THE UNIVERSITY *of* EDINBURGH

MODELLING HETEROGENEOUS BIOMOLECULAR DYNAMICS IN NEURONS USING NONLINEAR MIXED EFFECTS MODELS AND SCIENTIFIC MACHINE LEARNING

DOMAS LINKEVIČIUS

THE UNIVERSITY
of EDINBURGH

Doctor of Philosophy
Centre for Doctoral Training in Biomedical Artificial Intelligence
Institute for Adaptive and Neural Computation, School of Informatics
University Of Edinburgh

2025

ABSTRACT

Computational neuroscience employs mathematical, computational and physical abstractions of biological structures ranging from the scale of ion channels to systems to investigate the brain and the central nervous system. Computational neuroscience has a well-established set of tools and methodologies which are used to handle the available data. However, with the development of new experimental techniques, automation of experiments and increased data-sharing, different analytical tools may be necessary to best use and understand the high volumes of novel data. In this thesis I use two tools which have hitherto had limited application in computational neuroscience – non-linear mixed effects modelling (NLME) and scientific machine learning (SciML). NLME is a hierarchical modelling framework that can account for the sources of within- and between-subject variability. SciML is a discipline in which machine learning is blended with classical mathematical models. By supplementing the set of existing computational neuroscience approaches with NLME and SciML, I tackle three distinct topics that are related to synaptic plasticity and neuronal function.

The first topic is related to the modelling of chemical reaction networks in neurons. There are many published studies containing a relatively large amount of different protein species. Among them is calmodulin, a Ca^{2+}-binding molecule essential in cellular signalling, particularly in synaptic plasticity, learning and memory. There are many different published Calmodulin models, but there is no systematic comparison between the different models. I used the data from Faas et al. (Faas et al., 2011) to fit and evaluate the existing Calmodulin models. The Faas et al. data contains the most accurate measurement of Calmodulin-Ca^{2+}binding dynamics to date. However, it suffers from uncertainties of important experimental factors which were not possible to control fully. Therefore, I used NLME to account for the uncertain experimental factors, using the data set to compare various published Calmodulin-Ca^{2+}binding schemes, analysing their ability to fit the data, and showing that some schemes fail due to structural limitations.

The second topic deals with the modelling of voltage-gated ion channels. More specifically, the class of the Hodgkin-Huxley-like ion channel models that assume independence between the different channel gating variables. I use the data of Ranjan et al. (Ranjan et al., 2019) to fit and evaluate various models of the voltage-gated potassium channel (K_v) gating dynamics. The data from Ranjan et al. contain current recordings of many different K_v types. However, their

data showed significant heterogeneity in the gating kinetics of the same K_v type. Therefore, I applied NLME to account for within- and between-K_v type heterogeneity. Moreover, I applied SciML to facilitate the Hodgkin-Huxley-like model fitting to multiple different K_v types. Application of these tools led to the creation of a single neural-network based model capable of accurately modelling the gating dynamics of 20 different K_v types.

The third topic I discuss is the application of NLME and SciML in the modelling of synaptic plasticity. Modelling of synaptic plasticity poses a significant challenge to the existing approaches used in computational neuroscience. The number of different protein species present in synapses, along with the heterogeneity of protein numbers in the same type of synapse, makes it very difficult to construct kinetic schemes capable of explaining various observed forms of synaptic plasticity. Therefore, I discuss various ways of applying NLME and SciML which could be productive in creating models of synaptic plasticity.

In conclusion, this thesis demonstrates the benefits of NLME and SciML in supplementing the existing toolbox of computational neuroscientists. These two additional tools were applied successfully to two challenging data sets and led to new insights, such as the limitations of the Calmodulin models used in the literature and the possibility of representing 20 different K_v channels using a single model. A wider adoption of these tools could result in solutions to a number of other existing challenges in neuroscience.

LAY ABSTRACT

Computational neuroscience uses a well-established set of mathematics and computer science tools to investigate the brain and the central nervous system. However, with the development of new experimental techniques, different mathematical tools become necessary. I use two new tools which have not yet seen wide application in computational neuroscience – non-linear mixed effects modelling (NLME) and scientific machine learning (SciML). NLME is a framework that allows us to model different sources of variability – within groups and between groups. SciML is a discipline in which machine learning is combined with classical mathematical models. I used NLME and SciML on three different topics in computational neuroscience.

The first topic is related to the modelling of chemical reactions in neurons. Some experiments are not able to control all of the relevant experimental factors. I used NLME to account for uncertainties in experimental conditions in experiments investigating calmodulin, an important cellular molecule. I compared various published calmodulin models, analysing their ability to reproduce the experimental data, highlighting that some models fail due to their structural limitations.

The second topic dealt with the modelling of voltage-gated ion channels, which are necessary for signal propagation in the brain. Creating models of voltage-gated ion channels is a complex task since they can exist in many different states, but only the states which conduct current are observable. Moreover, experimental data showed significant differences in behaviour between cells that contained identical channel types. I applied NLME to account for the different sources of variability between cells and channel types. Moreover, I applied SciML to facilitate the creation of different ion channel gating models. Importantly, I created a single model able to reproduce data of 20 different channel types, whereas past approaches required a single model per channel type.

The third topic I discussed is the application of NLME and SciML to model synaptic plasticity, which is related to learning at the cellular level. Modelling of synaptic plasticity is very complex due to many different protein species involved and differences between synapses. Therefore, I discussed various ways of applying NLME and SciML which could help to create better models of synaptic plasticity.

In conclusion, this thesis demonstrates the benefits of NLME and SciML as tools in computational neuroscience. These tools were applied successfully to two challenging problems and led to new insights. A wider adoption of these tools could result in solutions to a number of other existing challenges in neuroscience.

ACKNOWLEDGMENTS

There are many people who I would like to thank, who helped me in the journey to complete my PhD thesis. Since I am generally not one to mince words and I prefer expressing things through actions rather than words, the terseness of these acknowledgements should not be taken to mean the absence of gratitude.

First and foremost, I must extend gratitude towards my mother who has sacrificed a lot to bring me up. She is the reason I: got to Edinburgh, am the way that I am, am alive.

Secondly, I am deeply grateful towards Dr. David Sterratt who is my primary supervisor and has, among his myriad of other responsibilities, dedicated the time to supervise me, starting from my first MScR, up to the end of my PhD studies. He is the reason I: am going to be a PhD, have grown tremendously as a researcher, have been able to explore as many things as I did.

Finally, I extend my heartfelt gratitude towards E. She is the reason I: stayed in Edinburgh, became a man (if a little too late), started to try learning how to sing.

Inevitably, there are many more people who have been a part of this journey and whom I must thank (apologies for any misspellings of names or missing accents, etc.). In no particular order: Dr. Melanie Stefan, Dr. Angus Chadwick, Dr. Michael Gutmann, Dr. Meriem El Karoui and Dr. Aušra Saudargienė, for excellent supervision I have received from you at certain points throughout my studies. I also must thank the DeepPumas team for the opportunities provided and all the lessons learned: Niklas, Mohamed, Lorenzo, Andreu. Moreover, I have to thank all the people who made the PhD bearable in their own little, yet meaningful and significant ways: Nuria, Matus, Kasia, Nikitas, Rayna, Achille, Lucas, Leo, Karim, Argyris, Isha, Ola, Michal, Aryo, Max, Robert, the Medics MMA squad, the Edinburgh Muay Thai squad, the Edinburgh Judo squad, everyone else who I will remember having submitted this and will hate myself for having forgotten. A day among such great people demands a lifetime of efforts to deserve it. I would also like to thank the squirrels, crows and rabbits of Edinburgh (not the seagulls), the Lidl on Nicolson st., the Meadows, the Holyrood park, the Pleasance gym, the Informatics Forum, the Highlands and the Scottish rain. Finally, I would like to thank life. Living life has made me insane. I think I am fine with it.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION AND BACKGROUND

But Mousie, thou art no thy-lane,
In proving foresight may be vain:
The best laid schemes o' Mice an' Men
Gang aft agley,
An' lea'e us nought but grief an' pain,
For promis'd joy!

Excerpt from "*To a mouse (On Turning her up in her Nest, with the Plough)*" by Robert Burns, November 1785

1 INTRODUCTION

Whether it is the oxygen molecules binding to haemoglobin in the lungs of vertebrates or the photosynthesis reactions in the leaves of plants, chemical reactions, transformations of a certain kind of chemical substance to a different kind, underpin all life on earth. Billions of chemical reactions are occurring in our bodies each second, without our conscious input or control. Chemical reactions generally do not occur in isolation: the products of one reaction can be the reactants of another, forming highly complex chemical reaction networks. Therefore, characterization, description and understanding of the functions of various chemical reaction networks is essential to understanding life. Even though some of the chemical reaction networks are well-understood, for example, the Krebs cycle, which is essential in utilizing energy from nutrients, others, due to various reasons are not, necessitating further investigations and research.

Describing a chemical reaction network is a difficult task as each network is made up of multiple components. There are three key pieces of information necessary to describe any chemical reaction network: (i) the chemical species which are the basic building blocks in the network; (ii) the chemical complexes formed out of the building blocks; (iii) the reactions describing which species produce other species in the network (Feinberg, 2019). Whether it is the chemical reactions occurring when a cake is being baked or the refining of oil, knowing these three components is essential.

Some of the most complex phenomena that are possible to represent via chemical reaction networks can be found in the brain, for example, biochemical signalling in learning and memory (Magee and Grienberger, 2020) and information processing (Tozzi, 2015). The number of reactions underpinning complex neural phenomena is staggering – there are at least tens of thousands unique types of molecules and hundreds of thousands of different protein-protein interactions in the mammalian brain, forming various different local chemical reaction networks within different cells, often in specific parts of cells (Sorokina et al., 2021). Based on the nature and the amount of the available knowledge, there are different challenges in describing and understanding different types of chemical reaction networks which can broadly be categorized into three different cases.

First, there are situations in which there is full knowledge about the basic species present in the chemical reaction network, as well as significant knowledge about the complexes which are formed and reactions which are forming them. In such cases the main challenge

is determining the parameters of the individual reactions such that the observed data is explained well, distinguishing between the alternative possible reaction structures that may explain the data equally well and accounting for the experimental noise and the uncertainties in the experimental conditions. Of the three cases, finding an accurate description of the underlying chemical reaction network is relatively the easiest and most well-defined problem. However, due to the costs associated with conducting the necessary experiments, it is the rarest.

Second, there are situations when a chemical reaction network is used as an abstraction to represent, for example, ion channel gating (Hodgkin and Huxley, 1952). Such cases generally have one or a few observable variables which guide the creation of the model, for ion channels it would be the recorded currents conducted by the channel. However, in this case the states within the chemical reaction network are generally used as abstractions without a clear correspondence to conformational structure of the channel. Since it is not possible to use observations to constrain the abstract states, they are underdetermined, i. e. there are many different models that may fit the observed data to the same degree. Therefore, some additional Occam's razor-like criteria may be necessary to produce models with the most explanatory power while maintaining some level of parsimony. The main area where these kinds of chemical reaction networks arise is in the aforementioned modelling of the gating of ion channels, which are an important subject in computational neuroscience and therefore have received a lot of attention, coupled with significant amounts of measured data.

Finally, the third and the most common situation arising in the modelling of the chemical reaction networks present in the brain is that there are significantly more species and complexes in the network, than there is data to constrain the parameters of reactions that produce them. Moreover, the species that are included in the model contain only a small part of the real number of species (Heil et al., 2018), making the models potentially both under-constrained and incomplete. Even though the knowledge of what chemical species constitute the neuronal proteome is expanding (Roy et al., 2018; Sorokina et al., 2021), the knowledge of the kinetics behind the protein-protein interactions is not keeping up due to its significantly larger experimental cost. Therefore, this case, despite being the most common, is also the most challenging, requiring different methodological approaches compared to the previous two.

Two common themes unify the three cases outlined above. First of all, neuronal data is heterogeneous, whether it is the composition of the synaptic proteome (Roy et al., 2018) or the kinetics of ion channels (Ranjan et al., 2019). Secondly, there are many complex functional relationships involved in the neuronal chemical reaction networks. Some examples are the incomplete and under-constrained models of the

synaptic chemical reaction networks with many species (Heil et al., 2018) or the voltage-dependent forward and backward reaction rates used in the gating of ion channels (Ranjan et al., 2019). Therefore, in order to deal with these challenges, I introduce additional methodologies which have been used in other scientific fields to deal with the same challenges when dealing with the chemical reaction networks in the brain. The two extensions to the current approaches are the nonlinear mixed effects (NLME) modelling and scientific machine learning (SciML; more details and background on both will be provided in Chapter 2). Briefly, SciML refers to a set of approaches wherein the classical scientific models, such as systems of ordinary differential equations, or partial differential equations, etc. are integrated with modern machine learning methodologies (Baker et al., 2019). I use SciML to address the complex functional relationships present in the neuronal chemical reaction networks (see below for specific applications in this thesis). NLME modelling is a methodology that takes into account both within- and between-subject variability present in many different data sources (Lindstrom and Bates, 1990), for example the variability present in the data due to uncertainty in experimental conditions (Faas et al., 2011) or the variability between kinetics of ion channel gating present in cells expressing the same voltage-gated ion channel type (Ranjan et al., 2019). I apply these two methodologies to tackle three specific challenges, each corresponding to one of the aforementioned cases of neuronal chemical reaction networks.

FIRST CONTRIBUTION In Chapter 3 I address an instance of the first situation in which there is plenty of data and the components of the chemical reaction network are relatively well known. Some of the best characterized chemical reaction networks in the brain are those that are involved in the postsynaptic Ca^{2+} signalling cascade (Bartol et al., 2015; Jedrzejewska-Szmek et al., 2017). Ca^{2+} is a key ion in various forms of synaptic plasticity, which is defined as cellular and molecular changes in response to certain patterns of stimuli. Synaptic plasticity is a molecular correlate of learning and memory (Magee and Grienberger, 2020). There are multiple detailed empirical and computational investigations into the intracellular control (buffering) of Ca^{2+} levels, as well as Ca^{2+} involvement in synaptic plasticity via the molecules that it interacts with (Jedrzejewska-Szmek et al., 2017; Kim et al., 2011; Nair et al., 2014; Sabatini, Oertner, and Svoboda, 2002; Scheuss et al., 2006). However, many of the chemical reaction network models resulting from and used in the investigations of the Ca^{2+}-related reaction networks are poorly validated, lacking a systematic comparison between different published models. Given the role that Ca^{2+} plays in neuronal chemical reaction networks and the available amounts of data, it is essential to have well-validated models of the Ca^{2+}-related chemical reaction networks, especially if they are used

as components in larger chemical reaction networks, reducing the overall uncertainty in their parametrisation.

Chapter 3 provides an in-depth analysis, evaluation and fitting of chemical reaction networks to data for one of the key neuronal molecules Calmodulin (CaM; Berridge, Bootman, and Roderick, 2003; Xia and Storm, 2005) and its interactions with Ca^{2+}. The Ca^{2+}-CaM cascade has been one of the most common parts in the investigations of the synaptic chemical reaction network (Heil et al., 2018). I use a dataset presented in Faas et al. (2011), which contains fluorescence measurements in a system where Ca^{2+} is binding CaM. To fit models to the data, NLME modelling was necessary to address uncertainty in the experimental conditions under which the binding was investigated. Even though Ca^{2+}-CaM interactions are a common component of many different models, the different Ca^{2+}-CaM models used in various different studies have not been systematically investigated. Therefore, Chapter 3 fills this gap in the literature and provides additional insights into the structural features that are necessary for an accurate Ca^{2+}-CaM model, raising a question whether the previously published larger models that use suboptimal Ca^{2+}-CaM schemes should be revisited in order to confirm their findings.

SECOND CONTRIBUTION In Chapter 4 I address an instance of the second situation in which chemical reaction networks are used as an abstraction of the underlying system, the voltage-gated potassium channels (K_v). The dataset published in Ranjan et al. (2019) showed that there is significant kinetic heterogeneity within the cells expressing the same K_v type, in addition to the expected between channel type variability. This kinetic heterogeneity presents a challenge to the currently used modelling approaches, which do not inherently include components to account for it. Therefore, I used NLME modelling to account for the within-K_v family heterogeneity in addition to the between-K_v family differences. Moreover, deriving models for each K_v family is a complex task, therefore I used SciML to represent the functions that are used in the gating models via neural networks, fitting them in a data-driven manner. Importantly, previously used approaches resulted in a single model per K_v type, whereas, using these two methodologies, I was able to construct a single unified K_v model capable of fitting 20 different channel types. Therefore, Chapter 4 addresses an important and practically relevant challenge in computational neuroscience – modelling of the ion channel gating – in a novel and more powerful way.

THIRD CONTRIBUTION Finally, in Chapter 5, I address the third and final case, in which there is a large and complex chemical reaction network, but there are significant gaps in knowledge about a large number of its constituent species and reactions. The best ex-

emplar of this case is the synaptic chemical reaction network. Many attempts at modelling it have been made, yet most have included only a very small fraction of the total number of unique proteins found in a synapse (Heil et al., 2018). Since most of the chemical reaction network is not observed, it is next to impossible to properly inform most of the reaction parameters in it. Moreover, since a large part of the species are not included in the model, the chemical reaction network is fundamentally incomplete. Therefore, in Chapter 5 I discuss how the SciML and NLME ideas can be used to circumvent both of these challenges by representing the species not included in the chemical reaction network as a black box neural network model, providing concrete ideas to which species should it be linked to and how. Such an approach would offer flexibility, removing the need to account for the incompleteness of the chemical reaction network by adjusting the reaction rates for parts of the network that are well-constrained. Furthermore, it would provide additional clarity, delimiting the known parts from the unknown parts. In particular I build on the work presented in Chapter 3, focusing on the Ca^{2+}-CaM dependent cascade, its involvement in synaptic plasticity, as well as presenting ideas on how to best integrate it with other pathways that respond to different stimulation protocols (Edelmann, Cepeda-Prado, and Leßmann, 2017; Magee and Grienberger, 2020). The ideas presented in Chapter 5 form a natural extension and a research plan using the ideas presented in this thesis, which, unfortunately, I did not have the time to undertake.

These three chapters are followed by a higher level discussion in Chapter 6, focusing on Chapter 3 and Chapter 4, discussing limitations and presenting opportunities for future work. The thesis is concluded with Chapter 7, returning back to the high level considerations on the understanding of the brain and how the work undertaken in this thesis contributes to this goal.

2

BACKGROUND

In this chapter I provide the background that is necessary to understand the thesis at a technical level. The background starts with the basic definition of what a chemical reaction network is. Next, a brief overview of how chemical reaction networks are used in computational neuroscience is given, focusing on ion channel gating to illustrate their usage. Next, I overview the fitting of chemical reaction networks to data, making a distinction between the optimization of reaction rate constants and the structure of the chemical reaction network. Having described the basics of chemical reaction networks, I move on to the description of nonlinear mixed effects (NLME) modelling, providing the basic definitions, the description of the optimization objectives and algorithms used in NLME, followed by various examples of NLME usage. This is followed by a section on scientific machine learning (SciML), starting off with some basic definitions and examples of usage, describing three different existing approaches. With the definitions of NLME and SciML present, I describe the combination of these two tools and examples of studies where they have been fruitfully combined to deal with complex scientific challenges.

2.1 CHEMICAL REACTION NETWORKS

There are multiple ways in which it is possible to build up to models of chemical reaction networks from first principles (e. g. from Markov chains as given in Anderson and Kurtz, 2011). I will be following the theoretical framework set out in Feinberg (2019), presenting only the necessary definitions with examples where appropriate. I start with the basic definition of what a chemical reaction network is (Feinberg, 2019, p.24):

Definition 2.1.1. *A chemical reaction network consists of three sets:*

1. *a finite set $\mathscr{S}$, elements of which are the species of the network*
2. *a finite set $\mathscr{C}$ of vectors in $\overline{\mathbb{R}}_+^{\mathscr{S}}$ called the complexes of the network*
3. *a set $\mathscr{R} \subset \mathscr{C} \times \mathscr{C}$ such that*
   a) *for each $\mathbf{y} \in \mathscr{C}$, $(\mathbf{y}, \mathbf{y}) \notin \mathscr{R}$*
   b) *for each $\mathbf{y} \in \mathscr{C}$ there is a $\mathbf{y}' \in \mathscr{C}$ such that $(\mathbf{y}, \mathbf{y}') \in \mathscr{R}$ or $(\mathbf{y}', \mathbf{y}) \in \mathscr{R}$.*

Members of $\mathscr{R}$ are the reactions of the network. For each $(\mathbf{y}, \mathbf{y}')$ in $\mathscr{R}$, we say that complex $\mathbf{y}$ reacts to complex $\mathbf{y}'$, and we write the more suggestive

$\mathbf{y} \longrightarrow \mathbf{y}'$ *in place of* $(\mathbf{y}, \mathbf{y}') \in \mathscr{R}$ *if and only if* $\mathbf{y}$ *reacts to* $\mathbf{y}'$. *The vector* $\mathbf{y}$ *is called the reactant complex of the reaction* $\mathbf{y} \longrightarrow \mathbf{y}'$, *and* $\mathbf{y}'$ *is called its product complex.*

In this definition $\overline{\mathbb{R}}_+^{\mathscr{S}}$ is a vector of non-negative real numbers of dimensionality equal to the number of species $|\mathscr{S}|$. Also, note that in this definition the usage of the word "complex" does not mean two or more bound species, as it does in some modelling languages (e. g. Kappa or BNGL) or natural language, rather it refers to vectors of species participating in reactions. Moreover, if we take a species A, the notation is overloaded in that A means an element of a set when referring to A as a species and a vector when referring to A as a part of a complex. The technical details (here omitted for brevity) of how the two are linked can be found in Feinberg (2019, p.22). I use a simple chemical reaction network with three species and two reactions to illustrate the definition

$$A + B \rightleftharpoons C \tag{1}$$

where $\mathbf{y} \rightleftharpoons \mathbf{y}'$ is a shorthand for the reactions $\mathbf{y} \longrightarrow \mathbf{y}'$ and $\mathbf{y}' \longrightarrow \mathbf{y}$. For this chemical reaction network the set of species is $\mathscr{S} = \{A, B, C\}$, the set of complexes is $\mathscr{C} = \{A + B, C\}$ and the set of reactions is $\mathscr{R} = \{A + B \longrightarrow C, C \longrightarrow A + B\}$. Since $\mathbf{y} \in \mathscr{C}$ are vectors, the complex $A + B$ is also a vector

$$A + B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}$$

Note that the Definition 2.1.1 only mentions the structural aspects of a chemical reaction network. Before describing the dynamical aspects and kinetics of chemical reaction networks, I need to provide a few additional definitions. The definition of support (used later) is (Feinberg, 2019, p. 21)

Definition 2.1.2. *Support of a vector* $\mathbf{x} \in \mathbb{R}^{\mathscr{S}}$, *denoted* $\operatorname{supp}(\mathbf{x})$, means the subset $\mathscr{I}$ on which $\mathbf{x}$ takes non-zero values

$$\operatorname{supp}(\mathbf{x}) = \{i \in \mathscr{I} : x_i \neq 0\} \tag{2}$$

Using the system in the example in Equation 1, if we have a vector of concentrations of each species $\mathbf{c} = [c_A \; 0 \; c_C]^T$, where $c_A > 0$ and $c_C > 0$, then $\operatorname{supp}(\mathbf{c}) = \{1, 3\}$. I next define a kinetics, where c is the concentration of species (Feinberg, 2019, p. 25):

Definition 2.1.3. *A kinetics for a reaction network* $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ *is an assignment to each reaction* $\mathbf{y} \longrightarrow \mathbf{y}' \in \mathscr{R}$ *of a continuously differentiable rate function* $\mathscr{K}_{\mathbf{y}\to\mathbf{y}'} : \overline{\mathbb{R}}_+^{\mathscr{S}} \to \overline{\mathbb{R}}_+$ *such that*

$$\mathscr{K}_{\mathbf{y}\to\mathbf{y}'}(\mathbf{c}) > 0 \text{ if and only if } \operatorname{supp}(\mathbf{y}) \subset \operatorname{supp}(\mathbf{c}). \tag{3}$$

In plain terms the technical condition $\operatorname{supp} \mathbf{y} \subset \operatorname{supp} \mathbf{c}$ means that all of the species in a given reactant complex $\mathbf{y}$ have non-zero concentrations. One of the most well-known and most commonly used kinetics is the mass-action kinetics (Feinberg, 2019, p. 26), which I use later on:

Definition 2.1.4. *The kinetics $\mathscr{K}$ for a reaction network $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ is mass-action if, for each $\mathbf{y} \longrightarrow \mathbf{y}' \in \mathscr{R}$, there is a positive number $k_{\mathbf{y}\to\mathbf{y}'}$ such that*

$$\mathscr{K}_{\mathbf{y}\to\mathbf{y}'}(\mathbf{c}) = k_{\mathbf{y}\to\mathbf{y}'} \prod_{s\in\mathscr{S}} c_s^{y_s} \tag{4}$$

The number $k_{\mathbf{y}\to\mathbf{y}'}$ is the rate constant for the reaction $\mathbf{y} \longrightarrow \mathbf{y}'$, c_s is the concentration of species s in the current state of the system $\mathbf{c}$ and y_s is the s'th element of the reactant complex vector $\mathbf{y}$.

