) follow distinctly different trajectories given sudden changes in the input; we have found empirically that this results in significantly better predictive models when compared to simpler choices of A and B. Figure S4 of the SI Appendix shows how the responses of a system with orthogonal impulse responses overlap less in time, providing ANNs with a richer set of input signals. Notice that for slow state components, a spike is well approximated by the impulse function, and hence spike responses of slow states will be near orthogonal (modulo a constant dependent on the mean of the spiking signal).

Whenever timescale separation between components of the state vector x_t is desired, the procedure is slightly modified. To construct A and B in those cases, we first partition the time constant sequence $\{\tau_j\}_{j=1,\ldots,n}$ into non-intersecting subsequences $\{\tau_j\}_{j=1,\ldots,m_1}$, ..., $\{\tau_j\}_{j=m_s,\ldots,n}$ where $s > 0$ is the number of desired timescale partitions. Then, we define A as a block-diagonal matrix whose diagonal (matrix) elements are each constructed according to Eq. (1) using the corresponding subsequences of discrete-time poles $\{\lambda_j\}_{j=1,\ldots,m_1}$, etc. This modification ensures that the state vector partitions $[x_1, \ldots, x_{m_1}]$, etc., do not share dynamical modes in the same timescales.

Parallelization with Teacher forcing. Teacher forcing lends itself exceptionally well for the purposes of training RMMs, and in case the state space matrices A and B remain fixed (non-trainable), massively parallelized computations in GPUs can be exploited to their fullest potential. To see why, assume A and B are fixed a priori, and let

$$\bar{v}_{0:T} = \begin{bmatrix} \bar{v}_0 & \bar{v}_1 & \ldots & \bar{v}_T \end{bmatrix}$$

denote a row vector containing all voltage measurements. Teacher-forced state estimates can be obtained from

$$\hat{x}_{t+1} = A\hat{x}_t + B\bar{v}_t \qquad [2]$$

which need to be computed only once as A and B will not be trained. The state estimates can also be put in matrix form as

$$\hat{x}_{0:T} = \begin{bmatrix} \hat{x}_0 & \hat{x}_1 & \ldots & \hat{x}_T \end{bmatrix}$$

with $\hat{x}_0$ a "warmed-up" initial condition*. By precomputing $\hat{x}_{0:T}$, minimizing the teacher forcing cost function (Eq. 12 in main text) in θ reduces to the problem of training a feed-forward ANN with inputs and targets given by large matrices, enabling massive parallelization. Indeed, Eq. 12 (main text), and Eq. 13 (main text) can be rewritten as

$$\mathcal{C}(\theta, T) = \frac{1}{T}\|\Delta\bar{v}_{0:T-1} - \Delta\hat{v}_{0:T-1}(\theta)\|^2 + \sum_i \rho_i(\theta_i - \bar{\theta}_i)^2 \qquad [3]$$

with $\Delta\bar{v}_{0:T-1}$ a row vector of target ANN outputs, and $\Delta\hat{v}_{0:T-1}$ being the output of a feed-forward ANN with matrix inputs $\bar{v}_{0:T-1}$, $\hat{x}_{0:T-1}$, $\hat{z}^p_{0:T-1}$, and $I_{\text{app},0:T-1}$. Notice that in practice, the capacitance c is lumped into the output layers of the φ sub-networks, and recovered as the inverse of the ANN input weight corresponding to I_{app}.

Computing the gradient of Eq. (3) can be done efficiently with backpropagation, which for moderate number of layers in the model's MLPs causes no exploding nor vanishing gradients†. Notice that this "static" version of teacher forcing relies on two facts: first, the fact that bounds on physiological time constants are well-known, which allow fixing A and B in a reasonable manner; and second, the fact that A can be trivially constrained to be stable (Hurwitz), a property that ensures that the states $\hat{x}_t$ and $\hat{s}^p_t$ remain bounded for any possible measurements $\bar{v}_t$ and $\bar{v}^p_t$.