Combining the definitions of a chemical reaction network and kinetics (2.1.1 and 2.1.3 respectively) we get the definition of a kinetic system

Definition 2.1.5. *A kinetic system is a reaction network $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ taken with a kinetics $\mathscr{K}$.*

The next few definitions we present finally bring us to the definition of an ordinary differential equation that can be solved numerically in order to simulate the dynamics of a chemical reaction network

Definition 2.1.6. *Let $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ be a reaction network. The reaction vector corresponding to reaction $\mathbf{y} \longrightarrow \mathbf{y}' \in \mathscr{R}$ is the vector $\mathbf{y}' - \mathbf{y} \in \mathbb{R}^{\mathscr{S}}$.*

If we again take the example chemical reaction network given in Equation 1, the two reaction vectors of this system would be:

$$\begin{aligned} A + B \longrightarrow C: \qquad C - A - B &= \begin{bmatrix}0\\0\\1\end{bmatrix} - \begin{bmatrix}1\\0\\0\end{bmatrix} - \begin{bmatrix}0\\1\\0\end{bmatrix} = \begin{bmatrix}-1\\-1\\1\end{bmatrix} \\ C \longrightarrow A + B: \qquad A + B - C &= \begin{bmatrix}1\\0\\0\end{bmatrix} + \begin{bmatrix}0\\1\\0\end{bmatrix} - \begin{bmatrix}0\\0\\1\end{bmatrix} = \begin{bmatrix}1\\1\\-1\end{bmatrix} \end{aligned} \tag{5}$$

With these definitions we have the essential parts necessary for the simulation of the dynamics of a chemical reaction network

Definition 2.1.7. *For a kinetic system $\{\mathscr{S}, \mathscr{C}, \mathscr{R}, \mathscr{K}\}$ the species-formation-rate function $f(\cdot)$ is defined by the requirement that, for all $\mathbf{c} \in \overline{\mathbb{R}}_+^{\mathscr{S}}$,*

$$\frac{d\mathbf{c}}{dt} = f(\mathbf{c}) = \sum_{\mathbf{y}\to\mathbf{y}'\in\mathscr{R}} \mathscr{K}_{\mathbf{y}\to\mathbf{y}'}(\mathbf{c})(\mathbf{y}' - \mathbf{y}) \tag{6}$$

With this definition we have a general ordinary differential equation that can be solved to simulate the dynamics of chemical reaction networks. I will expand the basic definitions where appropriate, e. g. to include voltage-dependent kinetics in ion channel gating, but the basic definitions form the general backbone for all chemical reaction networks used in this thesis. Moreover, these definitions and basic features serve to inform and constrain the scientific machine learning approaches used later in the thesis. I next describe the usage of chemical reaction networks in computational neuroscience, with a particular emphasis on ion channel gating.

2.1.1 *Chemical Reaction Networks in ion channel gating*

An important application of chemical reaction networks is the modelling of ion channel gating. There are many different mechanisms that can control the opening and closing of ion channels – for example, various ligands (Alexander, Mathie, and Peters, 2011), intracellular Ca^{2+} (O'Reilly-Shah, Chagot, and Chazin, 2015), mechanical stimuli (Kefauver, Ward, and Patapoutian, 2020) – some of which can be represented as elements in a basic chemical reaction network with mass-action kinetics. Recall that the species-formation-rate function (Definition 2.1.7) requires a reaction network $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ with a kinetics $\mathscr{K}$. Whereas the gating mechanisms due to the binding of ligands or ions can be modelled via the mass action kinetics (Definition 4), different kinetics need to be used to model ion channel gating which may be voltage-dependent. For example, fractal models of ion channel gating assume that the reaction rate depends on the time the channel spent closed (Sansom et al., 1989). Other types of ion channel gating models assume specific reaction network structure, for example, diffusion models assume that the channel has a large number of identical closed states with identical parameters which are used to model the channel "diffusing" away from the open state (Sansom et al., 1989). However, in this thesis, the most important feature of ion channel gating is its dependence on voltage (Ahern et al., 2015; Ranjan et al., 2019) and the extensions to the basic chemical reaction network machinery that are necessary to account for it.

In order to account for the voltage-gating of ion channels it is essential to expand the mass-action kinetics by making the reaction rate constant $k_{\mathbf{y}\to\mathbf{y}'}$ voltage dependent.

Definition 2.1.8. *The kinetics $\mathscr{K}$ for a reaction network $\{\mathscr{S}, \mathscr{C}, \mathscr{R}\}$ is voltage-dependent mass-action if, for each $\mathbf{y} \longrightarrow \mathbf{y}' \in \mathscr{R}$, there is a non-negative function $k(V)_{\mathbf{y}\to\mathbf{y}'}$, where V is voltage, such that*

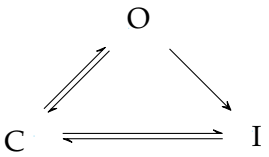
$$\mathscr{K}_{\mathbf{y}\to\mathbf{y}'}(\mathbf{c}, V) = k(V)_{\mathbf{y}\to\mathbf{y}'} \prod_{s\in\mathscr{S}} c_s^{y_s}. \tag{7}$$

An example chemical reaction network with voltage-dependent mass-action kinetics is

$$\mathrm{C} \underset{\beta(V)}{\overset{\alpha(V)}{\rightleftharpoons}} \mathrm{O} \tag{8}$$

where O denotes the open state, C denotes the closed state, the rate of transitioning from C to O is $\alpha(V) = 0.01\frac{V+10}{\exp(\frac{V+10}{10})-1}$ and the reverse rate is $\beta(V) = 0.125\exp(\frac{V}{80})$. The chemical reaction network in Figure 8 along with the functions $\alpha(V)$ and $\beta(V)$ formed the basis of voltage-gated potassium channel gating in the seminal work of Hodgkin and Huxley (1952).

The chemical reaction networks that model ion channel gating in the literature are generally tailored to different ion channel types and can contain multiple closed, open or inactivating states, each with a different reaction rate function. Due to their flexibility, such chemical reaction networks (also called Markov schemes) have been widely and successfully applied to model many different types of ion channels (Lampert and Korngreen, 2014). For example, a cyclic reaction network with open state O, closed state C and inactivated state I could be (reaction rates omitted for clarity)



(9)

Note that in this case the transition from the opened to the inactivated state is unidirectional, i. e. an inactivated state cannot open directly without passing through the non-inactivated closed state. There is no theoretical limit to the complexity such chemical reaction networks can take, which is both a strength and a weakness. The reason why it is a strength is the expressiveness such networks can provide. The main challenge is in determining which and how many states should be used – the only state that is ever observable is the open state, therefore all other states, their number and the topology of the network are under-parametrized and not identifiable from the data (more on that in Section 2.1.2). Therefore, most current models of ion channel gating use an approach first taken by Hodgkin and Huxley (1952).

The reaction networks used by Hodgkin and Huxley (1952) had a relatively simple structure. They used chemical reaction networks that could be decomposed into independent subnetworks of gating "particles". For example, for the voltage-dependent sodium channel, for which they observed activating and inactivating dynamics, they used two basic gating particles m (with three units) and h (with one unit) associated with a single channel in a patch of membrane. Each

of these particles had its own reaction network, independent of the other particle, specifically

$$\begin{array}{ccccccc} m_0 & \underset{\beta_m(V)}{\overset{3\alpha_m(V)}{\rightleftharpoons}} & m_1 & \underset{2\beta_m(V)}{\overset{2\alpha_m(V)}{\rightleftharpoons}} & m_2 & \underset{3\beta_m(V)}{\overset{\alpha_m(V)}{\rightleftharpoons}} & m_3 \\ h_0 & \underset{\beta_h(V)}{\overset{\alpha_h(V)}{\rightleftharpoons}} & h_1 & & & & \end{array} \tag{10}$$

where a channel is said to conduct current only if the gates are in the states m_3 and h_1 respectively. However, the scheme can be written jointly as well

$$\begin{array}{ccccccc} m_0h_0 & \rightleftharpoons & m_1h_0 & \rightleftharpoons & m_2h_0 & \rightleftharpoons & m_3h_0 \\ \downarrow\uparrow & & \downarrow\uparrow & & \downarrow\uparrow & & \downarrow\uparrow \\ m_0h_1 & \rightleftharpoons & m_1h_1 & \rightleftharpoons & m_2h_1 & \rightleftharpoons & \color{red}{m_3h_1} \end{array} \tag{11}$$

with the current conducting state in red. The gating particles are explicitly chosen to model certain features of the recorded data, such as activation or inactivation, leaving the voltage-dependent functions $\alpha_x(V)$, $\beta_x(V)$ and the "power" n_x (the number of units for each gate, $n_m = 3$ and $n_h = 1$) of each gate to be determined. Many different functional forms for $\alpha_x(V)$ and $\beta_x(V)$ have been used (with some models even using piece-wise definitions), but the generalized function that is used most commonly

$$\frac{A_p(V - B_p)}{\exp(\frac{V-B_p}{C_p}) - D_p} \tag{12}$$

resulting in a total of 4 parameters per gating particle, A_p, B_p, C_p and D_p. For example, to obtain $\alpha_n(V)$ and $\beta_n(V)$ from Hodgkin and Huxley (1952) one would need $A_p = 0.01, B_p = -10, D_p = 1, C_p = 10$ and $A_p = \frac{0.125}{V_m}, B_p = 0, D_p = 0, C_p = -80$ respectively.

The gating model is an essential but not the only part necessary to model the current conducted by a channel. The full equation used to model the conducted current over time $I(t)$ also depends on the channel density in a given patch of membrane $\bar{g}$, the reversal potential for the ion being conducted by the channel E and the voltage V, where $o(t)$ is the open channel fraction at time t, whether from a Hodgkin-Huxley-like or from a general Markov scheme

$$I(t) = o(t)\bar{g}(V(t) - E) \tag{13}$$

Combination of multiple such currents can explain complex electrophysiological phenomena, such as action potentials, spread of signals along dendritic trees, depolarization block and others. Therefore, accurate modelling of ion channel gating is important when trying to understand neuronal dynamics, signal processing and propagation.

2.1.2 *Fitting Chemical Reaction Networks to Data*

Having provided some examples of chemical reaction networks in modelling of synapses and ion channel gating, I will now overview some methods of fitting them to data. I make a distinction between the chemical reaction network structure and the reaction rate constants used for a given structure. The distinction might initially seem artificial since it is not possible to optimize the structure of a chemical reaction network without having some values for the reaction rates. However, it is practically relevant because a significant number of publications assume the chemical reaction network structure and then simply optimize the reaction rates. Therefore, I will describe algorithms that optimize both the chemical reaction network structure and the reaction rate constants, as well as algorithms that only optimize the reaction rate constants. There is a family of novel approaches that utilize machine learning as a core part, I discuss these as a part of the scientific machine learning approach in Section 2.3.

Before going into the algorithms developed to optimize different parts of the chemical reaction networks, it is important to discuss that a significant portion of them have been hand-tuned. Hand-tuning is a model optimization approach where a researcher uses some data, accompanied by large amounts of common sense, intuition, domain knowledge and, at times a large amount of, trial-and-error to find a model that fits the data. At best, it is an unsystematic means of model optimization that nonetheless has been used in many existing publications that produced many important results. It can be difficult to determine when hand-tuning was used, and some publications do not contain any information about how the model parameters or its structure were arrived at. In such cases, where the means of finding the parameters and/or the chemical reaction structure are not described, I assume that the model has been hand-tuned. Hand-tuning of models is not a scalable approach and therefore is progressively falling out of favour. The work undertaken in this thesis aims to further contribute to the deprecation of model hand-tuning.

2.1.2.1 *Whole model optimization*

Three sets are necessary to define a chemical reaction network, the species $\mathscr{S}$, the complexes $\mathscr{C}$ and the reaction vectors $\mathscr{R}$ (see Definition 2.1.1). Of these three sets, the set of species $\mathscr{S}$ is usually possible to determine empirically with relative certainty (but see Section 5.1). It is generally more difficult to establish the reactions $\mathscr{R}$ and the resulting complexes $\mathscr{C}$. The difficulty lies in the discrete optimization routines necessary to optimize $\mathscr{R}$, as well as the fact that generally a large amount of intermediate complexes are not observed empirically. Therefore, current approaches that attempt to infer the chemical reaction networks from some chemical species utilize quantum

mechanical simulations that directly simulate interactions between molecules (Unsleber and Reiher, 2025). Basing the chemical reaction network structure on quantum mechanical simulations allows to determine which species can form complexes via physical interactions. However, such quantum mechanical simulations are computationally expensive, especially for large molecules. For example, Zhao and Savoie (2021) investigate a system with 47 atoms and 11 rotatable bonds and state that it is at the edge of what is computationally feasible. For reference, a CaM molecule consists of 148 amino acids and each amino acid can have around 20 atoms, resulting in 3000 atoms to simulate. While there are more coarse grained simulation methods than quantum mechanical, e. g. molecular dynamics, which are capable of simulating the interactions between CaM and parts of a voltage-gated Ca^{2+} channel (Yaduvanshi, Ero, and Kumar, 2021), a simulation-based screening of possible interactions does not currently seem feasible. Therefore, since such whole-model discovery and optimization techniques are not applicable to the chemical reaction networks of synapses and ion channel gating, they are outside the scope of this thesis.

2.1.2.2 *Reaction Rate Constants*

The task of finding a set of reaction rate constants once the chemical reaction network is fixed is significantly less complex. Since the reaction rate constants can be expressed as variables on the real line (by log transforming them before optimization to make them unbounded), there is an abundance of classical optimization techniques that could be used. For example, it is possible to use gradient-based approaches to fit reaction rate constants (Pepke et al., 2010). There are many gradient-based approaches which can be used to find optimal parameter values, such as BFGS (Zhu et al., 1997) or conjugate gradient, obtaining the gradients of the loss with respect to the parameters e. g. via sensitivity analyses (Turanyi, 1990) or other methods (Anand Krishna, Krishna Dutt, and Premchand, 2023). It is advisable to use second order gradient-based methods that take into account the curvature of the loss surface because it can be highly elliptical with steps based only on the gradient being highly inefficient (Daniels et al., 2008; Gutenkunst et al., 2007). However, gradient-based optimization techniques generally suffer from getting stuck in local optima, as there are generally no guarantees that the loss surface being optimized is convex. Therefore, some works have used alternative approaches, for example, particle swarm (Byrne et al., 2009), genetic algorithms (Polifke, Geng, and Döbbeling, 1998), or a mix of approaches, e. g. initial fitting done via a genetic algorithm and fine-tuning via a gradient-based approach (Maeder, Neuhold, and Puxty, 2004).

An alternative to gradient-based optimization techniques, particularly relevant to chemical reaction networks due to the necessity of using simulators, is simulation-based inference (SBI, Cranmer, Brehmer, and Louppe, 2020). Simulation-based inference relies on simulators to define statistical models on which the inference is done, specifically, given a vector of parameters θ, the simulator produces a sample of latent variables $z_i \sim p_i(z_i|, \theta, z_{<i})$ (here the subscript $_{<i}$ denotes the state before i) which are used to produce the data vector $x \sim p(x|\theta, z)$. In the context of this thesis simulators are ODE solvers which simulate the dynamics of a chemical reaction network. More traditional inference techniques usually require a likelihood function, but SBI does not necessarily rely on it, using different means of establishing the model fitness to data. Some examples of alternatives to the likelihood function are approximate likelihood ratio estimation (Hermans, Begy, and Louppe, 2020), neural-network based surrogates (Gloeckler et al., 2025) and others (Cranmer, Brehmer, and Louppe, 2020). SBI approaches have been successfully applied in many scientific problems, for example the inference of the densities of ion channels in a compartmental neuron model (Lueckmann et al., 2017), on gravitational-wave data (Dax et al., 2025), and EEG, ECoG and LFP data (Vetter, Macke, and Gao, 2024). Therefore, SBI should become a more prevalent modelling approach for models with intractable likelihoods once a standard set of approaches that leverages the advances in machine learning is converged to. However, not all SBI approaches require machine learning and I overview some of them next.

The Bayesian family of approaches deserves a particular focus due to its prevalence and utility. These approaches result in a posterior distribution of parameter values, resulting in a more detailed understanding compared to a point estimate obtained via the aforementioned approaches. For example, if a parameter has a wide posterior distribution, it is likely under-constrained given the current data and should be taken with more caution. Some Bayesian approaches used in literature are approximate Bayesian computation (Eriksson et al., 2019), which omits the usually costly likelihood-based evaluation of the sampled parameters and uses an easier to compute rejection criterion, as well as a full Markov Chain Monte Carlo approach, which produces an approximation to the posterior by sampling from it (De Oliveira et al., 2025). It is worth elaborating further on the MCMC approaches, which are among the most often used tools.

There are many software suites that implement different types of MCMC, for example, Hamiltonian Monte Carlo (HMC) implemented in the `Stan` package (Carpenter et al., 2017), the `PyMC` package (Abril-Pla et al., 2023), which contains the No U-Turns Sampler (NUTS), implemented in Python, and the `Turing.jl` in Julia (Fjelde et al., 2025). There are many factors that affect the practicality of Monte Carlo based approaches. Fitting done in this thesis requires integrating the

model, a system of ODEs, over time, as well as being able to handle large numbers of parameters arising from the usage of neural networks. Even though HMC has been used to optimize neural networks (Neal, 1996), the cost of evaluating gradients over time introduces a significant bottleneck, necessitating highly performant ODE solvers. Benchmarks on non-stiff and stiff ODE problems (see https://docs.sciml.ai/SciMLBenchmarksOutput/stable/MultiLanguage/ode_wrapper_packages/) show that the ODE solvers in Julia are the most performant, making them invaluable, enabling the work done in Chapter 4.

To conclude, the task of inferring the reaction rate constants once the structure of a chemical reaction network is fixed, even though still a complex problem, is significantly easier than the joint optimization of both the structure and the reaction rate constants. Methods widely applied in machine learning and optimization can also be applied to this task as well, along with their caveats, strengths and weaknesses.

2.1.2.3 *Model identifiability*

Fitting of model parameters is a complex task that can be approached in multiple different ways, as discussed above. However, even with well-behaved and powerful algorithms and a rich data source, significant challenges may still remain. One of these challenges is identifiability – the existence of a unique solution to the optimization of the reaction rate constants or the chemical reaction network structure. In fact, there are chemical reaction systems for which a set of unique reaction rate constants may be impossible to determine (Craciun and Pantea, 2008). Craciun and Pantea (2008) prove that the reaction rate constants for a chemical reaction network are identifiable if the reaction vectors are linearly independent. For example, the reaction rate constants of the chemical reaction network in Figure 1 can not be identified uniquely because $r_3 = a_1 r_1 + a_2 r_2$ for some constants a_1 and a_2. Moreover, it is possible that two different chemical reaction networks result in dynamics that are identical. Craciun and Pantea (2008) provide the two networks shown in Figure 2.

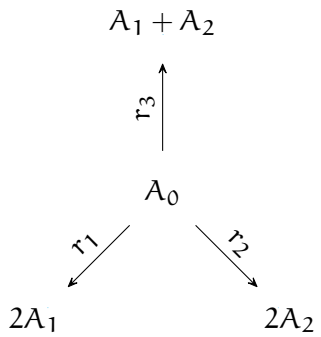


Figure 1: An example of a chemical reaction network for which the reaction rate constants are non-identifiable.

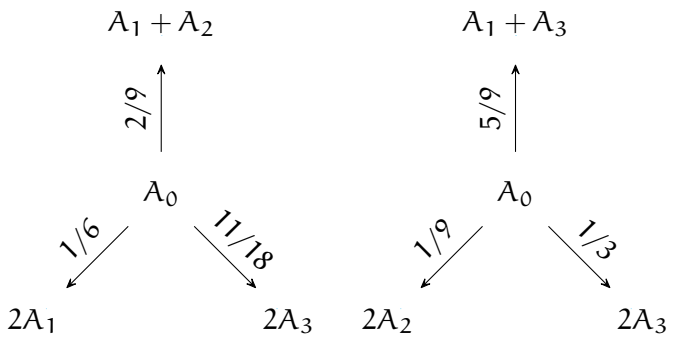


Figure 2: An example of two structurally different chemical reaction networks that result in identical dynamics.

Therefore, even if a sufficiently rich data set is used, there are chemical reaction networks which may not be distinguishable from one another because, given mass-action kinetics, they result in identical differential equations. Such chemical reaction networks are called confoundable. These two forms of non-identifiability are also known as structural non-identifiability.

However, the model structure is not the only factor to consider. As mentioned above, I assumed the availability of a sufficiently rich data, which is not always the case. Therefore, another form of non-identifiability is practical, or data, non-identifiability. Wieland et al. (2021) define practical non-identifiability via confidence intervals – a parameter is identifiable if, after fitting, it has a finite confidence interval. In these terms structurally non-identifiable parameters have infinite confidence intervals. It is important to note that identifiability can be a feature of specific parameters, rather than of individual models. For example, if an experiment measures the dynamics of a sufficiently large subset of the chemical reaction network, but contains no information about other parts of it, the parameters in measured parts may be practically identifiable, whereas the parameters in the unmeasured parts may be practically non-identifiable.

Considering the synaptic and the ion channel gating chemical reaction networks, it is immediately clear that most of their parameters are currently practically non-identifiable due to the paucity of data. Moreover, it has been shown that the voltage dependent equilibrium fractions in the Hodgkin-Huxley-like models of ion channel gating are structurally non-identifiable (Walch and Eisenberg, 2016). Given these practical and theoretical results, identifiability concerns are an important consideration in model fitting.