Parallelization with Multiple shooting. Letting

$$\begin{aligned} V_t^{(n)} &= \begin{bmatrix} v_{t+nT_s} & v_{t+2nT_s} & \ldots & v_{t+(N-1)T_s} \end{bmatrix}, \\ X_t^{(n)} &= \begin{bmatrix} x_{t+nT_s} & x_{t+2nT_s} & \ldots & x_{t+(N-1)T_s} \end{bmatrix} \end{aligned}$$

and similarly defining S_t and S^p_t, the model can be simulated in parallel on each of the N multiple shooting intervals by simulating Eqs.1-6 (main text) blockwise, that is, replacing v_t by $V_t^{(n)}$, x_t by $X_t^{(n)}$, and so on. It follows that in multiple shooting there is a natural tradeoff between the speed of training (enabled by large N and massive parallelization) and the quality of predictions, which tends to increase (up to a bound determined by exploding gradients) with longer intervals T_s (see (5), (6) for a formal analysis).

*Warming up consists in choosing $\hat{x}_0$ by simulating the state space model with a reserved part of the voltage data: suppose the initial segment of the dataset is given by $\{\bar{v}_t\}_{t\in[-N,-1]}$; then this simply means computing Eq. (2) for $t = -N, \ldots, -1$.

†In practice, additional tricks used to train feed-forward neural networks with gradient descent can be employed to improve generalization of the trained model. Two of those tricks are *mini-batching*, which divides the dataset into multiple batches so as to perform more frequent descent steps, and *shuffling*, which randomly permutes the columns of input and output matrices at the beginning of each training epoch. The joint application of such tricks has come to be known as *stochastic gradient descent*.

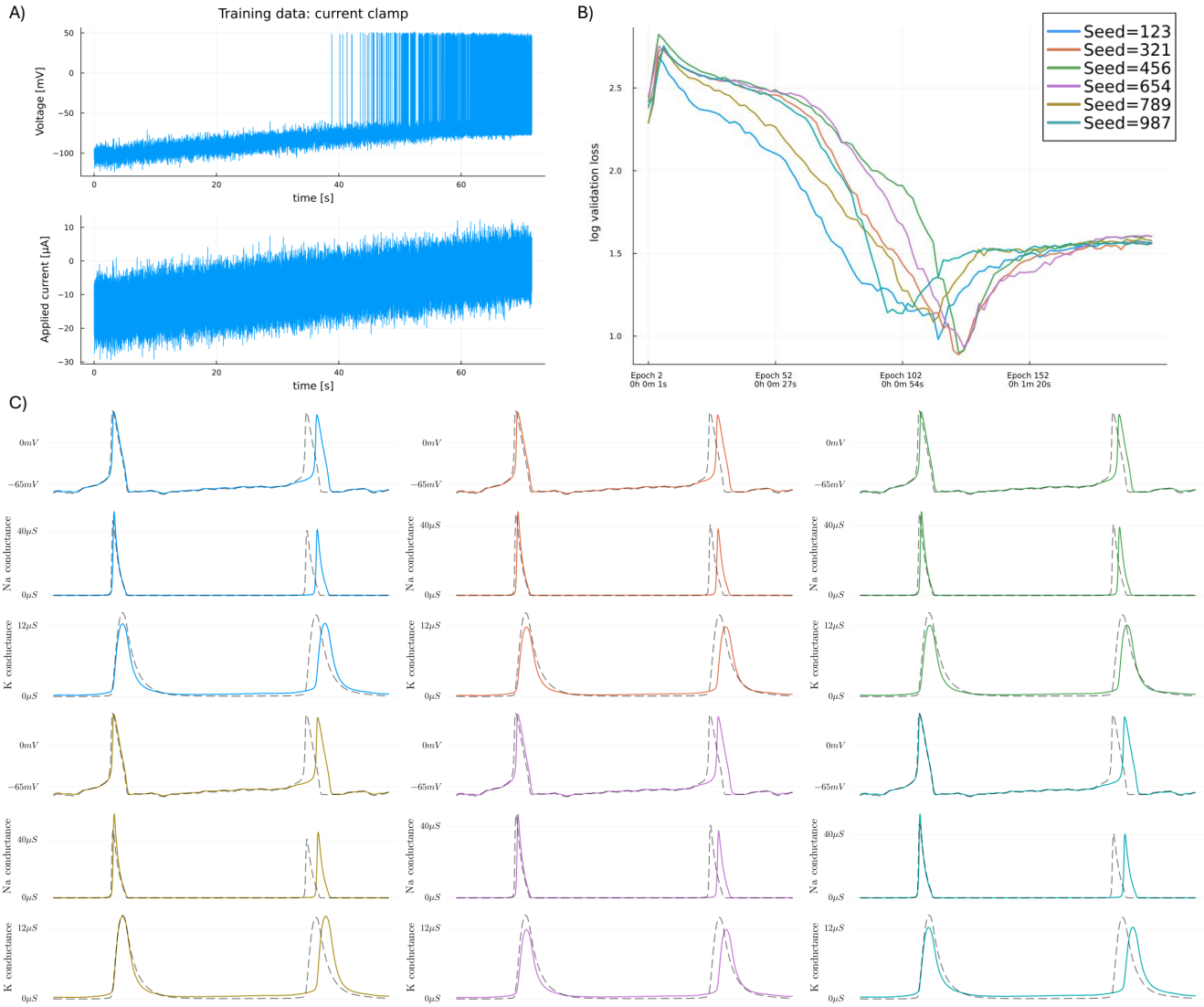

Fig. S1. A) One realization of the input-output data used to train the Hodgkin-Huxley RMM. Three realizations of the same stochastic applied current were used to train the model. B) Validation loss of six Hodgkin-Huxley RMMs trained with the data of (A). Training used the same hyperparameters, but different random number generator (RNG) seeds. RNG seeds were used to randomize parameter initialization, as well as to randomize the shuffling algorithm of stochastic gradient descent. C) Comparison of the predictive accuracy of the voltage and conductance traces of all six models whose validation curves are plotted in (B). Models were selected at the minimum of the validation curves.