2.2 NON-LINEAR MIXED EFFECTS MODELLING

A large number of modelling paradigms rely on the assumption that a single set of parameters coupled with a suitable model architecture can be fit to a set of observations of interest which are independent

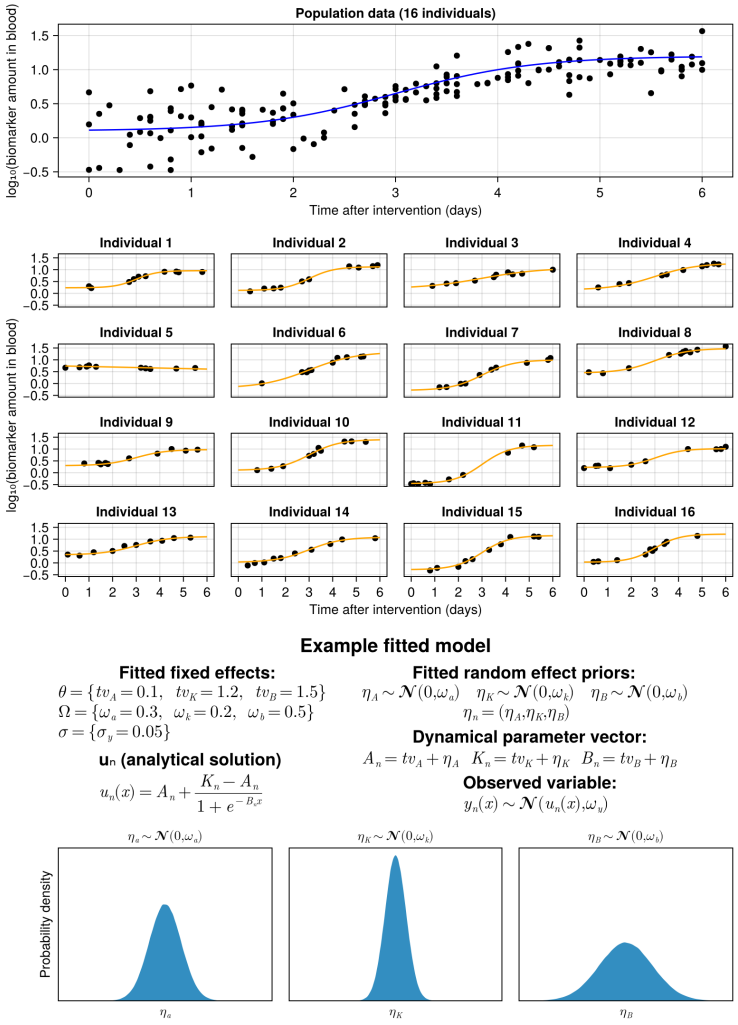


Figure 3: A basic NLME model fitting example where the dynamics could be solved analytically. The orange lines in the individual plots are the individualized predictions, whereas the blue line is the population level prediction with parameters tv_A, tv_K and tv_B, where tv stands for typical value.

and identically distributed (iid). For example, in image classification it is assumed that all the images used to train a classifier can be classified successfully using a single set of parameters, or it is assumed that a set of drug pharmacokinetics measurements can be fit by a dynamical system via a single set of parameters. However, often one sample provides information about another sample, thus breaking the iid assumption. For example, a gaze estimation data set can contain multiple measurements from the same person (Xiong, Kim, and Singh, 2019) or a psychological intervention data set can contain multiple measurements from the same person (Laban, Chiang, and Gunes, 2025). In such cases it is possible to distinguish between the

variance between different measured entities (e. g. different people) and the variance within a measured entity (same person).

One way to systematically account for the different sources of variance is mixed-effects modelling, specifically its most general form – non-linear mixed effects modelling (NLME, see Figure 4 for a visual representation and Figure 3 for an example data fitting procedure using a simple NLME model). In general, mixed effects models can have an arbitrarily deep hierarchy, for example, repeated measurements from students at particular schools in particular countries (three levels), but this thesis uses only two levels due to the current restrictions in the software used.

NLME modelling has historically been motivated by problems in medical and clinical research (Pillai, Mentré, and Steimer, 2005). The need to account for the within- and between-subject variability is important in, for example, estimating whether a drug or a treatment had a significant effect on a cohort of patients. However, more recently, the scope of NLME applications has expanded, tackling questions in biomedical sciences more broadly. NLME models have been used in infectious disease modelling (Adéoti et al., 2025) with over 124 papers dealing with various infectious diseases such as COVID-19, Ebola, Dengue and others. The NLME approaches are inherently well-suited for such applications due to the heterogeneity of recording lengths, robustness to missing data and model flexibility. NLME has been applied to a systems biology problem of parameter inference for multiple cells and compared to a standard approach used in the field, finding that NLME is favourable under conditions of lower amounts of data (Karlsson et al., 2015). Notably, NLME has also been applied in neuroscience, to analyse the electrophysiological features of mouse stellate cells, such as their resonance frequency, spike maximum, membrane potential (Pastoll et al., 2020). Finally, NLME modelling has also been applied in scientific fields other than biomedical sciences, for example, it was used it to model tree diameters (Xu et al., 2014), vegetation moisture (Oddi et al., 2019) and energy demands in buildings (Roach, Hyndman, and Ben Taieb, 2021). Therefore, NLME modelling is finding more and more applications and, if current trends hold, its usage will extend to more different scientific fields.

I adapt the definitions of NLME provided in Duchesne et al. (2021) and Rackauckas et al. (2022) which are more specific for dynamical systems, compared to more generic definitions, e. g. Lindstrom and Bates (1990). The NLME modelling framework comprises a two level hierarchical structure, the fixed and the random effects. The upper level captures the between-entity variance in a set of parameters called the fixed effects Θ. Fixed effects can be broadly grouped into

- model parameters θ
- random effect prior distribution parameters Ω

- observation model noise parameters σ

The lower level consists of random effects η_n, where $n \in [1, N]$ and N is the number of entities in the data set and each $\eta_n \in \mathbb{R}^m$, i. e. it is an m-dimensional vector. Random effects account for the portion of the variability of the observations that is not shared between different entities by individualizing the model parameters θ. Therefore, contrary to the fixed effects Θ, random effects vary between observed entities (which can be individual people, cells, etc.), the nth entity having its own set of parameters η_n. Furthermore, a set of covariates Z_n (which are known at the outset) are associated with the nth entity, for example, the age and weight of the nth person, or the temperature and solution composition in which the nth cell was recorded, etc.

The three sets of values (θ, η_n and Z_n) are collated via the parameter model function g into the dynamical parameter vector p_n for the nth entity

$$p_n = g(\theta, Z_n, \eta_n) \tag{14}$$

Note that the covariates Z_n could impact the latter portions of the model, e. g. the observational model via the function h_j (h_j is defined below), but I omit repeated references to it for clarity. The dynamical parameters p_n are then fed into the structural model function f, which in this thesis are usually systems of ordinary differential equations (ODEs)

$$\frac{du_n}{dt} = f(u_n, p_n, t) \tag{15}$$

where u_n are the dynamical variables being solved for. The final step is to use the u_n values to derive the experimentally observed quantities y_{nj} (biomarker concentration in the blood, see Figure 3), where j denotes the jth observed quantity for the nth entity. Since the relation of y_{nj} to u_n may be complex, I denote it as $h_j(u_n(t))$. Finally, each observed quantity will generally be noisy, therefore an observational model is necessary:

$$y_{nj}(t) \sim \mathcal{D}_j\left(h_j(u_n(t)), \sigma_j\right) \tag{16}$$

where $\mathcal{D}_j$ is the error model probability distribution and σ_j are the parameters of the jth observational model. In most cases $\mathcal{D}_j$ is simply a normal distribution $\mathcal{N}(h_j(u_n(t)), \sigma_j)$ with standard deviation σ_j (overloading the notation slightly). Having presented the basic NLME framework, I will briefly present examples of various applications and link to relevant reviews for further reading.

There are many different optimization algorithms used to fit NLME models. Some of the most popular are stochastic approximation expectation maximization (SAEM) approaches (Dempster, Laird, and Rubin, 1977), likelihood-based approaches (Wang, 2007) and Markov

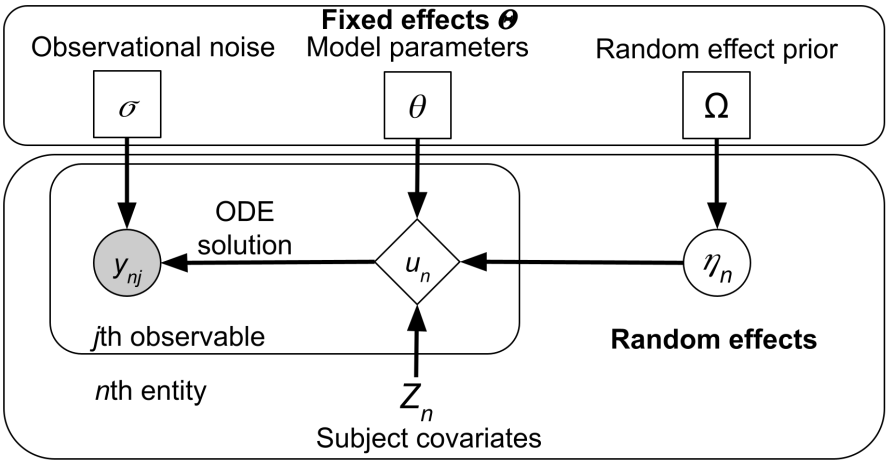


Figure 4: Visual representation of an NLME model, rectangle nodes in the top box denote parameters (fixed effects), circles denote random quantities which are either latent (unfilled) or observed (filled), diamonds are deterministic given the inputs, and nodes without a border are constant.

Chain Monte Carlo-based approaches (Lunn and Aarons, 2002). In this thesis I only use the likelihood-based approaches, therefore I will forego description of other approaches that are not relevant to the current work and instead delve deeper into the ones that are actually used.

2.2.1 *NLME Optimization Objectives*

There are two distinct groups of optimization objectives for NLME models that are used in this thesis; the first group optimizes the marginal likelihood (Wang, 2007), whereas the second optimizes the conditional likelihood (Sartori and Severini, 2004). Algorithms that optimize the marginal likelihood generally consist of a two level optimization scheme (optimizing the fixed effects while random effects are held constant and vice-versa), whereas conditional likelihood optimization is joint – the fixed and the random effects are optimized together. When creating the dynamical parameter vector p_n (see Equation 14) we need a single value for θ, as well as η_n.

Due to the different possible framings of the optimization objectives (see below), there are multiple ways in which θ and η_n values can be obtained. In particular, these values can be maximum likelihood estimates (MLEs), point estimates obtained via optimization procedures, or maximum *a posteriori* (MAP) estimates, modes of distributions, if fixed effects have priors (random effects by default are assumed to have priors). For conceptual clarity, I will split the presentation of the NLME optimization objectives along the lines in which

the θ and η_n values can be obtained and whether the objective is the marginal or the conditional likelihood.

MARGINAL LIKELIHOOD WITH MAXIMUM LIKELIHOOD ESTIMATES

The MLE marginal likelihood objective finds the vector of fixed effects Θ^* which maximizes the likelihood of the data set y. For numerical precision reasons and framing the problem as minimization, rather than maximization, the negative of the log of the marginal likelihood is usually used as the objective. I will illustrate the negative log marginal likelihood objective in this one case and omit it from further objectives because the derivations follow the same pattern:

$$\begin{aligned}\Theta^* &= \arg\min_{\Theta}[-\log p(y \mid \Theta)] \\ &= \arg\min_{\Theta}\left[-\log \prod_{n=1}^{N} p(y_n \mid \Theta)\right] \\ &= \arg\min_{\Theta}\left[-\sum_{n=1}^{N} \log p(y_n \mid \Theta)\right]\end{aligned} \tag{17}$$

where $p(y \mid \Theta)$ is the total likelihood from all observed entities, $p(y_n \mid \Theta)$ is the total likelihood for the nth entity, taking into account different observed quantities along with the number points over time and N is the total number of observed entities. I will now focus on a single term in the sum $\log p(y_n \mid \Theta)$ because deriving the details for it is enough to illustrate the general principle behind the optimization.

The marginal likelihood is called this because the random effects η_n are marginalized out:

$$p(y_n \mid \Theta) = \int p(y_n \mid \eta_n, \Theta)\, p(\eta_n \mid \Theta) d\eta_n \tag{18}$$

Unfortunately, the calculation of the marginal likelihood is generally intractable and therefore requires approximations. In practice multiple levels of approximation can be used, each progressively decreasing in accuracy due to more restrictive assumptions:

1. Laplace approximation
2. First order conditional estimate (FOCE) approximation
3. First order (FO) approximation

For full derivations of each of them see Wang (2007), here I will show the basic distinctions between each of these practically used approximations.

LAPLACE APPROXIMATION

The Laplace marginal likelihood approximation is obtained via the second order Taylor expansion of the marginal likelihood integral. For reasons that will become clear, let

us assume that the fixed effects are not varying so that the marginal likelihood is only a function of the random effects η_n and let

$$g(\eta_n) = \log f(\eta_n) = \log \left(p(y_n \mid \eta_n, \Theta) p(\eta_n \mid \Theta) \right) \tag{19}$$

The general form of the second order Taylor expansion of a function $g(\eta_n)$ around a point η_0 is:

$$g(\eta_n) \approx g(\eta_0) + (\eta_n - \eta_0)^T g\prime(\eta_0) + \frac{(\eta_n - \eta_0)^2}{2!} g\prime\prime(\eta_0) \tag{20}$$

where $g\prime$ and $g\prime\prime$ are the gradient vector and the Hessian matrix of the function g (recall $\eta_n \in \mathbb{R}^m$). There is a small technical difference between cases where η_0 is a mode of $f(\eta_n)$ (denoted η_n^*) and when it not, but I will skip these details and focus on the case where η_0 is not a mode (see Wang, 2007, for full details). Given that η_0 is not a mode of $f(\eta_n)$, for now disregarding how we obtain it, from Equation 18, the above definitions of $f(\eta_n)$, $g(\eta_n)$ and using the Taylor approximation and letting $\tau = (\eta_n - \eta_0)$ we get

$$\begin{aligned} \int f(\eta_n)\, d\eta_n &= \int \exp(g(\eta_n))\, d\eta_n \\ &\approx \int \exp(g(\eta_0) + \tau^T g\prime(\eta_0) + \frac{1}{2}\tau^T g\prime\prime(\eta_0)\tau)\, d\eta_n \\ &= f(\eta_0) \int \exp(\tau^T (g\prime(\eta_0) + \frac{1}{2} g\prime\prime(\eta_0)\tau))\, d\eta_n \end{aligned} \tag{21}$$

Notice that the term in the integral is simply the moment generating function of a multivariate Gaussian distribution with the mean $g\prime(\eta_0)$ and variance $g\prime\prime(\eta_0)$. Therefore, the integral gets replaced with the Gaussian density

$$\begin{aligned} p(y_n \mid \Theta) &\approx f(\eta_0) \int \exp(\tau^T (g\prime(\eta_0) + \frac{1}{2} g\prime\prime(\eta_0)\tau))\, d\eta_n \\ &= f(\eta_0) \frac{1}{\sqrt{(2\pi)^m | - g_n\prime\prime(\eta_0)|}} \exp\left(\frac{-g_n\prime(\eta_0)^T g_n\prime\prime(\eta_0)^{-1} g_n\prime(\eta_0)}{2} \right) \end{aligned} \tag{22}$$

Initially it may seem that the approximated likelihood does not depend on Θ, but recall that $f(\eta_n) = p(y_n \mid \eta_n, \Theta) p(\eta_n \mid \Theta)$. The final component we need to understand the successive approximation is expressing $g\prime(\eta_0)$ and $g\prime\prime(\eta_0)$ in terms of the conditional likelihood $p(y_n \mid \eta_n, \Theta)$. Omitting some constant terms, but still following Wang (2007), let the gradient of the conditional likelihood for the entity y_n w.r.t η_n be $\Gamma_n(\eta_n) = \frac{\partial p(y_n|\eta_n,\Theta)}{\partial \eta_n}$ and the Hessian be $\Delta_n(\eta) = \frac{\partial \Gamma_n(\eta)}{\partial \eta}$. It is possible to express $g\prime(\eta_0)$ and $g\prime\prime(\eta_0)$ via $\Gamma_n(\eta)$ and $\Delta_n(\eta)$ evaluated at η_0, whereas $\Gamma_n(\eta)$ and $\Delta_n(\eta)$ can either be obtained via automatic differentiation (e. g. via forward mode using

dual numbers) or, in rare cases, analytically. The only remaining question is the choice of η_0 in Equation 21 around which the Gaussian approximation is done. This value is called the empirical Bayes estimate (EBE) η_n^* of the random effect and is obtained by optimizing the conditional likelihood with the current estimate of the fixed effects Θ^*

$$\mathrm{EBE}_n = \eta_n^* = \arg\max_{\eta_n} \left(p(y_n \mid \Theta = \Theta^*, \eta_n) \cdot p(\eta_n \mid \Theta = \Theta^*) \right) \quad (23)$$

Therefore, the initial assumption of a fixed Θ is necessary to find the EBE values which would serve as the η_0 around which the Gaussian approximation in Equation 22 could be done. For simplicity the objective is written as maximization, but in practice it is often framed as the minimization of the negative of the log of the conditional likelihood. Using a gradient-based optimizer and given some initial values for Θ^* and η_n^*, the optimization scheme consists of two stages: the inner loop which finds an improved set of η_n^* and the outer loop, which uses the current η_n^* to find an improved estimate of the fixed effects Θ^*. This completes the brief description of the necessary terms required to approximate the marginal likelihood and solving the Equation 17 and Equation 23 via the Laplace approximation.

FOCE APPROXIMATION The FOCE approximation starts off exactly the same as the Laplace approximation until we come to the definitions of the gradient and the Hessian of the conditional likelihood. The gradient is still defined as in the Laplace approximation, $\Gamma_n(\eta_n) = \frac{\partial p(y_n|\eta_n,\Theta)}{\partial \eta_n}$, however, due to the difficulty and the cost of approximating the Hessian, it is instead approximated via the gradients by taking an expected value over different measured entities y_n, namely $\Delta_n(\eta) \approx \mathbb{E}_{y_n}\left[\Gamma_n(\eta)\Gamma_n(\eta)^T\right]$. The FOCE approximation to the marginal likelihood maintains many of the benefits of the Laplace approximation but with a decreased accuracy.

FO APPROXIMATION The final approximation to the marginal likelihood, FO, makes a much stronger assumption that may not always hold, therefore further significantly decreasing the accuracy of the marginal likelihood approximation. Specifically, it assumes that $\mathrm{EBE}_n = 0$ instead of finding a new value for it. Therefore, FO makes the marginal likelihood approximation essentially a single, rather than two, level optimization scheme, drastically reducing the computational cost of the approximation, but at the cost of significantly reduced accuracy which tends to significantly impact the quality of the model fits.

CONDITIONAL LIKELIHOOD WITH MAXIMUM LIKELIHOOD ESTIMATES Conditional likelihood is an alternative objective to the marginal likelihood approximations when training NLME models. It tra-

des off reduced computational cost for an increased chance of overfitting when compared to the marginal likelihood. Contrary to the marginal likelihood objectives, which require a two stage optimization scheme (higher level for Θ, lower level for $\mathtt{EBE}_n$, except for FO), using the conditional likelihood optimization the fixed effects Θ and the random effects η are optimized jointly, so the objective is (recall that N is the total number of entities/individuals)

$$\Theta^*, \eta^* = \arg\max_{\Theta,\eta} \prod_{n=1}^{N} p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta) \tag{24}$$

Many different optimization algorithms can be used to optimize it, but generally, due to the generally cheaper cost of computing the gradients via automatic differentiation tools, gradient-based optimization of the conditional likelihood is a viable means of fitting an NLME model. Also, in practice, similarly to the marginal likelihood, instead of directly maximizing the conditional likelihood, the negative log of it is minimized to avoid numerical accuracy issues.

MAXIMUM A POSTERIORI (MAP) OBJECTIVES The basic idea behind the maximum a posteriori objectives is to include prior knowledge about the fixed effects into the optimization objective. Practically this means including penalties due to such priors into the optimization objective, while analytically the objectives are expressed very similarly to the MLE objectives. For the marginal likelihood the MAP objective is

$$\begin{aligned} \Theta^* &= \arg\max_{\Theta} \; p(\Theta) \cdot p(y \mid \Theta) \\ &= \arg\max_{\Theta} \; p(\Theta) \cdot \prod_{n=1}^{N} \int p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta) \, d\eta_n \end{aligned} \tag{25}$$

Note that now the fixed effects Θ are treated as a random variable coming from a prior distribution $p(\Theta)$, rather than as a point estimate, but the outcome of this objective is still a point estimate which is a mode of the posterior, rather than the full posterior distribution. Similarly, the conditional likelihood MAP objective is

$$\Theta^*, \eta^* = \arg\max_{\Theta,\eta} \; p(\Theta) \cdot \prod_{n=1}^{N} p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta) \tag{26}$$

where $p(\Theta)$ is again some prior distribution of the fixed effects.

Re-iterating, the MAP objectives have the same underlying machinery as the MLE objectives, except there are prior distributions $p(\Theta)$ over the fixed effects which are practically encoded as penalties in the optimization. The prior distributions are generally chosen based on domain expertise and then turned into optimization penalties either analytically, e. g. the Gaussian prior, or can be used in gradient-based optimizers in concert with automatic differentiation tools.

2.3 SCIENTIFIC MACHINE LEARNING

Scientific machine learning (SciML, the research area, not to be confused with the Julia ecosystem which implements it https://sciml.ai/citing/) is the second tool applied to the problems in computational neuroscience in this thesis. SciML (a term seemingly first coined in Baker et al., 2019) draws both from mechanistic mathematical modelling, data science and machine learning, combining them to produce novel, more powerful data-driven models. Many different approaches are under this umbrella term, most notably physics-informed neural networks (Raissi and Karniadakis, 2018) and universal differential equations (Rackauckas et al., 2020). There are many examples that illustrate the general idea in a wide variety of scientific fields. For example, Dandekar, Rackauckas, and Barbastathis (2020) used a neural network to augment the differential equations of the epidemiological SIR model to account for unknown dynamics when modelling the COVID-19 data. Furthermore, Virtanen et al. (2020) used neural network terms in a partial differential equation system modelling an advection-diffusion-sorption system. Martensen et al. (2024) used a neural network to represent a feedback term in a pharmacokinetics/pharmacodynamics model. The breadth of examples showcases that these general ideas can be applied in many scientific fields, including the field of chemical reaction networks.

The field of SciML is experiencing rapid growth, development and validation of techniques and determination of the cases where and when it is useful. Investigators in many other fields, e. g. chemical engineering (Lima et al., 2023), pharmacology (Martensen et al., 2024), geosciences (Shen et al., 2023), are starting to introduce techniques from SciML to supplement the established mathematical modelling techniques in their fields.

2.3.1 *SciML and chemical reaction networks*

There are many different ways in which machine learning could be used in modelling chemical reaction networks. Here I will overview some recent chemical reaction network modelling publications that use machine learning to represent parts of the chemical reaction network. There are other strands of work where machine learning is used to improve the chemical reaction fitting process, for example, to predict the reaction rate constants based on the inputted reactants (Heid and Green, 2022; Liu, Mo, and Cheng, 2025) and these works, while valuable, are outside the scope of this thesis.

One of the most obvious ways in which machine learning can be used to model chemical reaction networks is by simply using universal function approximators to represent the species formation function. For example, neural ordinary differential equations (NODEs)

modelling (Chen et al., 2018) has been used to represent the species-formation-rate function as a neural network either in full, or in part, by adding the NODE outputs to an existing model (Thöni et al., 2025). However, naive usage of NODEs may result in models that do not respect physical laws, and for example, result in systems that allow concentrations of species to take negative values. To address this Ji and Deng (2021) (also see Nieves, Dandekar, and Rackauckas, 2024) set up a neural network architecture and non-linearities, encoding conservation of mass and Arrhenius' law, using log concentrations as inputs, the outputs as the derivative entries in the species-formation-rate function, and the intermediate weights interpretable as the reaction rate constants and stoichiometric coefficients. Therefore, their approach is a NODE with a specific architecture. The authors called their approach the chemical reaction neural network (CRNN) and it was tested on a few example chemical reaction networks.