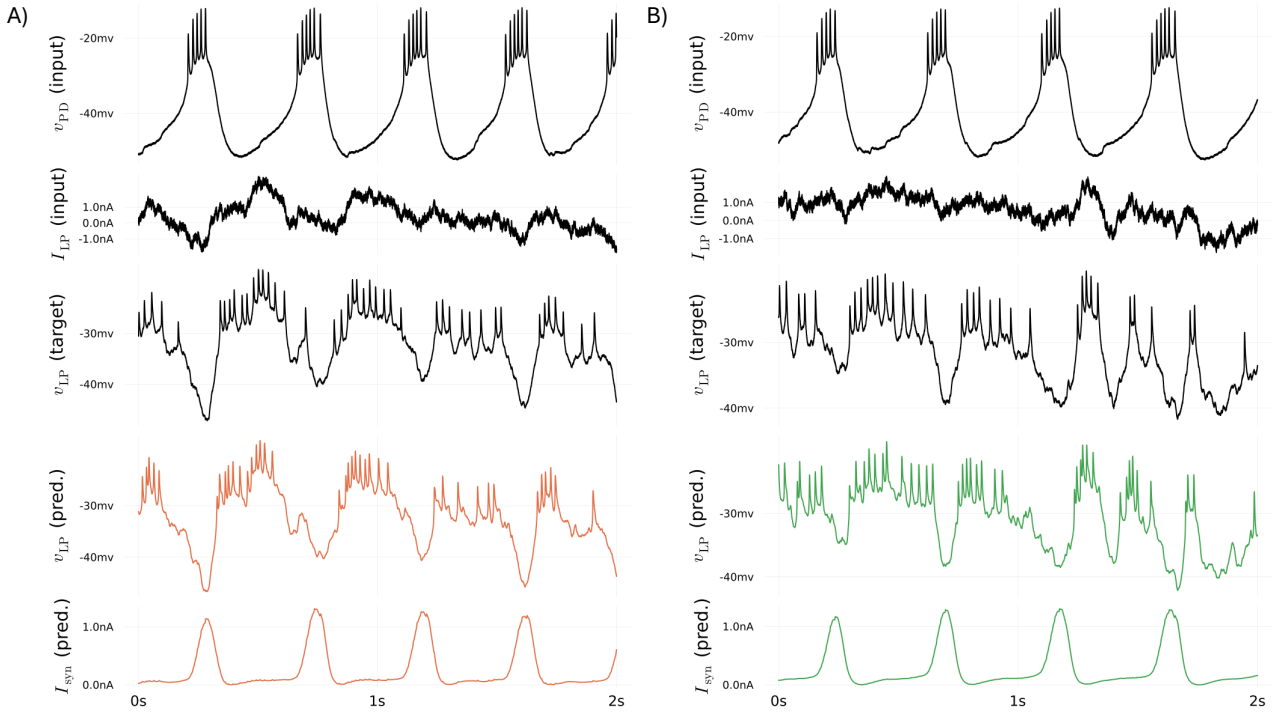

Fig. S2. Prediction results for the two LP models whose training curves are presented in the main body of the text.

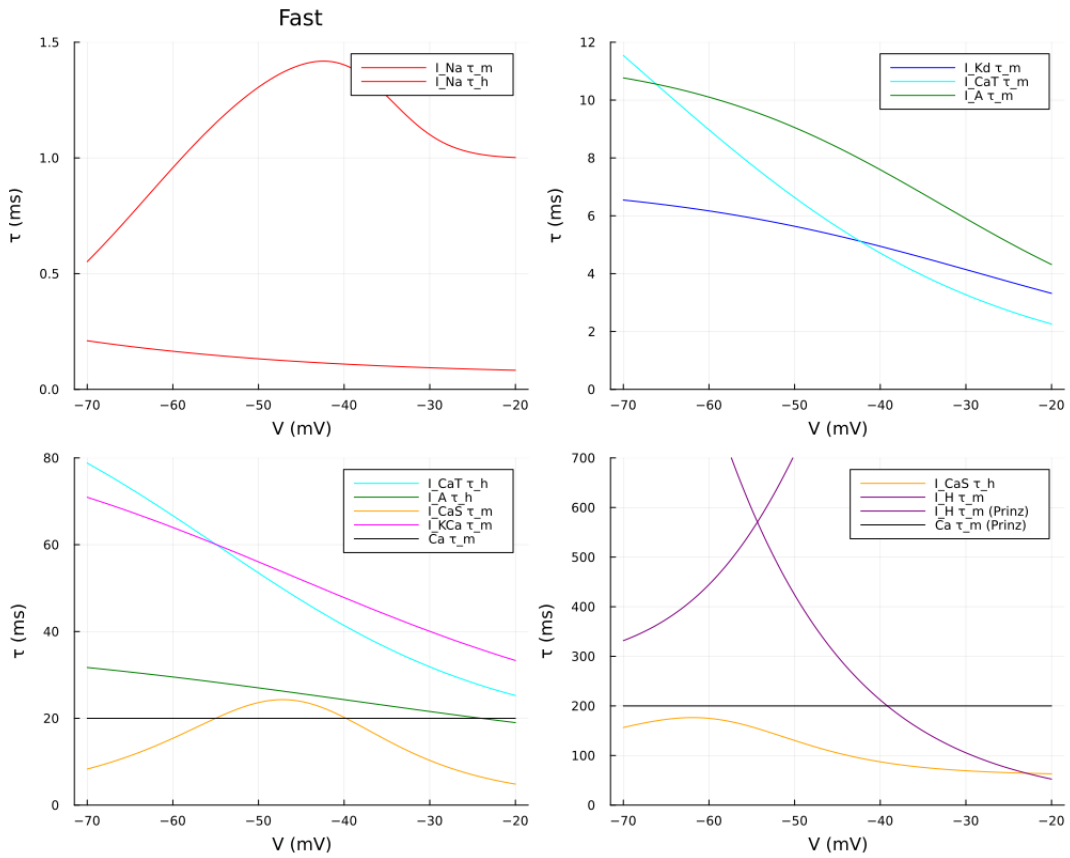

Fig. S3. Time constant functions of the conductance-based model of (7). These functions are also used in (8), who considered different time constants for calcium concentration and the H current, and doubled the values of all time constant functions plotted above.

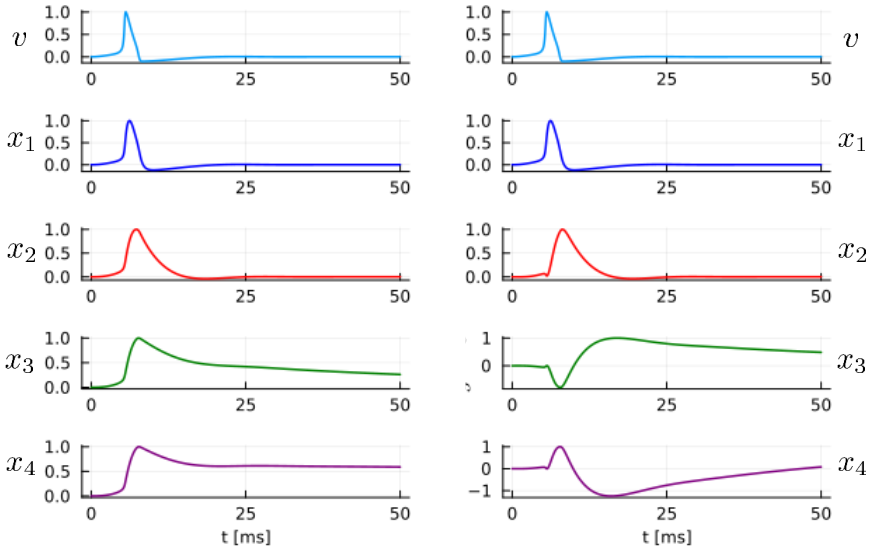

Fig. S4. Spike responses of two different choices of state space systems. **Left:** state component responses $x_{i,t}$ of a four-dimensional diagonal state space system given by $A = \mathrm{diag}(\lambda_i)$, $B = ((1-\lambda_1), \ldots, (1-\lambda_n))^\intercal$). Here, $\lambda_i = \exp(-\delta/\tau_i)$ is computed using a sampling period of $\delta = 0.01$ and continuous time constants $\tau_1 = 0.5$, $\tau_2 = 5$, $\tau_3 = 50$, and $\tau_4 = 500$. Spike responses, which are shown normalized by maximum value, are the output of the system to the spike shown on top. **Right:** spike responses of orthogonal filters, constructed with the same sequence of λ_i according to the procedure described in Methods. In the immediate aftermath of a spike, orthogonal filters provide richer signal features for artificial neural networks to be trained and recombine into ionic currents.