Even though CRNN was able to infer the underlying data generating chemical reaction networks correctly, the validation of their approach contains several shortcomings. First of all, they assumed a situation where every species in the system is observed. As argued previously, this is seldom the case for more complex systems, and at times, completely unfeasible (e. g. for models of ion channel gating where the states are useful abstractions). Secondly, the example chemical reaction networks they tested CRNN on are rather relatively simple, they do not contain bidirectional reactions, feedback terms or more complex network motifs. Therefore, the CRNN approach, while promising, requires further validation.

Another approach popular in the SciML literature is physics-informed neural networks (PINNs, Raissi, Perdikaris, and Karniadakis, 2019; Raissi and Karniadakis, 2018). PINNs take a different approach to NODEs and instead of embedding a universal function approximator in an ODE system, take the states of the system x and time t as inputs, outputting variables u subject to the (soft) physical constraints of the system under consideration. For example, when chemical reaction networks are modelled using PINNs, the physical constraints arise from conservation of mass and mass-action kinetics (Bibeau, Boffito, and Blais, 2024); when wave propagation or flows are being modelled (Almanstötter, Vetter, and Iber, 2025; Wong et al., 2025), the constraints arise from the respective partial differential equations (Burger's and Navier-Stokes equations, for example).

PINNs have shown good performance in many challenging problems, with main uses being in fluid and solid mechanics, as well as earth sciences (Wong et al., 2025), however their application is not without difficulties. Most notably, owing to the linear combination of the loss terms due to errors in the data prediction and violation of physical constraints, PINNs can suffer from ill-conditioned and complex loss surfaces with many local optima (Krishnapriyan et al.,

2021) making them difficult to train. More recent approaches are starting to modify the way in which the data loss and constraint loss are being combined, leading to significant improvements in training (Almanstötter, Vetter, and Iber, 2025; Krishnapriyan et al., 2021). Further developments in PINN training methodologies that decrease the likelihood of the optimization getting stuck in local optima and promote compliance to the physical constraints may increase their appeal in modelling chemical reaction networks.

An additional way in which machine learning can be applied to the data from chemical reaction networks is symbolic regression (Cava et al., 2021) and sparse identification of dynamical systems (SINDY; Brunton, Proctor, and Kutz, 2016). The symbolic regression objective can be formulated via the following equation

$$\frac{d\mathbf{X}}{dt} = \Theta(\mathbf{X})\Xi \tag{27}$$

where $\mathbf{X}$ are the states of the system, $\Theta(\mathbf{X}) = [\mathbf{1}; \mathbf{X}; \mathbf{X}^2; ... \sin(\mathbf{X}); \cos(\mathbf{X}) ...]$ is the library of candidate functions and Ξ are the coefficients for the candidate functions, with the objective being to find a Ξ that fits the observed data best. Moreover, in sparse symbolic regression, a prior $p(\Xi)$ on the coefficients is usually introduced to induce sparsity in the functional bases, aiming for the most parsimonious fit. Considering chemical reaction networks, it is possible to set up a library $\Theta(\mathbf{X})$ corresponding to the mass action kinetics, which would generally scale with the number of species N as $\mathcal{O}(N^2)$ if we restrict it to bimolecular reactions. Using an appropriate prior Ξ may reduce the number of reactions to a sparse set. An additional approach where symbolic regression can be used is by first fitting a black box model, for example a neural network with strong regularization, and then fitting symbolic regression to the neural network, as done in Martensen et al. (2024). Such an approach essentially tries to interpret the neural network using the library of functions provided, but still benefits from the initial flexibility offered by the neural network. However, there are still many open questions before the symbolic regression techniques could be used in a streamlined manner. Sparse regression approaches do not perform as well under noise levels observed in biological experiments, can suffer from data undersampling (Prokop and Gelens, 2024), can display poor extrapolation qualities and may require different approaches when some of the variables in the system are not observed, as is common in most biological systems (Grigorian, George, and Arridge, 2024; Somacal et al., 2022).

As presented above, SciML has been a useful tool in a number of different fields. Therefore, it is also applied to the neuronal chemical reaction networks and models of voltage-gated ion channels in this thesis.

2.4 MACHINE LEARNING AND MIXED EFFECTS MODELS

The combination of mixed effects modelling and machine learning has already been fruitfully applied in other published work. In this section I will present some previous studies which combined these two modelling approaches, outlining the main implications for the present thesis. One of the first applications of mixed effects along with neural networks was done in Xiong, Kim, and Singh (2019), where the authors used a ResNet architecture (calling it $\Gamma(\cdot)$) together with a mixed effects framework, calling the resulting model mixed effects neural network (MeNet). Using a MeNet they processed image data X_n and then used the extracted features with the linear mixed effects framework where β are the fixed effects, u_n are the random effects and y_n is the dependent variable:

$$y_n = \Gamma(X_n)\beta + \Gamma(X_n)u_n \tag{28}$$

Since the gaze estimation data they used were obtained by repeated measurements from the same participants, taking into account these correlations between the data resulted in up to 20% improvements in errors over the state of the art. Note that their model is still linear because the fixed effects and the random effects are summed. Therefore, the authors demonstrate the benefit of combining a mixed effects framework for the prediction task where a neural network was used. Note that in this case the neural network did not process either the fixed or the random effects, it was rather used as a feature extractor, arguably not fully utilizing the power of the neural networks because the function captured by the neural network is not personalised, does not take random effects as its inputs.

The MeNet approach outlined in Xiong, Kim, and Singh (2019) has been generalized in Wörtwein et al. (2023). More specifically, instead of placing a neural network as a feature extractor (a fixed effect), they included random effects on the neural network weights, using both person-generic parameters $\bar{\theta}$, person-specific parameters θ_n and taking their sum

$$\theta = \bar{\theta} + \theta_n \tag{29}$$

as the neural network parameters, calling their approach neural mixed effects (NME). The authors used an assumption that the person-specific parameters come from a multivariate normal distribution with zero mean and covariance matrix Σ, which was used as a regularization term $\theta_n^T \Sigma^{-1} \theta_n$. NMEs were compared with MeNets from Xiong, Kim, and Singh (2019) and outperformed them on six different tasks. Therefore, the NME approach, individualization of the neural network weights, provided additional benefits to performance. However, caution should be used when allowing each parameter of a neural network its own random effect as that may make it easy to over-fit,

allowing the random effects to capture majority of the individuals variance, unless strong regularization on Σ is used.

One of the potential shortcomings of the NME approach is the over-parametrization of the random effects. Individual neural network parameters are likely to have very little influence on its overall behaviour, therefore allowing each parameter to have an associated random effect is inefficient both computationally, as well as conceptually. An alternative approach to the NMEs has been used in Martensen et al. (2024) (initially published in Rackauskas and Ivaturi, 2019). The approach they used, called DeepNLME, instead of putting a random effect on each parameter of the neural net $f_{NN}(\cdot)$, uses random effects η_n^{NN} as inputs to the neural network, in addition to its regular inputs x_0

$$\begin{aligned} x_0^* &= [x_0, \eta_n^{NN}] \\ f_n^{NN} &= f_{NN}(x_0^*) \end{aligned} \tag{30}$$

This way the neural net weights are fixed effects, yet the resulting functions f_n^{NN} are individualized for the nth observed entity. Moreover, usage of η_n^{NN} as inputs results in a more efficient representation of the random effects than the addition of random effects on each parameter done in NMEs, since the random effect inputs percolate throughout the whole neural network, affecting the function that it represents. Therefore, such usage of random effects is more computationally efficient and likely much less prone to over-fitting via the random effects, as there is simply not as many degrees of freedom as in, for example, NMEs.

It is worth noting that the field of combining mixed effects frameworks with machine learning is still young and therefore there are many different competing approaches with best practices and standards still not yet fully developed. Moreover, neural networks are not the only machine learning model that has been combined with the mixed effects frameworks. For example, another popular machine learning model that was used in many publications in combination with mixed effects is the random forests (Hajjem, Bellavance, and Larocque, 2011, 2014; Hajjem, Larocque, and Bellavance, 2017; Hu et al., 2023). Concluding, the integration of a mixed effects framework and a machine learning approach makes it possible to take advantage of the hierarchical structure of the data with multiple studies showing performance gains over approaches that do not take the structure of the data into account. Therefore, this combination of approaches offers a powerful means of tackling many real world challenges more effectively.

Part II

APPLICATION OF NLME AND SCIML IN COMPUTATIONAL NEUROSCIENCE

If you can dream – and not make dreams your master;
If you can think – and not make thoughts your aim;
If you can meet with Triumph and Disaster
And treat those two impostors just the same
If you can bear to hear the truth you've spoken
Twisted by knaves to make a trap for fools,
Or watch the things you gave your life to, broken,
And stoop and build 'em up with worn-out tools:

Excerpt from "*If*" by Rudyard Kipling, 1895

3

FITTING Ca^{2+}-CALMODULIN CHEMICAL REACTION NETWORKS

This chapter tackles the challenge of fitting a chemical reaction network in a situation where there is, arguably, a large amount of data, as well as a significant amount of knowledge about the possible structures that the chemical reaction network can adopt. However, in addition to these two factors, there is uncertainty in experimental conditions, necessitating the nonlinear mixed effects (NLME) modelling approach.

One of the main motivations for the work undertaken in this chapter is the importance of the Ca^{2+}-Calmodulin cascade in synaptic plasticity, occuring in almost two thirds of the current models (see Figure 4 in Heil et al., 2018). A good understanding of the relative merits of the different Ca^{2+}-Calmodulin models is important, especially when they are used in a larger model of the synaptic chemical reaction network. This understanding was lacking due to an absence of comparison between the different Ca^{2+}-Calmodulin models used in different publications.

The bulk of this chapter is formed by the included publication: Linkevicius, D., Chadwick, A., Faas, G. C., Stefan, M. I., & Sterratt, D. C. (2025) 'Fitting and comparison of calcium-calmodulin kinetic schemes to a common data set using non-linear mixed effects modelling' PLoS ONE, 20. The publication is followed by the basic supplemental information that was included with the publication, where some parts have been extended with a more in-depth discussion of the results included.

The published results showcase which published Ca^{2+}-Calmodulin models are not able to adequately fit the data from Faas et al. (2011) and Shifman et al. (2006), derives new and improved parameter sets for a number of different Ca^{2+}-Calmodulin chemical reaction network structures, as well as delves into the structural aspects which are necessary for a Ca^{2+}-Calmodulin model to be able to adequately fit the data. This work forms a solid basis on which further work including other molecular species, could be based on, namely investigating molecules like CaMKII (Shifman et al., 2006), various phosphodiesterases (Goraya and Cooper, 2005) and others.

PLOS ONE

RESEARCH ARTICLE

Fitting and comparison of calcium-calmodulin kinetic schemes to a common data set using non-linear mixed effects modelling

Domas Linkevicius[1], **Angus Chadwick**[1], **Guido C. Faas**[2], **Melanie I. Stefan**[3], **David C. Sterratt**[1]*

1 Institute for Adaptive and Neural Computation, School of Informatics, University of Edinburgh, Edinburgh, United Kingdom, **2** Department of Neurology, School of Medicine, University of California Los Angeles, Los Angeles, California, United States of America, **3** Faculty of Medicine, Medical School Berlin, Berlin, Germany

* David.C.Sterratt@ed.ac.uk

OPEN ACCESS

Citation: Linkevicius D, Chadwick A, Faas GC, Stefan MI, Sterratt DC (2025) Fitting and comparison of calcium-calmodulin kinetic schemes to a common data set using non-linear mixed effects modelling. PLoS ONE 20(2): e0318646. https://doi.org/10.1371/journal.pone.0318646

Editor: Pan Li, Institute for Basic Science, KOREA, REPUBLIC OF

Received: November 1, 2024

Accepted: January 21, 2025

Published: February 7, 2025

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pone.0318646

Data Availability Statement: https://doi.org/10.6084/m9.figshare.c.7614467.v1.

Abstract

Calmodulin is a calcium binding protein that is essential in calcium signalling in the brain. There are many computational models of calcium-calmodulin binding that capture various calmodulin features. However, existing models have generally been fit to different data sets, with some publications not reporting their training and validation performance. Moreover, there is no model comparison using a common benchmark data set as is common practice in other modeling domains. Finally, some calmodulin models have been fit as a part of a larger kinetic scheme, which may have resulted in parameters being underdetermined. We address these three limitations of previous models by fitting the published calcium-calmodulin schemes to a common calcium-calmodulin data set comprising equilibrium data from Shifman et al. and dynamical data from Faas et al. Due to technical limitations, the amount of uncaged calcium in Faas et al. data could not be predicted with certainty. To find good parameter fits, despite this uncertainty, we used non-linear mixed effects modelling as implemented in the `Pumas.jl` package. The Akaike information criterion values for our reaction rate constants were significantly lower than for the published parameters, indicating that the published parameters are suboptimal. Moreover, there were significant differences in calmodulin activation, both between the schemes and between our reaction rate and those previously published. A kinetic scheme with independent lobes and unique, rather than identical, binding sites fit the data best. Our results support two hypotheses: (1) partially bound calmodulin is important in cellular signalling; and (2) calcium binding sites within a calmodulin lobe are kinetically distinct rather than identical. We conclude that more attention should be given to validation and comparison of models of individual molecules.

Introduction

Calmodulin is among the most important calcium binding proteins in the brain. It is essential in the translation of intracellular Ca^{2+} signals to downstream processes, such as gene

PLOS ONE

Comparison of calcium-calmodulin kinetic schemes to a common data set

Funding: Domas Linkevicius was funded by a PhD stipend by the United Kingdom Research and Innovation (grant EP/S02431X/1), UKRI Centre for Doctoral Training in Biomedical AI at the University of Edinburgh, School of Informatics. Guido C. Faas was funded by NIH grants NS027528 and NS030549 and The Wendy and Leonard Goldberg Endowment to UCLA. David C. Sterratt, Melanie I. Stefan and Angus Chadwick did not receive any specific funding for this work.

Competing interests: The authors have declared that no competing interests exist.

regulation, protein activation, metabolic regulation and synaptic plasticity [1, 2]. Calcium signals can provide information both via their amplitude (nanomolar to millimolar) and via their duration (microseconds to hours) [1, 2]. Calcium-calmodulin binding kinetics underlie the translation of Ca^{2+} signals, therefore correct kinetic models of binding are an important aspect in studying calcium signalling in the brain.

Calmodulin's signalling properties arise from its structure—it comprises a 148 amino acid residue polypeptide with four EF hands divided into C and N lobes capable of binding two calcium ions per lobe [3]. It can adopt many conformational states, especially when bound to different molecules [4]. Moreover, calmodulin lobes have been reported to differ in their kinetics and affinity for Ca^{2+}–the N lobe binding faster with lower affinity and the C lobe binding slower with higher affinity [5–7].

There are at least 19 published computational models of synapses that include various models of calmodulin in their chemical reaction network [8]. The published kinetic schemes describing calcium-calmodulin binding vary significantly in the number of calmodulin features they capture. For example, some calmodulin models do not have independent lobes [9–11] while others do [5, 12, 13]. Some schemes are event-based—only concerned about the Ca^{2+} binding events [10, 11], whereas others explicitly indicate which lobe and/or site is being bound to [5, 12, 13]. Moreover, some models assume that two Ca^{2+} ions bind to a lobe at the same time [9, 10, 12], others leave this dependent on reaction rate constants [5, 11, 13]. Some models assume that Ca^{2+} binding sites within a lobe are unique [11, 13] while others assume that they are non-unique [5, 12]. Finally, some models include details such as calmodulin conformational states [14].

Most current computational calmodulin models suffer from three limitations. First of all, different models have generally been tuned to different data sets, making their relative performance difficult to compare. Secondly, most models have not been cross-validated, making their generalization performance uncertain. Thirdly, some models have been tuned as a part of a larger scheme, e.g. including CaMKII, potentially making calcium-calmodulin binding parameters underdetermined. We discuss the sources of data to which calcium-calmodulin models we investigate have been fit in the Methods section, therefore we will next elaborate on the second and the third limitations.

The second limitation relates to cross-validation, a crucial step in the process of parameter inference used to establish model performance outside of the training data and to avoid overfitting (see page 241 in [15]). Ideally different models are cross-validated on a single data set across publications using consistent quantitative metrics. For example, the MNIST data set [16] is used to compare the error rate of image processing models. Given the lack of rich open access calmodulin data sets, none of the published calmodulin models were quantitatively cross-validated during development. At best, publications that contain calmodulin kinetic schemes include some indication (usually visual, rather than quantitative) of performance compared to the source of data they are being tuned to. However, this is not a rigorous way of ensuring that a model will perform well outside of the training data, leaving the generalization performance uncertain.

The third limitation is that some calmodulin schemes have been tuned to data from experiments that include other calmodulin-binding molecules, with large numbers of reaction rate constants that have to be fit, for example the calcium-calmodulin-CaMKII cascade [13]. Systems biology models naturally exhibit sloppiness [17], which tends to get more pronounced with an increasing number parameters being fit, resulting in loosely constrained parameter values. It is often possible to trade off between reaction rate constants: a calcium-calmodulin-CaMKII cascade being fit to CaMKII activity measurements may fit data better if Ca^{2+} binds to calmodulin with higher affinity or calcium-calmodulin binds to CaMKII with higher

affinity, or some mix of the two. Moreover, a similar trade-off is possible between the binding sites and/or lobes even with only a calcium-calmodulin cascade. Because of these trade-offs, the true reaction rate constants might be completely different to the ones obtained via the fitting procedures. Some publications attempt to test for such parameter sloppiness via sensitivity analyses [18] or by calculating the eigenvalues of the Hessian [17], but it usually is too difficult to test the parameter combinations in a sufficiently dense and wide manner to ensure that the reaction rate constants are not under-determined.

Calmodulin models, in particular some of the simpler ones [9, 12], have been used to investigate calmodulin interactions with other molecules [18–20] and in complex chemical reaction networks [9, 21–23] to model higher order phenomena occurring in neurons, e.g. synaptic plasticity. However, given the aforementioned model limitations, it is important to scrutinize the previous modelling work, its basic assumptions, and to check whether the assumptions made in previous work hold when tested under more rigorous conditions, with powerful methods using richer data sets.

We address the three aforementioned limitations of existing calmodulin models by using a common data set where the only free kinetic parameters are calcium-calmodulin binding reaction rate constants. The common data set comprises subsets of data from Faas et al. [5] and Shifman et al. [11]. Faas et al. [5] contains time-series of fluorescence measurements after laser-induced Ca^{2+} uncaging and therefore is informative about calmodulin dynamics. In contrast, Shifman et al. [11] contains measurements of calmodulin properties at equilibrium. To deal with incomplete experimental control of the amount of calcium uncaged by a laser flash in Faas et al. [5] we use the novel and highly efficient non-linear mixed effects (NLME) model fitting algorithms implemented in `Pumas.jl` [24]. NLME is a hierarchical modeling framework that can deal with phenomena where there are constant intra-individual parameters, but significant inter-individual variability due to individual level parameters [24, 25].

We use the common data set to fit reaction rate constants from scratch and compare our results to the reaction rate constants in the literature. By calculating the Akaike information criterion (AIC) [26] values for both our and the published reaction rate constants we show that the published reaction rate constants are suboptimal. Moreover, using the same criterion, we show that some kinetic schemes are suboptimal and fail to fit calmodulin dynamics and equilibrium behaviour at the same time. We then compare the Ca^{2+} signal integration properties of different calmodulin schemes when either the published reaction rate constants or the ones determined by our approach are used. We show that there are significant differences in calmodulin calcium integration properties when using the suboptimal published reaction rate constants. Similarly, we show that the models using suboptimal calmodulin schemes display qualitatively different calcium integration behaviour compared to better performing schemes. Finally, we calculate the partial rank correlations between the reaction rate constants that we fit and show that for some calmodulin schemes our parameter fits are highly correlated which is indicative of parameter sloppiness or underdetermination.

Our results highlight that a sufficiently expressive calmodulin model structure is essential for capturing both calmodulin dynamics and equilibrium behaviour. Moreover, we conclude that, given the suboptimality of the previously published parameter sets, arguments and findings built on these models may warrant re-visiting.

Methods

Data

Dynamical calmodulin data. We use the calcium uncaging data from Faas et al. [5], in which different concentration mixes of the fluorescent Ca^{2+} indicator OGB-5N, the light

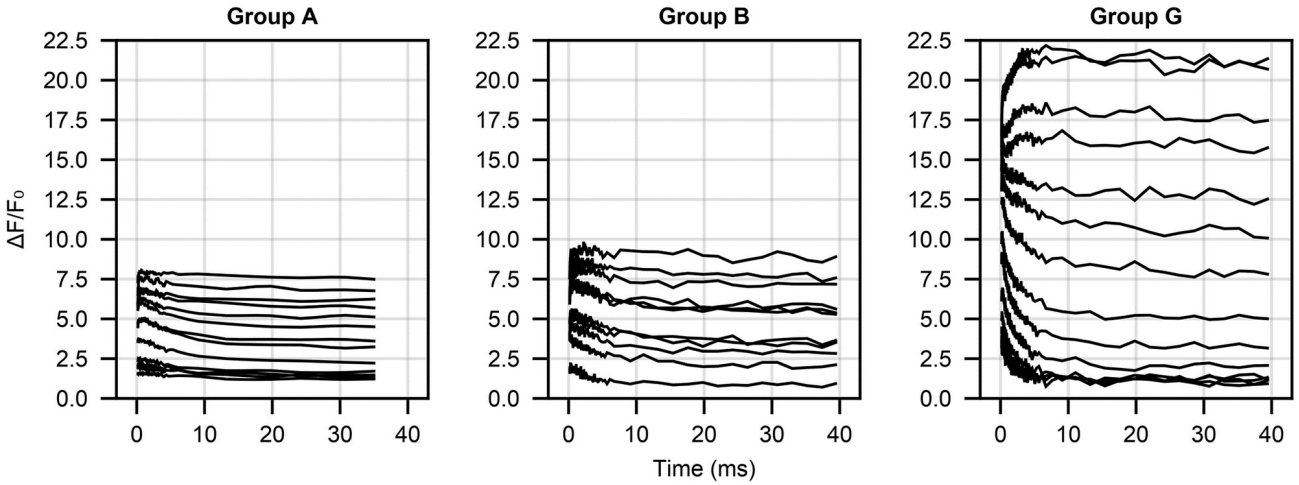


Fig 1. Relative fluorescence ($\Delta F/F_0$) time-series data from [5] for three different initial condition groups (A, B and G). Different lines within a plot are due to different laser uncaging strength (the higher the laser strength, the larger the $\Delta F/F_0$ value).

https://doi.org/10.1371/journal.pone.0318646.g001

sensitive Ca^{2+} chelator DM-nitrophen (DMn), calmodulin and titrated free Ca^{2+} were used to make seven different groups of solutions A–G (see S2 Appendix for specific concentrations). For different batches of each group of solutions, a sequence of laser pulses of increasing strength was used to induce Ca^{2+} uncaging from DMn while OGB-5N fluorescence was observed at 35˚C. The stronger the pulse, the more calcium is released. Due to technical limitations, it was not possible to predict the amount of released calcium for each laser pulse precisely. We elaborate on how we model the fraction of uncaged Ca^{2+} below. Fig 1 shows the fluorescence time courses for three of the seven groups—A, B and G—and different uncaging laser strengths. We use a subset of the data and split it into training, validation and test data sets (see Supplemental Text S1 Appendix for more information).