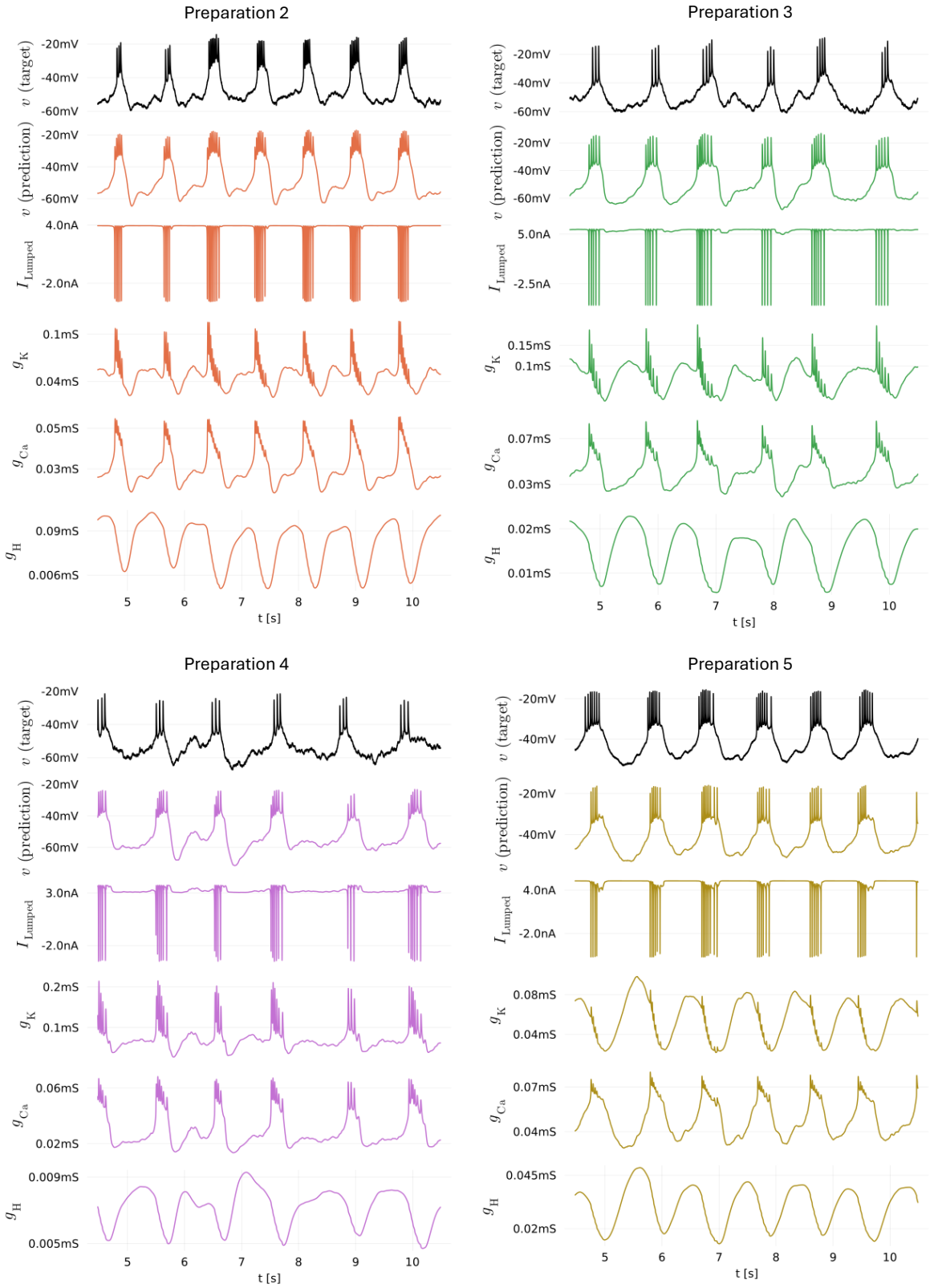

Fig. S5. Prediction results for four different PD model preparations in addition to the preparation whose predictions are shown in Figure 3 of the main text. The RMMs were trained on these preparations using the same hyperparameters used to obtain the results of Figure 3, except for the overall regularization constant ($\rho_{\text{prep2}} = 7.5 \times 10^{-5}$, $\rho_{\text{prep3}} = 5 \times 10^{-5}$, $\rho_{\text{prep4}} = 5 \times 10^{-5}$,$\rho_{\text{prep5}} = 2 \times 10^{-5}$). While electrophysiological properties of the PD neurons are visibly different, the RMM is able to capture their dynamics reasonably well. All models whose predictions are shown in this figure have been trained in around 2 minutes (see Hardware in Methods, in the main text).