Calmodulin equilibrium data. Steady-state calcium-calmodulin binding came from an experiment in Shifman et al. [11], which measured the number of Ca^{2+} ions bound per calmodulin molecule at different free Ca^{2+} concentrations (their Fig 1B). Their experimental chamber contained a fluorescent indicator Fluo4FF (5μM), calmodulin (5μM) and a varying amount of free Ca^{2+}. The amount of free Ca^{2+} was titrated until a required concentration (between approximately 10^{-7}M and 5.5×10^{-5}M) was reached. We used a digital tool (https://automeris.io/WebPlotDigitizer/) to extract this data from their plots, giving the 107 points shown in Fig 2. This data was obtained at 25˚C but calmodulin does not show significant temperature dependent changes in equilibrium behaviour [27], so we do not adjust for temperature dependent changes in calmodulin kinetics.

Kinetic schemes and published reaction rate constants

We investigated six different calcium-calmodulin binding schemes from the literature that span the complexity of the most commonly used calmodulin models (Fig 3). The reference we give for a scheme may be its original source, or a source that is frequently cited for the scheme. There are more complex published calmodulin schemes that we did not use [14], because they would be prohibitively computationally expensive to fit.

The simplest scheme (Scheme 1, Fig 3), from Kim et al. [9], is made up of three calmodulin states—CaM0, CaM2Ca, CaM4Ca, respectively calmodulin bound to no, two and four Ca^{2+} ions. In Scheme 1, Ca^{2+} binding is assumed to be highly co-operative and binding of two Ca^{2+}

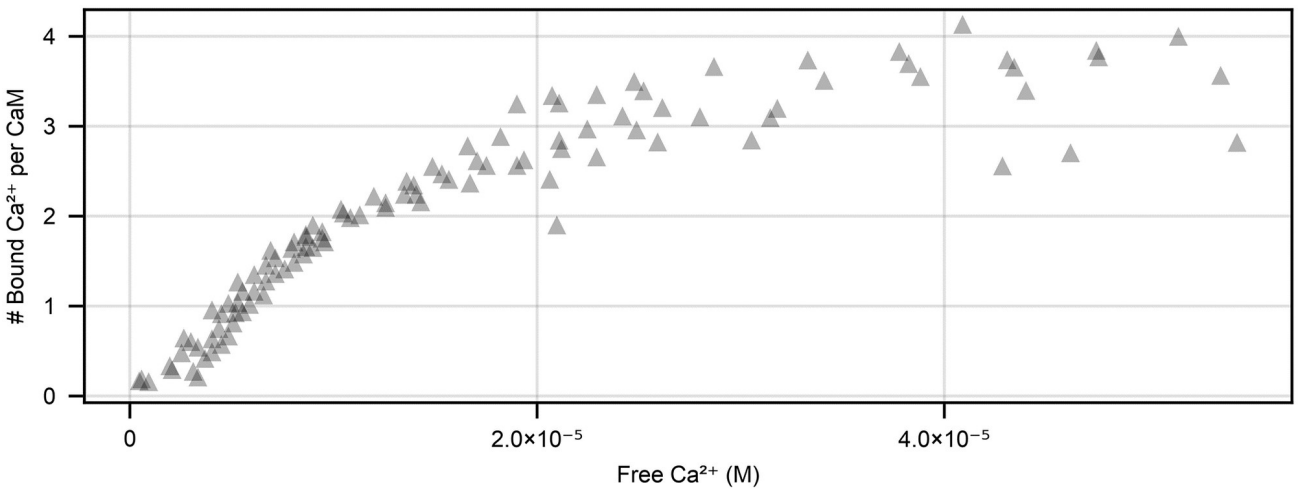


Fig 2. Equilibrium measurements of the number of Ca^{2+} ions per calmodulin molecule from [11]. Experiments were done using 5μM Fluo4FF and 5μM calmodulin.

https://doi.org/10.1371/journal.pone.0318646.g002

ions is treated as a single reaction. In principle Scheme 1 does not assume which lobe binds first; the first two Ca^{2+} ions could bind to the C lobe or the N lobe. However, the parameterisation of Scheme 1 by Kim et al. [9] implies that they treat the first Ca^{2+} binding event as being to the C lobe. The published reaction rate constants in Kim et al. [9] are based on stopped-flow fluorescence measurements [7]. Scheme 1 is parameterised by four reaction rate constants $\{k_i\}_{i=1}^4$, which can be used to derive two dissociation constants: $K_{D_1} = \frac{k_2}{k_1}$ and $K_{D_2} = \frac{k_4}{k_3}$.

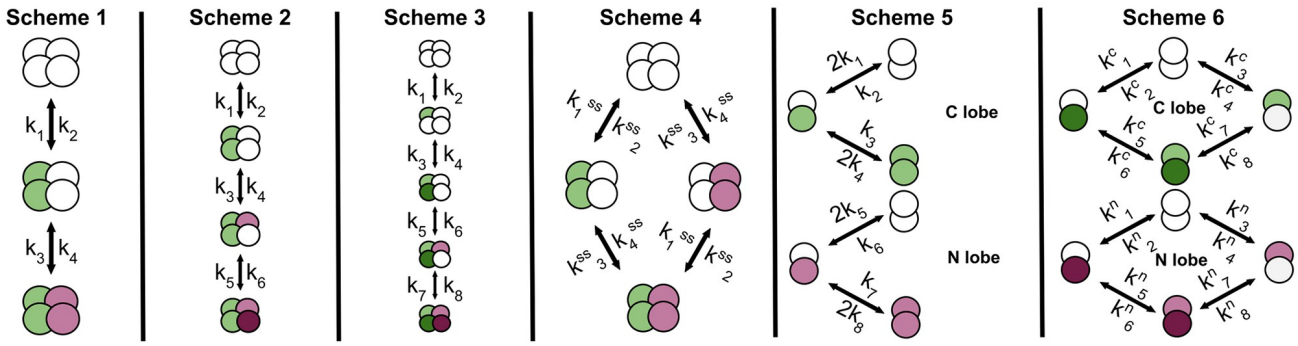


Fig 3. Six calmodulin kinetic schemes to which we fit parameters, and compare to performance with published parameter values. Scheme 1—due to strong co-operativity, each calmodulin lobe binds two Ca^{2+} ions at a time, with the lobes modelled sequentially (first C lobe then N lobe). Scheme 1 is parameterised by reaction rate constants $\{k_i\}_{i=1}^4$. **Scheme 2**—due to co-operativity the first reaction has two Ca^{2+} ions binding as a single event and then the next two Ca^{2+} ions binding sequentially. It is parameterised by reaction rate constants $\{k_i\}_{i=1}^6$. **Scheme 3**—fully expanded sequential calmodulin scheme where each binding event is represented individually. Depending on the reaction rate constants, the binding events could be mixed between the lobes, e.g. first binding event could be in the C lobe, the second in the N lobe, or partial combinations of different lobes. The visualised scenario is where the first two events are in the C lobe. This scheme is parameterised by reaction rate constants $\{k_i\}_{i=1}^8$. **Scheme 4**—calmodulin binds two Ca^{2+} ions at a time and, contrary to **Schemes 1–3**, the lobes are independent. It is parameterised by eight reaction rate constants $\{k_i\}_{i=1}^8$ which, along with free Ca^{2+}, are used to calculate the effective reaction rate constant $\{k_i^{ss}\}_{i=1}^4$ (see S3 Appendix for more details). **Scheme 5**—this scheme has independent N and C lobes, with a single Ca^{2+} ion binding at a time. Binding sites within a single lobe are identical. It is parameterised by reaction rate constants $\{k_i\}_{i=1}^8$. **Scheme 6**—this scheme has independent N and C lobes, with a single Ca^{2+} ion binding at a time. In contrast to Scheme 5, the binding sites within a single lobe are distinct (indicated by different shades of green/purple). The scheme is parameterised by 16 reaction rate constants $\{k_i^{c/n}\}_{i=1}^8$. In all schemes green circles indicate Ca^{2+}-bound C lobe sites, purple circles indicate Ca^{2+}-bound N lobe sites and arrows indicate bidirectional reactions (Ca^{2+} ions not shown).

https://doi.org/10.1371/journal.pone.0318646.g003

The next scheme (Scheme 2, Fig 3), from Bhalla and Iyengar [10], is made up of four calmodulin states—CaM0, CaM2Ca, CaM3Ca, CaM4Ca, respectively calmodulin bound to no, two, three and four Ca^{2+} ions. In Scheme 2, binding of the first two Ca^{2+} ions is assumed to be highly co-operative and treated as a single reaction, whereas the next two Ca^{2+} ions bind individually. The parameters in Bhalla and Iyengar [10] (as given in https://doqcs.ncbs.res.in/, also see [28]) do not match neatly to either lobe and the description of how the rate constants were derived was unavailable at the time of writing. Scheme 2 is parameterised by six reaction rate constants $\{k_i\}_{i=1}^{6}$, which can be used to derive three dissociation constants: $K_{D_1} = \frac{k_2}{k_1}$, $K_{D_2} = \frac{k_4}{k_3}$ and $K_{D_3} = \frac{k_6}{k_5}$.

The final linear scheme that ignores calmodulin lobe-based structure (Scheme 3, Fig 3) is from Shifman et al. [11]. It comprises five calmodulin states—CaM0, CaM1Ca, CaM2Ca, CaM3Ca, CaM4Ca—respectively calmodulin bound to no, one, two, three and four Ca^{2+} ions. The dissociation constants based on experiments in Shifman et al. [11] are 7.9μM, 1.7μM, 35μM, 8.9μM respectively for Ca^{2+} binding events one to four. Reactions in this scheme do not neatly map onto individual Ca^{2+} binding sites within calmodulin lobes; instead they are abstract binding events where, depending on the parameters, they may be probabilistic combinations between different binding sites. Scheme 3 is parameterised by eight reaction rate constants $\{k_i\}_{i=1}^{8}$, which can be used to derive four dissociation constants $K_{D_1} = \frac{k_2}{k_1}$, $K_{D_2} = \frac{k_4}{k_3}$, $K_{D_3} = \frac{k_6}{k_5}$ and $K_{D_4} = \frac{k_8}{k_7}$. This scheme is the most complex linear CaM scheme possible (without adding conformational calmodulin changes), modelling each Ca^{2+} binding site individually.

Our Scheme 4 (Fig 3) is from Pepke et al. [12] and comprises four states—CaM0, CaM2C, CaM2N, CaM4Ca—respectively calmodulin bound to no Ca^{2+} ions, two at the C lobe, two at the N lobe and four across both lobes. It is the simplest scheme that captures the lobe-based structure of calmodulin. It has eight reaction rate constants $\{k_i\}_{i=1}^{8}$ and is based on Scheme 5 (described below), but was simplified used a quasi-steady state approximation for calmodulin species that have a single bound Ca^{2+} ion. This approximation results in elimination of partially bound species from simulations by setting their derivatives to 0 and expressing the partially bound species in terms of the unbound and the fully bound species and permitting the appropriate substitutions in the equations for the unbound and the fully bound species (see S4 Appendix).

We draw our Scheme 5 (Fig 3) from model 1 in Pepke et al. [12], which is identical to the scheme used in [5]. It is made up of nine states—CaM0, CaM1C, CaM1N, CaM2C, CaM2N, CaM1C1N, CaM2C1N, CaM1C2N, CaM4—with the number of Ca^{2+} ions bound to calmodulin indicated by numbers preceding C and N. Even though in total there are nine states, since in this study calmodulin does not bind to any downstream species, we do not need to track individual calmodulin molecules. Therefore, we simulate the lobes as independent species which decreases the number of states we need to track from nine to six—CaM0N, CaM1N, CaM2N, CaM0C, CaM1C, CaM2C—without changing the scheme itself. Scheme 5 is parameterised by eight reaction rate constants $\{k_i\}_{i=1}^{8}$, which can be used to derive four dissociation constants $K_{D_1} = \frac{k_2}{k_1}$, $K_{D_2} = \frac{k_4}{k_3}$, $K_{D_3} = \frac{k_6}{k_5}$ and $K_{D_4} = \frac{k_8}{k_7}$. Pepke et al. [12] used two data sources on calmodulin equilibrium behavior: (1) data from wild-type and tryptic calmodulin fragments (one lobe expressed, other eliminated) [6]; (2) data from competition assays (calmodulin, either wild type or mutants with one active and one inactive lobe, and fluorescent indicator Fluo4FF) [11]. Pepke et al. [12] (in their supplemental information) give reaction rate constants as ranges—we take specific numerical values from this model's entry (model identifier: MODEL1001150000) in the BioModels Database [29, 30]. Faas et al. [5] tuned the model to their own UV-flash photolysis data.

Finally, our Scheme 6 (Fig 3) is from Byrne et al. [13]. It is made up of sixteen states, but similar to Scheme 5, we simulate the lobes as independent species which reduces the number of states to eight—CaM0N, $CaMN_1$, $CaMN_2$, CaM2N, CaM0C, $CaMC_1$, $CaMC_2$, CaM2C, where CaM0X denotes an unbound calmodulin lobe, $CaMX_1$ denotes Ca^{2+} bound to the first site of a lobe, $CaMX_2$ denotes Ca^{2+} bound to the second site of a lobe and CaM2X denotes a fully bound lobe. This scheme is parameterised by 16 reaction rate constants $\{k_i^{c/n}\}_{i=1}^{8}$, which can be used to derive eight dissociation constants $K_{D_1}^{c/n} = \frac{k_2^{c/n}}{k_1^{c/n}}$, $K_{D_2}^{c/n} = \frac{k_4^{c/n}}{k_3^{c/n}}$, $K_{D_3}^{c/n} = \frac{k_6^{c/n}}{k_5^{c/n}}$ and $K_{D_4}^{c/n} = \frac{k_8^{c/n}}{k_7^{c/n}}$. Reaction rate constants in Byrne et al. [13] are based on stopped-flow fluorescence and competitive binding assay data [31].

For each of the six schemes we use the reaction rate constants from the associated publication (we use both Pepke et al. [12] and Faas et al. [5] for Scheme 5). All of the reaction rate constants we used are given in S4 Appendix.

Ca^{2+} uncaging model

Faas et al. [5] used a linear model for laser induced Ca^{2+} uncaging

$$U(\mathrm{PCD}, x) = 0.0011 \times \mathrm{PCD} - 0.39 + x \tag{1}$$

where U is the uncaged DMn fraction, PCD is the specified Pockels cell delay, where a larger value results in a higher energy laser pulse and more Ca^{2+} uncaging, and x is used to account for the uncertainty of the actual PCD value, i.e. the difference between the specified and physically realised values.

Even though Faas et al. [5] used the linear model successfully in their study, its performance is not quantified and it has some limitations. Most importantly, U is not bounded to [0, 1] and can take negative values or values above 1, which would result in physically unrealistic initial conditions. Moreover, it is not clear that a simple linear relationship is optimal to accurately model the relationship between the PCD value and the uncaging fraction. Finally, the variable x is additive, and it is not clear that this formulation is optimal—it could be multiplicative or some more complex functional relationship.

Due to lack of the necessary data, we could not develop our own model of how the fraction of calcium uncaged depends on the PCD. Instead, of the linear model (Eq 1) we use an even simpler model that performed better in practice than either the linear model from Faas et al. [5] or a neural network. Our uncaging model does not take the specified PCD value and is a simple sigmoid function, bounding its output to [0, 1].

$$U(x) = \frac{1}{1 + \exp^{-x}} \tag{2}$$

With this approach, we do not claim that uncaging is completely independent of PCD; rather we use a single equation to capture both uncertainty in estimating the PCD, as well as other sources of variance.

Fitting our reaction rate constants

We combine and adapt the definitions and notation of NLME provided in [24, 32] and we present it using Scheme 1 as an example. NLME modeling framework comprises a two level hierarchical structure (shown visually in Fig 4) with fixed effects Θ at the upper level, which can be broadly grouped into

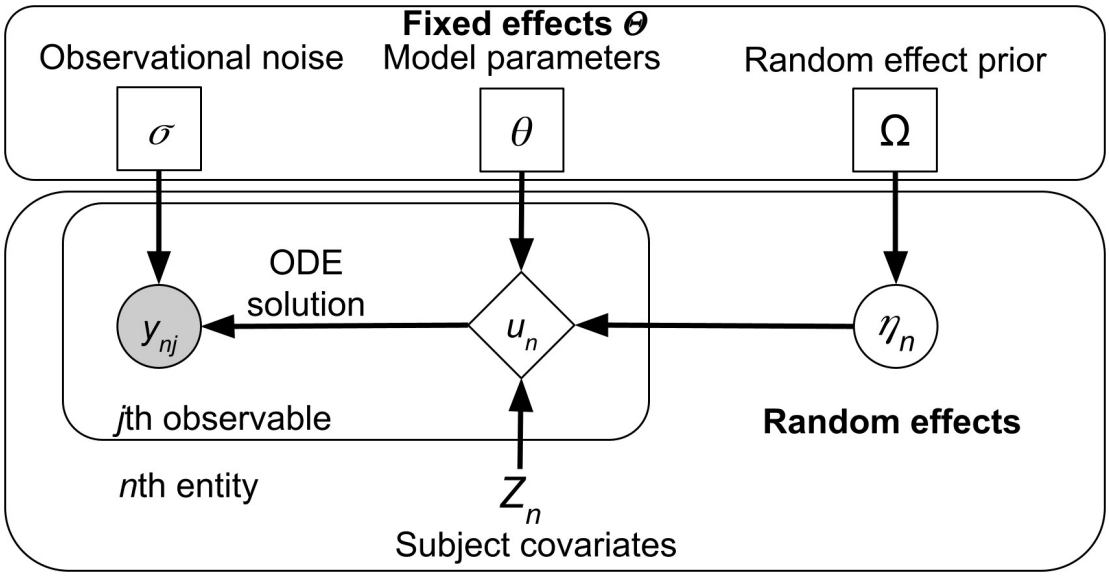


Fig 4. Visual representation of an NLME model, rectangle nodes in the top box denote parameters (fixed effects), circles denote random quantities which are either latent (unfilled) or observed (filled), diamonds are deterministic given the inputs, and nodes without a border are constant. Each symbol in the node can be either a number or a vector depending on the context.

https://doi.org/10.1371/journal.pone.0318646.g004

- model parameters θ, e.g. for scheme 1—reaction rate constants $\{k_i\}_{i=1}^{4}$
- random effect prior distribution parameters Ω, e.g. μ and ω used to parameterize the prior of η_n for the Ca^{2+} uncaging fraction (see paragraph below)
- observation model noise parameters σ

and do not vary between recordings.

The lower level is random effects η_n which account for the inter-individual variability of the observations y_{nj}, generally, by individualizing model parameters θ. We assume that both Ca^{2+} and calmodulin molecules are identical between experiments, hence reaction rate constants do not have the random effect-enabled individualization for each experimental run. The sole usage of random effects in this paper is in fitting the fraction of uncaged Ca^{2+} by passing $\eta_n \sim \mathcal{N}(\mu, \omega)$ to Eq 2.

Furthermore, there is a set of covariates Z_n associated with each recording, i.e. the total concentration of calmodulin, Ca^{2+} and OGB5N, which are known. These three sets of values are collated via the parameter model g into the dynamical parameter vector p_n of the nth recording

$$p_n = g(\Theta, Z_n, \eta_n) \tag{3}$$

The dynamical parameters p_n are then fed into the structural model (e.g. an ordinary differential equation (ODE) system)

$$u_n' = f(u_n, p_n, t) \tag{4}$$

where u are the dynamical variables being solved for (DMn, OGB5N, Ca^{2+} and their combinations and the various calmodulin species determined by the scheme being used). For Scheme 1 the system of equations (for brevity omitting DMn, OGB5N, Ca^{2+} and means of its input

amplitude, frequency and duration into the model) would be

$$\begin{aligned} \mathrm{d}[\mathrm{CaM0}]/\mathrm{dt} &= -k_1[\mathrm{Ca}]^2[\mathrm{CaM0}] + k_2[\mathrm{CaM2Ca}] \\ \mathrm{d}[\mathrm{CaM2Ca}]/\mathrm{dt} &= k_1[\mathrm{Ca}]^2[\mathrm{CaM0}] - k_2[\mathrm{CaM2Ca}] \\ &\quad -k_3[\mathrm{Ca}]^2[\mathrm{CaM2Ca}] + k_4[\mathrm{CaM4Ca}] \\ \mathrm{d}[\mathrm{CaM4Ca}]/\mathrm{dt} &= k_3[\mathrm{Ca}]^2[\mathrm{CaM2Ca}] - k_4[\mathrm{CaM4Ca}] \end{aligned} \tag{5}$$

Note that $\{k_i\}_{i=1}^4$ enter into Eq 4 via p_n, which can also be used to initialize the ODE system.

The final step is to link the numerical solution of the ODE system to the experimentally observed quantities. The jth observable quantity y_{nj} for the nth entity is calculated using the simulated variables $u_n(t)$ and the times t_m at which the observations were made via the observational model h

$$y_{nj}(t = t_m) = h_j(u_n(t = t_m), p_n, Z_n, \eta_n) \tag{6}$$

In this study there are two observable quantities: the relative fluorescence $\Delta F/F_0$ over time being fit to recordings from Faas et al. [5] and Ca^{2+} per calmodulin at equilibrium being fit to Shifman et al. [11]. $\Delta F/F_0$ is derived from OGB5N as follows

$$\Delta F/F_0(t = t_m) = \frac{[\mathrm{OGB5N}](t = t_m) + (F_{\max}/F_{\min})[\mathrm{CaOGB5N}](t = t_m)}{[\mathrm{OGB5N}](t = 0) + (F_{\max}/F_{\min})[\mathrm{CaOGB5N}](t = 0)} \tag{7}$$

where $F_{\max}/F_{\min} = 39.364$ [5] and $F_{\max}$, $F_{\min}$ are maximal and minimal recorded fluorescence values. Ca^{2+} per calmodulin is simply the sum of calmodulin species multiplied by the number of bound Ca^{2+} ions for each species divided by total calmodulin. After obtaining the observable quantities a Gaussian observation model is used to account for observational noise.

There are many ways to fit NLME models, both frequentist and Bayesian [33]. In this study we use the maximum aposterior (MAP) conditional log-likelihood objective which can be stated as

$$\Theta^*, \eta^* = \arg\max_{\Theta,\eta} \left(p(\Theta) \cdot \prod_{i=1}^{N} p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta) \right) \tag{8}$$

where Θ^* is the mode of the fixed effects, η^* is the mode of the random effects for each subject and $p(\Theta)$ is the fixed effect prior distribution. Conditional likelihood is much more numerically efficient due to Θ and η_n being optimized jointly whereas, for example, marginal likelihood generally requires a two level optimization scheme and Markov Chain Monte Carlo requires many more likelihood evaluations due to sampling. However, conditional likelihood requires appropriate handling (either fixing or priors, see next section) of Ω to avoid overly broad random effect distributions which barely penalize extreme η_n values and effectively result in different individual models due to the learning being offloaded mostly to the random effects.

We use the `Pumas.jl` [24] Julia package to fit to solve Eq 8. `Pumas.jl` contains efficient and powerful algorithms for NLME modelling, which was essential when fitting the ηs used to model the uncaging fraction. Specifically, we used the BFGS optimization algorithm from `Optim.jl` with the gradient calculations handled by `Pumas.jl`.

All fitting was done on the JuliaHub (https://juliahub.com/) cloud computing platform using nodes with 8 vCPUs and 64GB of memory. Individual fits took between one to ten minutes, depending on which scheme was used and whether some of the parameters were fixed.

For each scheme we conducted 20 fitting runs with different initial conditions. All the code that was used to define the models, run the simulations and perform the analysis is accessible at https://github.com/dom-linkevicius/FaasCalmodulin.jl.git, the data is accessible at [34].

Prior distributions

We incorporated the existing knowledge about calmodulin reaction rate constants for different kinetic schemes via per-scheme prior distributions that depend on the amount data available for each scheme. All of our priors are in $\log_{10}$ space as optimizing rate constants in log-space was more performant.

For Schemes 1 and 2, since they are simplified and contain fewer reaction rate parameters than an actual calmodulin molecule would, mapping from experimental data to reaction rate constants is difficult. Therefore, we opted to use wide uniform priors that reflects the small amount of available prior information: $\mathcal{U}(2, 9)$ for the forward reaction rate constants (corresponds to 10^2 $M^{-2}ms^{-1}$ to 10^9 $M^{-2}ms^{-1}$) and $\mathcal{U}(-9, -4)$ for the dissociation constants (corresponds to a range of 1 nM^2 to 10 μM^2).

For Scheme 3, which models Ca^{2+} binding events individually, there is a significant amount of prior information. Specifically, we use set priors on the dissociation constants based on Shifman et al. [11]. We used priors of the form $\mathcal{N}(r, 1)$, where r is a dissociation constant from Shifman et al. [11] Table 2 in $\log_{10}$. Unfortunately, setting a prior that could similarly constrain the forward reaction rate constants was not possible, therefore we again opted for wide uniform priors that we used for Schemes 1 and 2: $\mathcal{U}(2, 9)$.

For Schemes 4 and 5, since they share the same set of reaction rate constants, we used the same set of prior distributions. For each forward reaction rate and dissociation constant we used $\mathcal{N}(r, 1)$, where r are reaction rate constants from Faas et al. [5] in $\log_{10}$. We chose Faas et al. [5], rather than Pepke et al. [12], because their rate constants are based on dynamical data and upon initial simulation runs were performing better. Similarly for Scheme 6, we used the same approach, but centered the Gaussian priors on parameters from Byrne et al. [13].

Finally, we restrict the values of μ and ω which parameterize the prior distribution of random effects $\eta_n \sim \mathcal{N}(\mu, \omega)$ to avoid over-fitting due to the usage of the MAP conditional likelihood as the optimization objective. Specifically, we use $p(\omega) = \mathcal{N}(0, 1)$ and limit its domain to $[1, \infty]$, as well as limiting the domain of μ to be in $[-5, 5]$. We found that these were the minimal set of restrictions that prevented over-fitting of $\boldsymbol{\Omega}$.

Numerical ODE solving

We use the Julia programming language for numerical ODE solving both during and outside of parameter fitting. Specifically, we use the `DifferentialEquations.jl` package [35]. We use the `Rodas5P` numerical solver which can handle significant stiffness in the ODE system and which performed the best of the methods tried. We used it with the default settings, except for reducing the absolute error tolerance to `abs_tol = 1e-16` since some simulations that contained low concentrations of species suffered from significant errors in the numerical solution.

Model comparison

There are many ways to compare model performance, but for the purposes of this study we use two metrics: root-mean-square error (RMSE) and the Akaike information criterion (AIC) [26]. The RMSE for a single experimental observation vector $y_{nj} \in \mathbb{R}^m$ and model prediction

$y^*_{nj} \in \mathbb{R}^m$ is defined as

$$\mathrm{RMSE}(y_{nj}, y^*_{nj}) = \sqrt{\sum_{i=1}^{m} (y_{nj,i} - y^*_{nj,i})^2} \tag{9}$$

We calculated the RMSE values for each recording in each of the test data sets for each of the 20 optimization runs for a given scheme, pooling them. We also calculated the RMSE values for the same scheme but with published reaction rate constants, again pooling them. This gave us two samples of RMSE values, r_1 that was obtained using the reaction rate constants we fitted and r_2 that was obtained using the published reaction rate constants. We then compared r_1 and r_2 using a two sample T-test (assuming unequal variance) and calculated Cohen's d to establish the effect size of using our rates and the ones published in the literature. Cohen's d is defined as

$$d = \left| \frac{\bar{r}_1 - \bar{r}_2}{s} \right| \tag{10}$$

where $\bar{r}_1$ and $\bar{r}_2$ are the means of each sample and s is the pooled variance. We used the RMSE to focus directly on a models predictive performance.

In contrast, we used the AIC for selecting the model that performs the best when its complexity is taken into account. The AIC for model $\mathcal{M}$ with parameters θ and given data d is defined as

$$\mathrm{AIC}(d, \mathcal{M}, \theta) = 2k - 2\mathcal{L}(d|\mathcal{M}, \theta) \tag{11}$$

where k is the length of the parameter vector θ (in this paper—reaction rate constants for a particular scheme, noise parameter σ and random effect prior parameters μ and ω). Even though in model optimization we use the conditional likelihood, in AIC calculations we used the marginal likelihood $\mathcal{L}$ obtained via the Laplace approximation [36]. The AIC is a measure that evaluates model performance, but also penalizes model complexity via the $2k$ term. There are many other model comparison metrics [37], but the AIC is sufficient for the present study due to the inclusion of predictive model performance and penalizing model complexity along with it being computationally simple to calculate. For each of the 20 different optimization runs we calculated the AIC value for the test data set of a run using the given scheme with our reaction rate constants, as well as if the published reaction rate constants were used. This gave us one sample of AIC values per combination of scheme + reaction rate constants.

Results

General model fitting results

We fit each of the six kinetic schemes shown in Fig 3 to the fluorescence traces from Faas et al. [5] and the steady state calcium-calmodulin binding data from Shifman et al. [11]. We used the root mean square error (RMSE) to evaluate the goodness of fit between the models and the data. The fitting procedure was repeated 20 times with different random seeds, which set the random sampling of the training, validation and test data (see S1 Appendix) along with the initial parameters for optimization. Therefore, due to variability in training data and initial parameters, RMSE values (especially for our parameter fits) for each seed can be significantly different.

We now compare the performance of each kinetic scheme with the reaction rate constants we fit and the published ones. Table 1 shows the training (split between dynamical data from Faas et al. [5] and equilibrium data from Shifman et al. [11]), validation and test data set

Table 1. Summary of training, validation and test performance (RMSE ± SD) for different kinetic schemes with either parameters fit from scratch, fixed to values from publications or our modifications.

Scheme + rate constants	Training (Dynamical data)	Training (Equilibrium data)	Validation (Dynamical data)	Testing (Dynamical data)	Cohen's d‡
Scheme 1 + our fits	0.74 ± 0.12	1.47 ± 0.17	0.80 ± 0.16	0.77 ± 0.17	-
Scheme 1 + Kim et al.	1.51 ± 0.20	2.31	1.16 ± 0.06	1.11 ± 0.08 *	2.07
Scheme 2 + our fits	0.72 ± 0.09	2.00 ± 0.28	0.85 ± 0.15	0.81 ± 0.17	-
Scheme 2 + Bhalla and Iyengar	1.03 ± 0.04	2.23	1.16 ± 0.06	1.11 ± 0.08 *	1.77
Scheme 3 + our fits	0.43 ± 0.03	0.83 ± 0.46	0.48 ± 0.05	0.46 ± 0.05	-
Scheme 3 + Shifman et al.†	0.61 ± 0.05	0.46	0.70 ± 0.11	0.66 ± 0.10 *	3.76
Scheme 4 + our fits	0.42 ± 0.06	0.88 ± 0.10	0.46 ± 0.12	0.44 ± 0.11	-
Scheme 4 + Pepke et al.	0.82 ± 0.05	0.78	0.84 ± 0.06	0.82 ± 0.09 **	3.47
Scheme 5 + our fits	0.37 ± 0.02	0.44 ± 0.18	0.40 ± 0.03	0.38 ± 0.04	-
Scheme 5 + Faas et al.	0.45 ± 0.02	0.75	0.56 ± 0.03	0.53 ± 0.03 **	4.17
Scheme 5 + Pepke et al.	0.86 ± 0.05	0.75	0.88 ± 0.07	0.86 ± 0.09 **	13.5
Scheme 6 + our fits	0.35 ± 0.02	0.35 ± 0.03	0.38 ± 0.03	0.36 ± 0.03	-
Scheme 6 + Byrne et al.	0.47 ± 0.01	0.83	0.55 ± 0.03	0.53 ± 0.02 **	5.83

Highlighted rows were the best performing for that scheme. Note that for models with fixed reaction rate constants trained on Shifman et al. [11] data, no SD is given since all of the data is used for training and there is no variance in this metric. T-tests are done with reference to our reaction rate constants.

† Shifman et al. [11] only contained dissociation constant values, therefore we had to fit on-rate constants while fixing to their K_D values.

‡ Effect sizes evaluated via Cohen's d over 1.2 are considered very large and over 2.0 are considered huge [38].

* $p < 10^{-6}$

** $p < 10^{-10}$

https://doi.org/10.1371/journal.pone.0318646.t001

performance summary statistics for the six investigated kinetic schemes, reporting the average RMSE values for the 20 different seeds used. The distribution of the test RMSE values is shown in Fig 5. On average Schemes 5–6 performed the best, whereas other schemes were not able to capture either the equilibrium data (Schemes 3–4) or both the dynamical and the equilibrium data (Schemes 1–2).

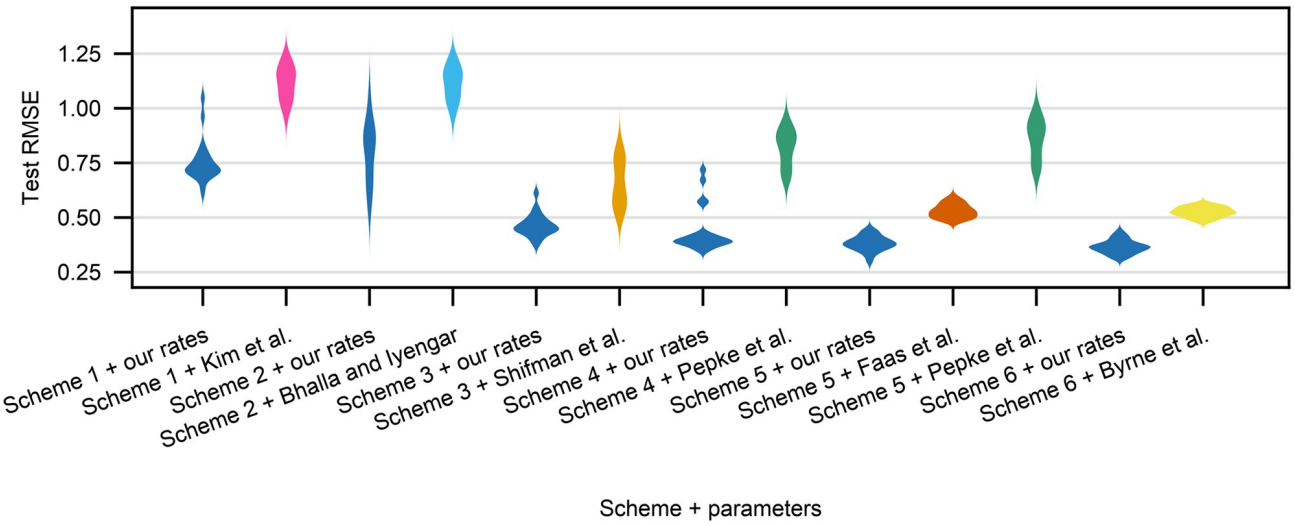


Fig 5. Violin plots of RMSE values for the test data set for each seed for all schemes for our own and the published parameter sets.

https://doi.org/10.1371/journal.pone.0318646.g005

Dynamical behaviour. To illustrate the differences between model fits, Fig 6 shows the measured fluorescence traces that were used as the validation data set for seed 1 and corresponding model predictions for each kinetic scheme with our fitted rate constants and the published rate constants. Each row shows the same 24 experimental traces (black), split between the seven groups of solutions which were used experimentally in Faas et al. [5]. Within each group, each trace corresponds to a different laser uncaging strength used—the stronger the laser, the more calcium gets released, the larger the $\Delta F/F_0$ values that are measured.

The biggest differences in measured and predicted dynamics are for Scheme 1, for which the RMSE differences are also the largest. In Fig 6 the main difference between our parameters and rate constants from Kim et al. [9] is that our rate constants give rise to traces that follow the experimental dynamics to some extent, whereas the published rate constants simply equilibrate to a value and barely display any dynamics (e.g. groups D–G). However, even though our rate constants result in dynamical behaviour, they do not show good equilibrium performance and in fact do not reach equilibrium when the data has long reached it.

The comparison of dynamics for Scheme 2 is similar to that of Scheme 1. Comparing our reaction rate constants with those in Bhalla and Iyengar [10], we see that the published reaction rate constants make the system equilibrate and not follow the data closely, whereas our reaction rate constants achieve a more accurate fit. However, with either our rate constants or the published rate constants, the traces and the average RMSE values indicate that Scheme 2 fits the data quite poorly.

Models with the level of complexity of Scheme 3 and higher are able to capture the dynamical data much better than the simpler Schemes 1 and 2. As shown in Fig 6, both Scheme 3 models perform adequately. However, our reaction rate constants trained from scratch still perform better, especially in capturing the initial rise and fall in $\Delta F/F_0$ (for example see columns D–F). Note that for this scheme the comparison is not entirely equivalent to other cases as we had to fit the forward reaction rate constants while we kept the dissociation rate constants fixed to those in Shifman et al. [11].

For Scheme 4, our reaction rate constants significantly outperform those of Pepke et al. [12] as shown in Fig 6. This is reflected in a smaller mean RMSE value of our rate constants and is evident in most experimental groups, where our rate constants result in reasonably accurate predictions whereas the Pepke et al. rate constants significantly under-predict $\Delta F/F_0$.

Looking at the dynamics for Scheme 5, based on the RMSE values in Table 1, our rate constants perform significantly better than the rate constants from either Faas et al. [5] or Pepke et al. [12], but the gap is much smaller for the former than the latter. The differences in dynamics between our fits and rate constants in Faas et al. are subtle, but generally our rate constants perform better for small amounts of uncaged Ca^{2+}. In contrast, comparing dynamics with our rate constants to dynamics with rate constants in Pepke et al. [12], their reaction rate constants result in significant mismatches to the data, to the point that the optimization procedure has to inject amounts of Ca^{2+} that lead to incorrect equilibrium levels (see Fig 7, which shows that their dissociation constants can fit equilibrium data well).

Finally, for Scheme 6, the main differences between the dynamics resulting from our reaction rate constants and those in Byrne et al. [13] are generally seen for small amounts of uncaged calcium (columns D–G bottom traces). Our reaction rate constants (for this seed) managed to capture calmodulin behaviour with low amounts of Ca^{2+} more accurately. Even though for some traces the published rate constants can outperform ours (e.g. top traces in either groups A or C), our reaction rate constants on average show a smaller RMSE value.

Equilibrium behaviour. Fig 7 shows the equilibrium behaviours of our reaction rate constants (for all 20 training seeds) and published ones. When using Scheme 1 (Fig 7, top row),

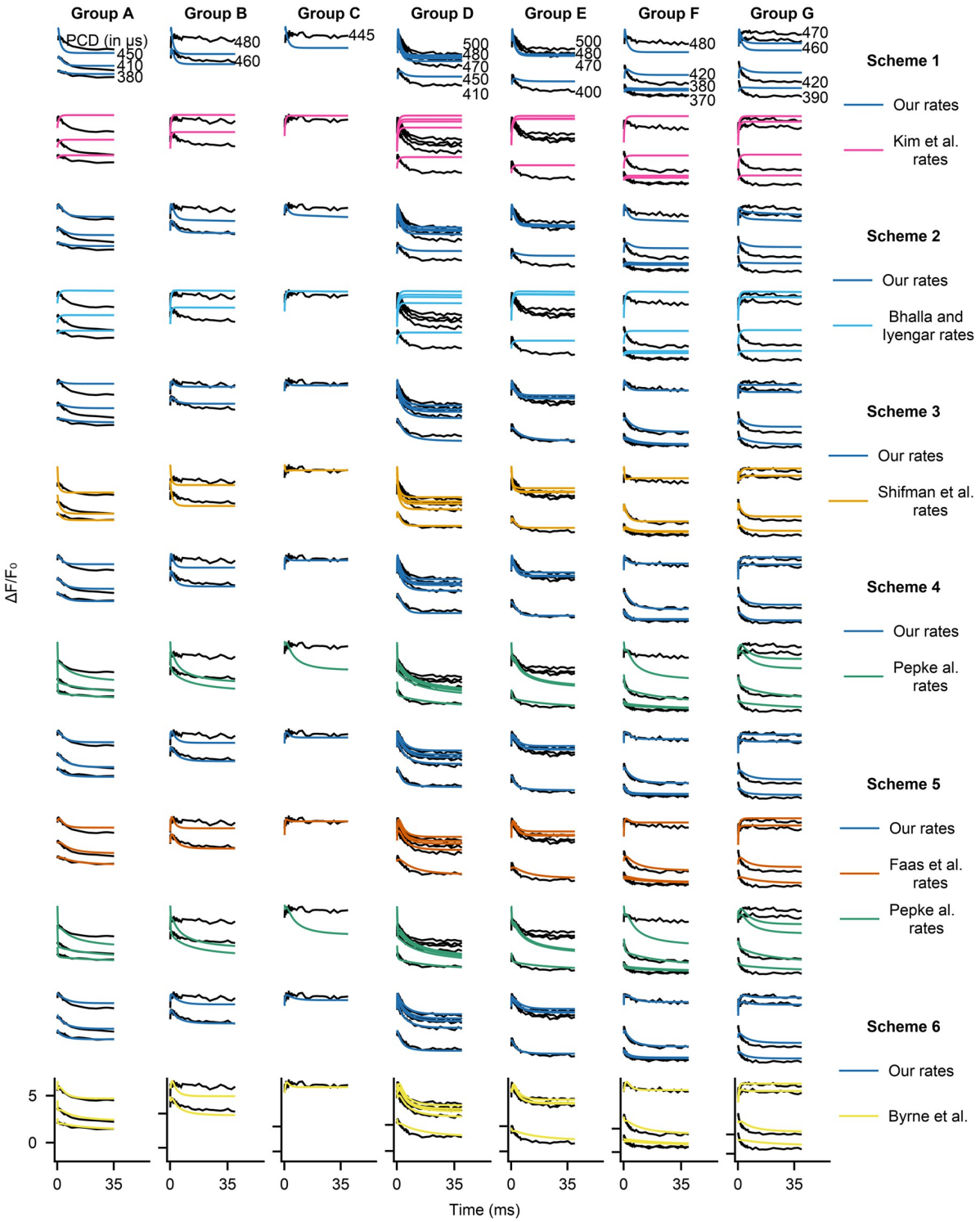


Fig 6. Sample dynamics for validation data for seed 1 for all trained models. Each column is a single data group A–G from Faas et al. [5], whereas each row is a different combination of scheme + reaction rate constants (specific combination given in the rightmost column). In all subplots y axis is $\Delta F/F_0$ and x axis is time. Black lines are empirical data and red lines are model outputs. Scale bars are given at the bottom row, each tick on the y-axis corresponds to the same value in each column.

https://doi.org/10.1371/journal.pone.0318646.g006

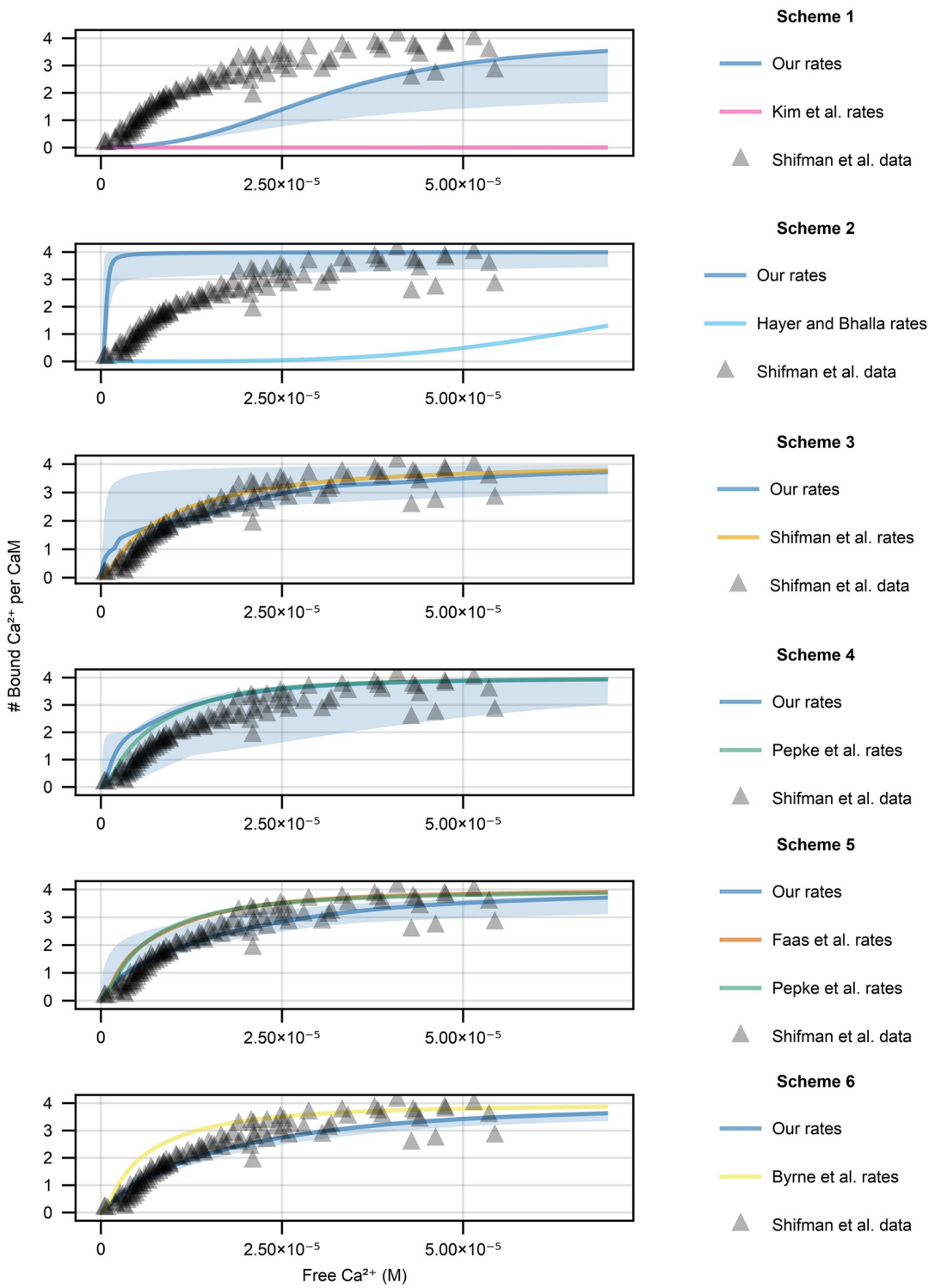


Fig 7. Calmodulin equilibrium behaviour when the free amount of Ca^{2+} is varied for all kinetic schemes for the 20 seeds that we tested (solid line is the median and shaded area is the 95% confidence interval). Model behaviour with our reaction rate constants are plotted in blue, whereas published ones are in the colors indicated. Note that at times the published reaction rate constants (Scheme 1 and Scheme 2) result in behaviours that are much more right-shifted, therefore show up as zero in the relevant range. We also include data points for wild type calmodulin for an equivalent experimental setup in Shifman et al. [11].

https://doi.org/10.1371/journal.pone.0318646.g007

the fits to data from Shifman et al. [11] are generally poor. A single run was able to fit the equilibrium data well, but in most fits, calmodulin was less sensitive to Ca^{2+} than indicated by the data. Some of this behaviour can be attributed to the prior because most of the runs hit the uniform distribution limits (especially the dissociation constants). When the limits were wider, there was a significant amount of training failures (up to 30%) and trained models showed step-like equilibrium behaviour that was overly sensitive to Ca^{2+}. Therefore, we opted to keep the narrower limits. Scheme 1 with our parameter sets is generally much more sensitive to Ca^{2+}than it is with the rate constants from Kim et al. [9], which result in calmodulin behaviour that does not show appreciable Ca^{2+} binding in the relevant Ca^{2+} range and is significantly right-shifted (Fig 7, top row pink line).