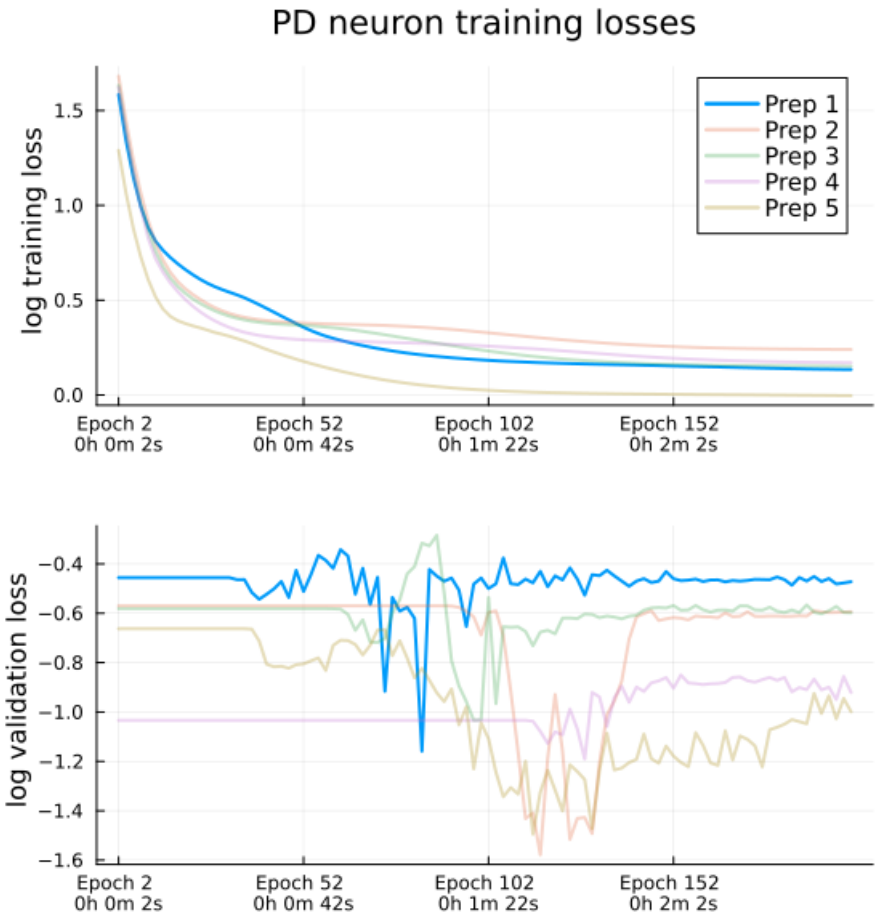

Fig. S6. Training and validation losses of the five PD neuron RMMs trained on experimental data. Blue traces refer to the RMM shown in Figure 3 of the main text, while other traces refer to RMMs shown in Figure S5.

References

1. D Nguyen, B Widrow, Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights in *1990 IJCNN International Joint Conference on Neural Networks*. pp. 21–26 vol.3 (1990).
2. L Farina, S Rinaldi, *Positive Linear Systems: Theory and Applications*. (John Wiley & Sons), (2000).
3. D Angeli, ED Sontag, Monotone control systems. *IEEE Transactions on Autom. Control*. **48**, 1684–1698 (2003).
4. PSC Heuberger, PMJ van den Hof, B Wahlberg, *Modelling and Identification with Rational Orthogonal Basis Functions*. (Springer, London), (2005).
5. AH Ribeiro, K Tiels, J Umenberger, TB Schön, LA Aguirre, On the smoothness of nonlinear system identification. *Automatica* **121**, 109158 (2020).
6. AH Ribeiro, K Tiels, LA Aguirre, T Schön, Beyond exploding and vanishing gradients: Analysing RNN training using attractors and smoothness in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. (PMLR), pp. 2370–2380 (2020).
7. Z Liu, J Golowasch, E Marder, LF Abbott, A Model Neuron with Activity-Dependent Conductances Regulated by Multiple Calcium Sensors. *J. Neurosci*. **18**, 2309–2320 (1998).
8. AA Prinz, CP Billimoria, E Marder, Alternative to Hand-Tuning Conductance-Based Models: Construction and Analysis of Databases of Model Neurons. *J. Neurophysiol*. **90**, 3998–4015 (2003).