For Scheme 2 (Fig 7 second row from the top), our fits result in behaviour that is significantly more sensitive to Ca^{2+} than the experimental data. Calmodulin would be close to fully bound under resting neuronal Ca^{2+} levels, in contrast to the reaction rate constants from Bhalla and Iyengar [10], which are significantly less sensitive to Ca^{2+} than the data indicate, not reaching full calmodulin saturation in the experimental data range. The failure to fit the equilibrium data is likely due to the inclusion of the dynamical data—reaction rate sets that would allow this scheme to fit equilibrium data do not fit the dynamical data well. The failure of Scheme 2 fitting both dynamical and equilibrium data is likely due to the first Ca^{2+} binding event including two Ca^{2+} ions and needing to be relatively fast to fit the dynamical data.

For Scheme 3 (Fig 7 third row from the top), the parameters from Shifman et al. [11] perform very well because they were explicitly tuned to only this data set. However, when the dynamical data from Faas et al. [5] is included in the fitting procedure, the resulting equilibrium behaviour varies between runs (blue shaded area). Similarly to Scheme 2, our fits result in behaviour that is much more sensitive to Ca^{2+} than the data indicates. However, contrary to Scheme 2, the range of behaviours is much more varied and a significant portion of fits match data from Shifman et al. [11] reasonably well.

For Scheme 4 (Fig 7, fourth row from the top), even though in general the fits are much better, there are still a few runs that do not perform as well. Moreover, our mean RMSE value is slightly worse than that parameters from Pepke et al. [12] for the equilibrium data in Shifman et al. [11]. Curiously the mean RMSE for the dynamics predicted via Scheme 4 with our rate constants is much smaller compared to the rate constants from Pepke et al. [12]. A possible explanation for this is that rate constants from Pepke et al. were first derived using Scheme 5 and then reduced to Scheme 4. Scheme 5 is a more powerful model due to having more state variables which could make optimization easier than simply using Scheme 4 (see results below).

Our fits to the Shifman et al. [11] equilibrium data follow a similar pattern for Schemes 5 and 6. For both Schemes the noisiness in model behaviour that was present for Schemes 1–4 is either gone or significantly smaller, and most fits match the data from Shifman et al. [11] reasonably well. In both cases, the average RMSE value using our rate constants is significantly smaller compared to the published reaction rate constants.

Model comparison via AIC

We now compare both the published reaction rate constants to our reaction rate constants and between the kinetic schemes via AIC evaluated on the test set of a random seed. AIC is a useful model comparison tool because it takes into account both model predictive performance as well as model complexity (number of parameters). Fig 8 shows the box plots for the AIC values for all the combinations of kinetic scheme and parameter set for all 20 seeds. As shown in Table 2, our reaction rate constants have lower median AIC values (lose less information)

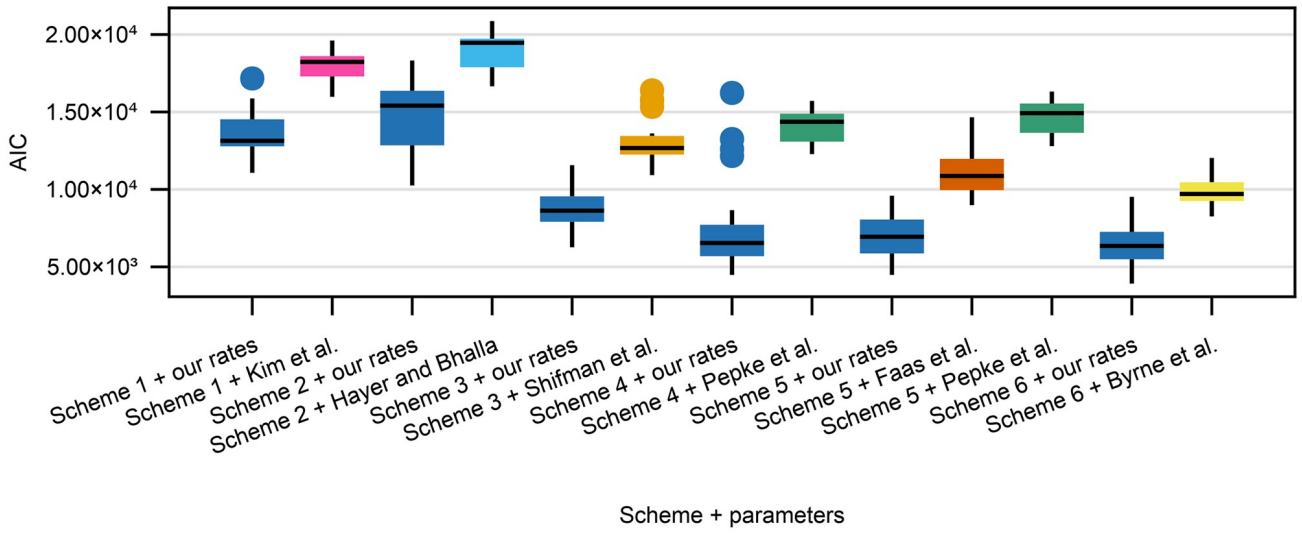


Fig 8. AIC box plots for all the different combinations of kinetic schemes and either previously published or our own reaction rate constants. Each box plot is based on training the model on 20 different random seeds, the AIC value is calculated on the test data set of a given seed.

https://doi.org/10.1371/journal.pone.0318646.g008

compared to the published ones. Moreover, the more complex the scheme, the lower the AIC value, with Scheme 6 performing the best. The median AIC values seem to asymptote and reach a lower value by Scheme 6 (the big change is after Schemes 2–3), so only qualitatively different model improvements are likely to decrease the AIC value more.

Calculating the relative likelihoods from median AIC values, where our reaction rate constants are the reference, all the published parameter sets have negligibly low relative likelihoods (largest being on the order of e^{-100}). Therefore, our reaction rate constants are significantly more likely compared to the published ones.

Since our reaction rate constants have lower AIC values (lose less information) compared to the published ones, we use our reaction rate constants to compare between different kinetic schemes. Given the results in Fig 8 and Table 2 and using the median AIC for Scheme 6 with our rate constants as reference, the other schemes with our parameter sets have a negligibly small relative likelihoods (again on the order of e^{-100}). Therefore, of the combinations of schemes + reaction rate constants that we found, Scheme 6 with our reaction rate constants is relatively the most likely.

Calmodulin Ca^{2+} integration properties

Having established that our reaction rate constants are significantly more likely than the published ones, we now ask whether this difference is meaningful practically. To answer this

Table 2. Median AIC values for all combinations of kinetic schemes and reaction rate constants (our fits or published).

	Median AIC values ($\times 10^3$)					
Parameter source	**Scheme 1**	**Scheme 2**	**Scheme 3**	**Scheme 4**	**Scheme 5**	**Scheme 6**
Ours	12.9	14.9	8.32	6.37	6.45	5.97
Published	17.9	18.4	11.5	14.2	10.2† / 14.8‡	9.21

† when reaction rate constants from Faas et al. [5] are used.

‡ when reaction rate constants from Pepke et al. [12] are used.

https://doi.org/10.1371/journal.pone.0318646.t002

question we probe the Ca^{2+} signal integration properties of calmodulin. CA1 pyramidal cell Schaffer collateral synapses undergo long-term potentiation dependent on CaMKII (and therefore on calmodulin) in response to three 1s trains of 50Hz stimulation [39]. Given these results, it is likely that calmodulin integrates the Ca^{2+} signal within a single train. Therefore, we set up a series of simulations where a model was stimulated by a 1s train of Ca^{2+} injections but the frequency was varied from 2Hz to 100Hz. Based on the results in Sabatini et al. [40], Ca^{2+} influx due to single synaptic stimulation event for a neuron at resting voltage is around 0.7μM (this mimics the experimental setup in Bayazitov et al. [39] best). To mimic the competition between calmodulin and other buffers and pumps we implemented a minimal Ca^{2+} extrusion model using values in Sabatini et al. [40] Table 1 for a CA1 pyramidal cell spine—Ca^{2+} decaying to a baseline of 100nM with a time constant τ = 12ms. Finally, we use a biologically realistic calmodulin concentration of 20μM [41]. After the simulation, we evaluate the calmodulin signal integration by calculating the area under the curve of both partially bound calmodulin and fully bound calmodulin, where bigger values indicate a larger level of Ca^{2+} signal integration.

As shown in Fig 9 columns one and two, Schemes 1 and 2 are not capable of integrating Ca^{2+} signals in the tested frequency range (except for a few outlier runs with Scheme 2). For Scheme 1 it is likely due to the fact that the models are not sensitive enough to Ca^{2+} (see Fig 7 top row). The same interpretation, however, does not hold for Scheme 2, whose equilibrium behaviour with our parameter fits was usually too sensitive to Ca^{2+} compared to experimental data. This behaviour for Scheme 2 may be explained by slow reaction rate constants compared to Ca^{2+} decay.

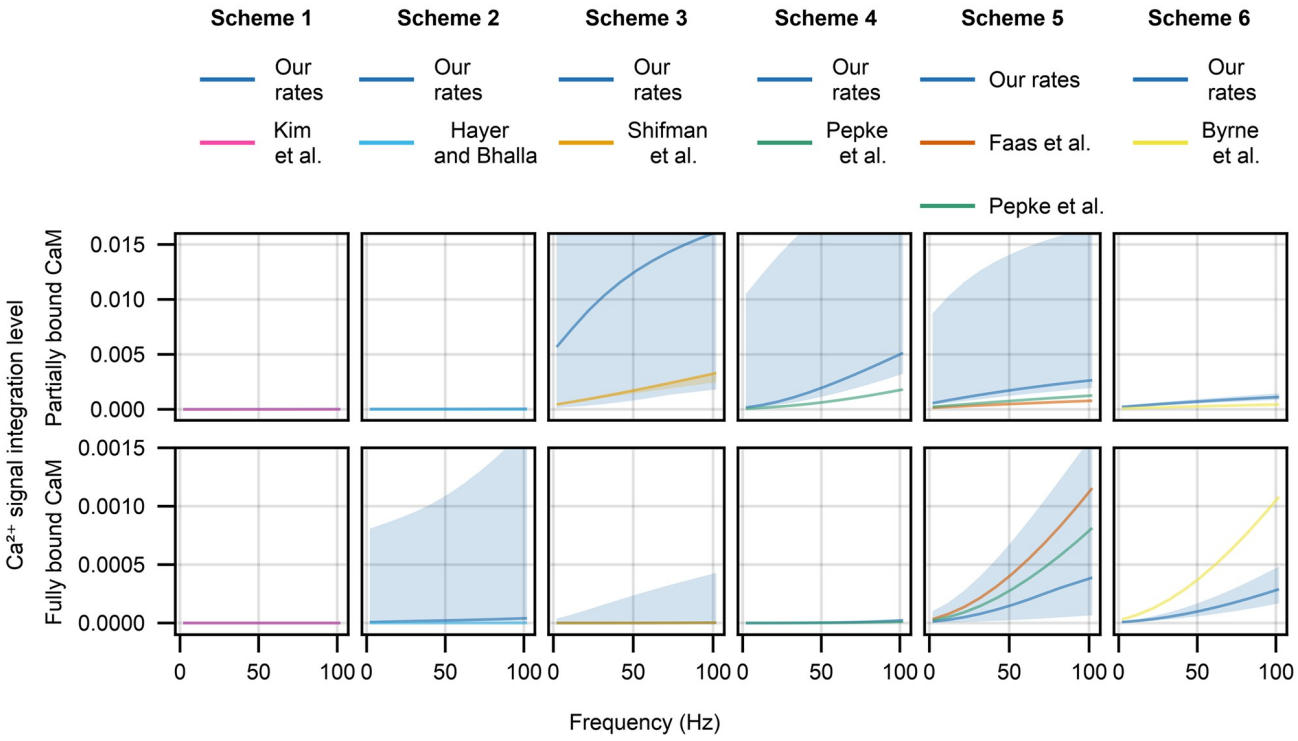


Fig 9. Ca^{2+} signal integration properties (measured as area under the curve) of partially (first row) and fully bound (second row) calmodulin species in response to a 1sec train of 2–100Hz stimulation that delivers 0.7μM Ca^{2+} per spike (same general patterns hold if 12μM Ca^{2+} per spike is delivered, results not shown). A different calmodulin scheme is used in each column and shows our parameter fits (deep blue lines) and the published parameter values (all other colours). The solid lines are median model behaviour and shaded areas are the 95% confidence intervals.

https://doi.org/10.1371/journal.pone.0318646.g009

The results are somewhat different for Schemes 3 and 4 (Fig 9 columns three and four), where both the published reaction rate constants and our own fits show significant Ca^{2+} integration in the partially bound calmodulin species, but barely any in the fully bound calmodulin. In both cases our reaction rate constants result in significantly higher Ca^{2+} signal integration. However, results for Scheme 3 with our reaction rate constants which show significant Ca^{2+} integration at 2Hz should be taken with caution due to the same fits being overly sensitive to Ca^{2+} at equilibrium (Fig 7 third row from the top).

Finally, for Schemes 5 and 6 we see integration of Ca^{2+} signals that results in both fully and partially bound calmodulin species (Fig 9 columns five and six). For both schemes our reaction rate constants predict that Ca^{2+} signal integration would result in more partially bound calmodulin compared to predcitions from the published rate constants. As for fully bound calmodulin, our reaction rate constants predict a lower level of fully saturated calmodulin than the predcitions from the published rate constants.

The difference between the partially and fully bound calmodulin signals is more pronounced with our reaction rate constants than with the published ones. For Scheme 5 the difference is around an order of magnitude for our reaction rate constants and under 2-fold for the published reaction rate constants, whereas for Scheme 6 the difference is around 4-fold for our reaction rate constants and around 2-fold for the published reaction rate constants. Given that our models reaction rate constants perform better, we predict that partially bound calmodulin species play a more significant part in Ca^{2+} signalling integration and propagation than predicted by previously published models.

Parameter correlations

We next examine the relationships between our parameter fits within a given scheme. Analysing relationships between parameters may point future experimental research questions. For example, if some reaction rate constants are correlated, they may be under-determined. Therefore, future model development would benefit from additional, more directed data to better constrain the correlated parameters. We use partial correlation as a measure of relationship between parameters [19]. Briefly, partial correlation quantifies the degree of association between two variables when the variance from a set of controlling variables is taken into account. For example, in Scheme 1 the partial correlation between k_1 and k_3 would indicate the relationship between these two rate constants when K_{D_1} and K_{D_1} is accounted for.

Since structurally there is nothing to distinguish between the C and the N lobes for Schemes 4–6, we calculate the dissociation constant for the two binding reactions in Scheme 4 and for the first binding reaction for both lobes in Scheme 5 and Scheme 6 and compare their values—the one that has a lower K_D value we call the C lobe and the one that has a higher value we call the N lobe. This is to avoid what we call the C lobe being functionally the N lobe and vice versa, which would result in artificially higher parameter spread or obscure parameter correlations. We show the parameter pair plots for all six schemes in the S5 Appendix, along with tables of individual reaction rate fits. We show partial correlation coefficients for all schemes in Fig 10.

First of all, even though the data set we use is richer, as it includes both the dynamical and the equilibrium data, there is still significant variance in our model parameter fits. For some parameters the pairs can span 5 to 10 orders of magnitude (see S5 Appendix). Moreover, there are significant correlations between multiple parameters in most schemes.

There is only one significant negative correlation for Scheme 1, between K_{D_1} and K_{D_2}. This correlation is most likely due to a limited number of degrees of freedom offered by this scheme. Assuming that a calmodulin molecule has an overall dissociation constant that is a

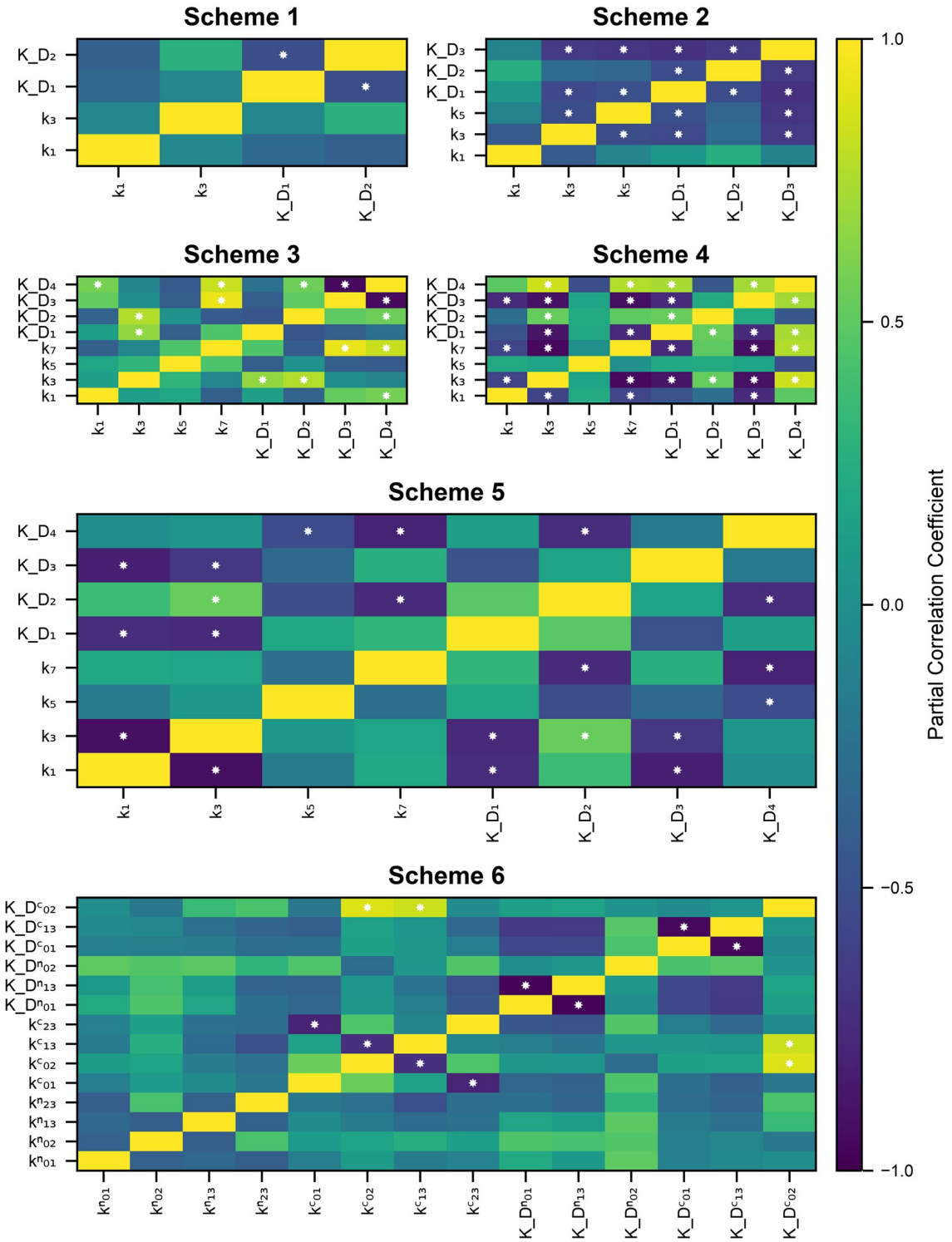


Fig 10. Partial correlation coefficients for our reaction rate constant fits for all schemes. White stars in the rectangle indicate $p < 0.05$ with null hypothesis that the partial correlation is zero.

https://doi.org/10.1371/journal.pone.0318646.g010

function of the dissociation constants of individual reactions, the dissociation constants of individual reactions have to co-vary in order to maintain the same overall behaviour.

Reaction rate constants for Schemes 2 have 8/15 significant correlations which indicates a high level of sloppiness in the system. Similarly to Scheme 1, given the high level of simplification used in this scheme, in order to maintain the same overall model behaviour, parameters have to co-vary. This is especially true for the dissociation constants, which are all are negatively correlated. We also see that the on-rate for the first reaction is not correlated to any other parameter and is therefore fairly well-determined.

Scheme 3 has proportionally fewer correlations than Scheme 2 (7 out of 28 parameters correlated). Most correlations are with the dissociation constant for the final Ca^{2+} binding event. This can be explained by thinking about a general dissociation constant for calmodulin, which would be a function of K_{D_1}, K_{D_2}, K_{D_3} and K_{D_4}—since K_{D_1} is likely better constrained due to the first binding event needing to be of particular speed to fit the rising/falling phase of the individual time series, the other dissociation constants may co-vary more freely and balance each other out. The on-rate for the first binding event is only correlated to one variable, which indicates that it is quite well determined.

Scheme 4 shows the largest fraction of correlations of all the schemes (15/28). This is most likely due to the quasi-steady state approximation which results in steady state reaction rate constants $\{k_i^{ss}\}_{i=1}^{4}$ that provide a lot of room for sloppiness via products and quotients of the full set of reaction rate constants $\{k_i\}_{i=1}^{8}$.

The point that model structure is of utmost importance in determining the levels of sloppiness in the system is further reinforced by Scheme 5, where 10 out of 28 reaction rate constants were correlated. More importantly, a significant number of correlations are within-lobe, for example k_1 and k_3—the first and second on-rate constants. There are also some cross-lobe correlations, for example K_{D_2} and K_{D_4} which are the second Ca^{2+} binding events for C and N lobes respectively.

Curiously, even though Scheme 6 is the most complex in terms of number of parameters and number of states, it shows only six significant correlations between reaction rate constants. Moreover, all correlations are within a lobe, rather than between lobes. More specifically, most of them are for parameters in the C lobe, rather than the N lobe.

Necessary structural components of a calmodulin model

As shown in Table 1 and in Fig 6, there is a large gap in training performance between Schemes 1–4 and Schemes 5–6. Even though training RMSE in both dynamical and equilibrium data significantly decreases going from Scheme 2 to Scheme 3, only from Scheme 5 onwards can both dynamics and equilibrium behaviour be captured well. There are two main differences between Schemes 1,2,4 and and 5–6: independence of lobes and structural assumption of co-operativity. Both Scheme 3 and Schemes 5–6 allow co-operativity (via reaction rate constants) but do not assume it structurally. Schemes 3 does not allow for independence of lobes, while Schemes 5–6 assume it structurally. In this section we provide an empirical argument that links model features to gaps in performance, focusing on event-based (as opposed to binding site-based) and structurally co-operative (especially for the C lobe) schemes to model calmodulin.

Assuming that the real calmodulin dynamics operate in a k-dimensional space, any model capable of modeling the dynamics would have to have at least that many dimensions (along with an appropriate structure). Calmodulin models framed in terms of events (fully abstracted from binding sites) can operate at most in a four dimensional linear subspace (since rank of such a network is four, see page 30 in [42]) of the five dimensional state space (see Scheme 3 in

Fig 3). Therefore, an immediate conclusion of this may be that $k > 4$, real calmodulin dynamics operate in a higher dimensional space than an event-based model allows for. However, Scheme 5, which is able to model both calmodulin dynamics and equilibrium behaviour (see Table 1), has rank 4 as well. The main difference between Schemes 3 and 5 are the independence of the lobes: Scheme 5 contains two independent subnetworks (each of which is rank 2). Therefore, based on our results, in order to accurately model both calmodulin dynamics and equilibrium behaviour, two independent subnetworks (independence of lobes) is a necessary model feature.

We next analyze whether a structural assumption of co-operativity, modelling the binding of two Ca^{2+} ions as a single event, within calmodulin lobes is reasonable. This is not the only way of modelling co-operativity, but it results in models with a smaller state space vector and therefore can be preferable computationally. Fractional calmodulin occupancy of the N and the C lobes using a well performing model (Scheme 6 with parameters from Byrne et al. [13]) is shown in Fig 11 columns one and two. Starting with the dynamics of the partially occupied N lobe, the model predicts around 20% of calmodulin molecules would have the first site occupied, with a negligible fraction having the second site occupied. Moreover, the dynamics of partially occupied sites in the N lobe do not show fast changes over the simulated time period, so the quasi-steady state approximation would hold reasonably well. The dynamics of the C lobe paint an opposite picture. It is immediately obvious that, due to its slower speed, the quasi-steady state approximation ($d[CaMC_1]/dt = 0$) does not hold for the C lobe as there are calmodulin dynamics occurring over the whole simulated time of 35ms. Therefore, even though it is a theoretically appealing tool to reduce the number of calmodulin states, the quasi-steady state approximation is too inaccurate for the C lobe and results in significant errors in either calmodulin dynamics or equilibrium behaviour.

Discussion

We used a rich dynamical [5] and equilibrium [11] data set to fit six calcium-calmodulin kinetic schemes from scratch in order to compare to published models. Our comparison resulted in a number of conclusions.First of all, the parameters we found, as opposed to the published ones, resulted in significantly better fits on our dataset (Table 1). Secondly, we showed that fully event-based schemes that do not utilize any features of the calmodulin physical structure (existence of C and N lobes) result in significantly worse generalization performance as measured via AIC (Fig 8). Thirdly, we investigated calmodulin signal integration properties by comparing our parameter fits to published reaction rate constants for different calcium-calmodulin schemes. Some schemes showed no Ca^{2+} signal integration in response to a stimulation protocol mimicking an empirically effective plasticity induction protocol highlighting the importance using more detailed calmodulin schemes (Fig 9). Fourthly, we calculated the partial correlations between our parameter fits (Fig 10). Partial correlations revealed that even with our data set, that is richer than anything used before, some parameters were correlated and therefore under-determined. Finally, we investigated the validity of the quasi-steady state approximation used in [12] and by using Faas et al. [5] data we showed that it is not accurate for the C lobe. We next discuss each of these conclusions individually.

First of all, model performance depends on the data which was used to parameterise it. Even though usage of multiple data sources to fit a calmodulin model is not new and was done in Pepke et al. [12], we are the first to combine a data source on calmodulin dynamics [5] and a data source on calmodulin equilibrium behaviour [11]. We used this combined data set to fit six different calcium-calmodulin kinetic schemes previously used in the literature. We then compared our parameters to the published ones which revealed that a significant number of

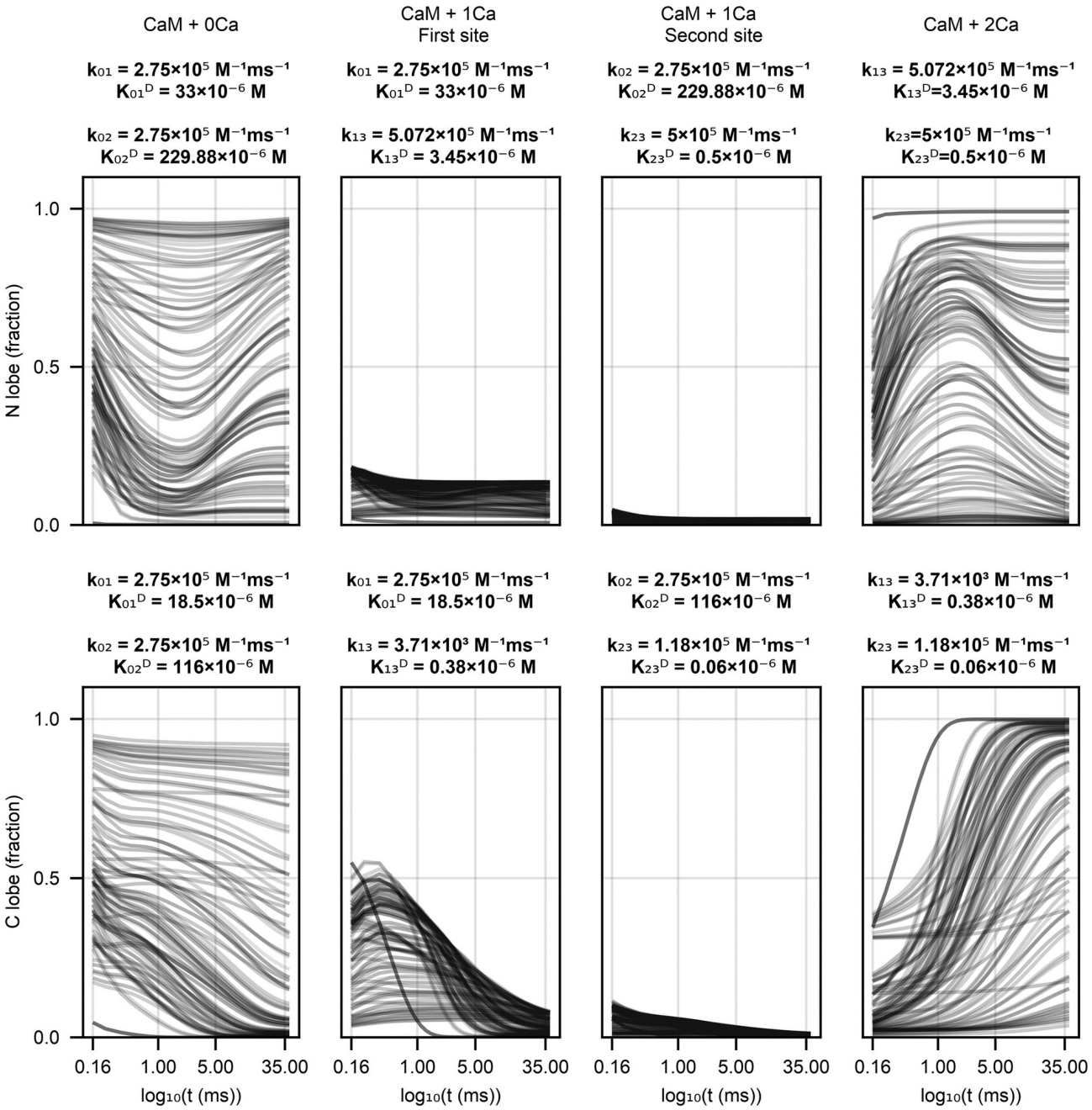


Fig 11. Scheme 6 behaviour with parameters from Byrne et al. [13] on dynamical data from Faas et al. [5]. Each line is one of the initial conditions (solution + uncaging strenght) Faas et al. used. Calmodulin has been normalized to total calmodulin used in an experiment, the first row shows N lobe dynamics, the second row shows C lobe dynamics. Each column shows a different calmodulin state—completely unbound (first column), Ca^{2+} bound to the first site on a lobe (second column), Ca^{2+} bound to the second site on a lobe (third column), Ca^{2+} bound to both sites of a lobe (fourth column). Above the plots we provide reaction rate constants from Byrne et al. [13] for the reactions a specific calmodulin state participates in. Note that time on the *x* axis is in log10 space to better show the initial dynamics.

https://doi.org/10.1371/journal.pone.0318646.g011

calcium-calmodulin models used in the literature are parameterized sub-optimally (see Table 1). Most published models (except Faas et al. [5]) have relied on either only equilibrium data [11, 12, 43] or dynamical data obtained under significant methodological limitations—such as dead time in stopped flow fluorimetry or presence of other biochemical species [9, 13]. Undoubtedly, it would be unfair to criticize past work for operating under the limitations of the day, but that does not prevent models from becoming outdated (much as this work will be one day). Therefore, an important contribution in this paper are the improved model parameters for calcium-calmodulin models—the best performing parameter sets for each scheme are given in Table 3 (see S5 Appendix for all 20 parameter sets for each scheme).

Secondly, our calmodulin model comparison uncovered discrepancies in performance between different kinetic schemes. The complexity of calmodulin schemes we investigated ranged from a model with three states and four parameters [9] to a model with eight states and sixteen parameters [13]. There were only two schemes (5 and 6, consisting of eight and sixteen parameters respectively) that were able to fit both sources of data well—both schemes modelled calmodulin lobes separately and consisted of individual, rather than lumped, Ca^{2+} binding reactions. Two further schemes (3 and 4), one of which modelled calmodulin lobes but not

Table 3. Our reaction rate sets that performed best on the test data and the published reaction rate constants from literature. All parameters are in $\log_{10}$, but are in different units, depending on the context: for second order reactions the forward reaction rate constants are in $M^{-1}ms^{-1}$, dissociation constants in M, for third order reactions the forward reaction rate constants are in $M^{-2}ms^{-1}$, dissociation constants in M^2.

Source	Parameters							
Scheme 1 parameters								
	k_1		K_{D_1}		k_3		K_{D_2}	
Our fits	8.10		-9.00		4.00		-9.00	
Kim et al.	3.60		-5.69		5.0		-4.96	
Scheme 2 parameters								
	k_1		K_{D_1}	k_3	K_{D_2}		k_5	K_{D_3}
Our fits	8.13		-9.00	2.25	-9.00		4.52	-6.40
Hayer and Bhalla	4.86		-6.00	3.56	-5.55		2.67	-4.68
Scheme 3 parameters								
	k_1	K_{D_1}	k_3	K_{D_2}	k_5	K_{D_3}	k_7	K_{D_4}
Our fits	5.41	-4.87	2.36	-7.83	4.33	-6.40	5.04	-6.19
Shifman et al.	-	-5.10	-	-5.77	-	-4.46	-	-5.05
Scheme 4 parameters								
	k_1	K_{D_1}	k_3	K_{D_2}	k_5	K_{D_3}	k_7	K_{D_4}
Our fits (S4)	4.33	-5.56	5.98	-6.01	7.03	-4.13	6.29	-5.73
Pepke et al.	3.60	-5.00	4.00	-6.03	5.0	-4.60	5.18	-5.30
Scheme 5 parameters								
	k_1	K_{D_1}	k_3	K_{D_2}	k_5	K_{D_3}	k_7	K_{D_4}
Our fits	4.50	-5.43	4.08	-5.84	5.69	-4.00	6.32	-5.30
Faas et al.	4.90	-4.60	4.40	-6.60	5.90	-3.70	7.50	-6.10
Scheme 6 parameters								
	k_1^c	k_3^c	k_5^c	k_7^c	$K_{D_1}^c$	$K_{D_2}^c$	$K_{D_3}^c$	$K_{D_4}^c$
Our fits	4.23	4.16	5.22	2.31	-5.40	-4.38	-5.38	-6.40
Byrne et al.	5.44	5.44	3.57	5.07	-4.73	-3.94	-6.42	-7.22

	k_1^n	k_3^n	k_5^n	k_7^n	$K_{D_1}^n$	$K_{D_2}^n$	$K_{D_3}^n$	$K_{D_4}^n$
Our fits	3.03	6.65	5.22	7.58	-4.06	-5.20	-2.88	-6.38
Byrne et al.	5.44	5.44	5.71	5.70	-4.48	-5.46	-3.64	-6.31

Shifman et al. only contained the dissociation constants, so the forward reaction rate constants have no point of comparison. The same set of reaction rate constants from Pepke et al. has been used in both Schemes 4 and 5, but they are only shown for Scheme 4 to avoid repetition and misleading as the implementation of Scheme 5 in Faas et al. and Pepke et al. is slightly different, structurally $k_{Pepke} = 2k_{Faas}$ for some reaction rate constants.

https://doi.org/10.1371/journal.pone.0318646.t003

individual binding, another which modelled individual binding but not lobes, were able to fit dynamical data, but not equilibrium data, reasonably well. Both of these schemes consisted of eight parameters, same as one of the schemes that fit both sources of data well, indicating that the number of parameters is not the only factor necessary for an accurate calcium-calmodulin model. Finally, two of the simplest schemes (Schemes 1 and 2) that did not model calmodulin lobes and modelled Ca^{2+} binding as lumped reactions were not able to fit either the dynamical data or the equilibrium data well. These results, along with median AIC values (Table 2) lead to the second contribution of this paper—Scheme 6 is the most accurate calcium-calmoduling binding scheme and, compared to some simpler schemes, by a significant margin.

Thirdly, our results provide implications for models that include calmodulin. We investigated the Ca^{2+} integration properties of calmodulin in response to a realistic Ca^{2+} spike train (see Fig 9). The biggest practical difference between our reaction rate constants and published ones is that there is a much more significant contribution from partially bound calmodulin species, rather than fully bound calmodulin. As shown in Shifman et al. [11], CaMKII can be activated by partially bound calmodulin. Moreover, calmodulin has many binding partners, such as Calcineurin [44], Phosphodiesterase 1 [45], Adenylyl cyclases 1 and 8 [46], Neurogranin [47, 48] and others [2]. Our results bring into question the accuracy of the results of publications where poorer performing schemes or parameterisations are used in larger models [9, 18, 19, 21–23, 49–52]. There are many ways to compensate for the poor performance of calmodulin scheme or parameters. For example, it is possible that in some cases the lack of calmodulin sensitivity to Ca^{2+} has been compensated for by an increased Ca^{2+} influx. However, for example Scheme 1 is used in [23] in a dynamical setting, stimulating their large model with many protein species with e. g. 180s of 5Hz or 1sec of 100Hz Ca^{2+} pulses. As our results show, the calmodulin Ca^{2+} integration properties are significantly different in this range when our reaction rate constants are used. Our third contribution is support to the hypothesis that partially bound calmodulin molecules arising in response to different Ca^{2+} stimuli is an additional dimension of signal encoding and propagation towards downstream pathways compared to spatial/concentration based fully bound calmodulin signalling. Future investigations into other calmodulin binding partners and their activation by partially bound calmodulin species would be able to falsify this hypothesis.

Fourthly, our results on the partial correlations between reaction rate constants form our fourth contribution—the call for more empirical investigations to test the distinctness of Ca^{2+} binding sites within a calmodulin lobe. Generally with increasing model complexity there were fewer correlations (except for Scheme 4, which had more than Scheme 3) between parameters, indicating the parameters were better determined by data. However, even for the most complex Scheme 6, there were some correlations between parameters within the same lobe. These correlations could only be eliminated by additional information on the properties of individual binding sites. Existing studies with mutations of individual calmodulin binding sites only include equilibrium measurements [11, 53, 54]. Since equilibrium behaviour only informs the ratio between the Ca^{2+} binding and unbinding rate constants, they are of limited usefulness in fitting. The closest to the necessary measurements were done in Faas et al. [5] where dynamical measurements with one inactive calmodulin lobe (either C or N) were made.

Finally, we investigated the validity of the quasi-steady state approximation used in [12]. Both Scheme 4, in which partially bound calmodulin species are not modelled due to the quasi-steady state approximation, and Scheme 5, which models them, can model calmodulin dynamics to a similar accuracy. The main difference between the schemes is in equilibrium behaviour, where in Scheme 4 the modelling of dynamics impedes modelling of steady state behaviour. These results imply that the quasi-steady state approximation used in Pepke et al. [12] does not hold in the context of the Faas et al. [5] data, at least not without significant

decrease in the accuracy of model behaviour. Ideally, an empirical measurement of the occupancy of individual calmodulin sites in a dynamical setting would be definitive in falsifying this approximation. Unfortunately, such data does not exist therefore we used Scheme 6 with Byrne et al. [13] parameters (since they fit the data reasonably well) and simulated the fractional occupancy of individual calmodulin sites under [5] experimental conditions (see Fig 11). These results support our fifth contribution—that the quasi-steady state approximation is not valid and results in a significant loss of accuracy, especially for the C lobe.

Having discussed the contributions of this paper we now reflect on their wider implications and practical reality of computational modelling. Suboptimal schemes or parameterisations of calcium-calmodulin models used in large models are a difficult challenge. It is not necessarily the case that the conclusions drawn from large models are made invalid. In large models it is likely possible to correct for the model-data mismatch arising due to inaccurate calmodulin behaviour via the parameters of reactions involving downstream molecules. This, however, may result in a panoply of different mechanistic hypotheses if different publications correct for these inaccuracies arising due to poor calmodulin models in different ways. A more co-ordinated community effort with some agreed upon set of model tests (such as the FindSim platform suggested by [55]), akin to continuous integration in GitHub, may be necessary to resolve such issues in the future and build performant large models.

Limited computational resources and the difficulty of writing large models mean that in some cases it may not be feasible to use a more detailed calmodulin scheme because of an exponential explosion in the number of species to be modelled and the subsequent increase in the computational cost of simulations. Rule-based modelling [56] with its "don't care, don't write" approach (only having to specify the features of a species which impact a reaction) allows models containing exponentially large numbers of complexes to be written down but may still be too computationally costly. Modeling is a complex task that involves many behind the scenes choices about acceptable trade-offs. Our results provide the information about the trade-offs in model accuracy being made when choosing one calmodulin scheme (or parameter set) over another.

In the final two paragraphs we discuss the methodology we used, the available alternatives and limitations. We used NLME fitting algorithms implemented in `Pumas.jl` to fit the reaction rate constants of the different calcium-calmodulin kinetic schemes. There are many published pipelines for fitting reaction rate constants of kinetic schemes. For example, Eriksson et al. [57] propose and use a pipeline based on approximate Bayesian computation Markov Chain Monte Carlo (ABC-MCMC, using R-vines). MCMC approaches are powerful tools which benefit from inherently providing uncertainty on model parameters, rather than having to run optimization on different random seeds as was done in this study. However, they are generally much more computationally expensive. Another popular option is the Data2Dynamics toolbox [58], which streamlines construction of models of chemical reaction networks and modeling of experiments while leveraging ODE solving capabilities of MATLAB, along with stochastic optimization. However, there are few modern software packages that deal with NLME models (which were required due to the nature of the dynamical data in Faas et al. [5]). Of these packages `Pumas.jl` is currently the most performant one [24]. This is in part because `Pumas.jl` is implemented in the Julia programming language which contains state of the art ODE solving capabilities, outperforming its competitors in terms of speed by orders of magnitude (see benchmarks.sciml.ai).

Even with a powerful computational pipeline, there are still many nuances, practical considerations and limitations. For example, the length of the time series to which parameters are being fit impacts the complexity of the loss surface—the more points, the more complex it is [59]. Therefore, we downsampled the initial part of the dynamical data from Faas et al. [5] (see S1 Fig). However, invariably, downsampling results in loss of signal, therefore more

performant downsampling techniques or multiple shooting based approaches may have resulted in even better fits. Moreover, we simplified the Ca^{2+} uncaging model used in [5] to make parameter optimization more stable. Also, [5] used Pockels cell delay (PCD) as the independent variable to predict the fraction of uncaged Ca^{2+} whereas we omitted this variable as it did not perform as well in practice. More data on the relationship between PCD and Ca^{2+} uncaging fraction would have allowed us to derive a better Ca^{2+} uncaging model that potentially could have improved model predictions with both published and our own reaction rate constants. Finally, in order to prevent training failures due to numerical instabilities in ODE solutions when using some schemes, we had to restrict the range of possible values taken by their reaction rate constants. Usage of novel ODE solvers capable of handling stiff systems is a potential avenue to remedy this limitation in future studies. Therefore, even with a more powerful software pipeline, some trial and error and practical trade-offs were necessary to fit our own parameters and efficiently and accurately compare different calmodulin models.

In conclusion, we believe that we have provided a number of important contributions that advance calcium-calmodulin modelling. We conducted a data-driven evaluation of both calcium-calmodulin kinetic schemes and parameter sets used in existing publications and showed which schemes or parameter sets performed poorly. It may be argued that behaviour of single molecules in large models matters less than the behaviour of the overall model. However, if large models are to be useful in predicting the behaviour of real biological systems, the individual molecules and their accurate generalization performance are of utmost importance.

Supporting information

S1 Appendix. Subset of data used and its splitting into training, validation and testing data sets.
(PDF)

S2 Appendix. Concentrations of species of different groups of solutions used in Faas et al. data.
(PDF)

S3 Appendix. Derivation of steady-state reaction rate constants for Scheme 4.
(PDF)

S4 Appendix. Published reaction rate constants used in this study.
(PDF)

S5 Appendix. Our parameter fits for all schemes.
(PDF)

S1 Fig. Comparison of the original data set and the data with subsampled initial period.
(TIF)

Author Contributions

Conceptualization: Domas Linkevicius, Melanie I. Stefan, David C. Sterratt.

Data curation: Guido C. Faas.

Investigation: Domas Linkevicius.

Methodology: Domas Linkevicius, Melanie I. Stefan, David C. Sterratt.

Software: Domas Linkevicius.

Supervision: Angus Chadwick, Melanie I. Stefan, David C. Sterratt.

Visualization: Domas Linkevicius, Melanie I. Stefan, David C. Sterratt.

Writing – original draft: Domas Linkevicius.

Writing – review & editing: Domas Linkevicius, Angus Chadwick, Guido C. Faas, Melanie I. Stefan, David C. Sterratt.

References

1. Berridge MJ, Bootman MD, Roderick HL. Calcium signalling: Dynamics, homeostasis and remodelling; 2003.
2. Xia Z, Storm DR. The role of calmodulin as a signal integrator for synaptic plasticity. Nature Reviews Neuroscience. 2005; 6:267–276. https://doi.org/10.1038/nrn1647 PMID: 15803158
3. Chin D, Means AR. Calmodulin: a prototypical calcium sensor. Trends in Cell Biology. 2000; 10(8):322–328. https://doi.org/10.1016/S0962-8924(00)01800-6 PMID: 10884684
4. Andrews C, Xu Y, Kirberger M, Yang JJ. Structural Aspects and Prediction of Calmodulin-Binding Proteins. International Journal of Molecular Sciences. 2021; 22(1). https://doi.org/10.3390/ijms22010308
5. Faas GC, Raghavachari S, Lisman JE, Mody I. Calmodulin as a direct detector of Ca2+ signals. Nature Neuroscience. 2011; 14:301–304. https://doi.org/10.1038/nn.2746 PMID: 21258328
6. Linse S, Helmersson A, Forsén S. Calcium binding to calmodulin and its globular domains. Journal of Biological Chemistry. 1991; 266(13):8050–8054. https://doi.org/10.1016/S0021-9258(18)92938-8 PMID: 1902469
7. Putkey JA, Kleerekoper Q, Gaertner TR, Waxham MN. A New Role for IQ Motif Proteins in Regulating Calmodulin Function. Journal of Biological Chemistry. 2003; 278:49667–49670. https://doi.org/10.1074/jbc.C300372200 PMID: 14551202
8. Heil KF, Wysocka EM, Sorokina O, Kotaleski JH, Simpson TI, Armstrong JD, et al. Analysis of proteins in computational models of synaptic plasticity. bioRxiv. 2018; p. 1–53.
9. Kim M, Huang T, Abel T, Blackwell KT. Temporal sensitivity of protein kinase A activation in late-phase long term potentiation. PLoS Computational Biology. 2010; 6. https://doi.org/10.1371/journal.pcbi.1000691 PMID: 20195498
10. Bhalla US, Iyengar R. Emergent properties of networks of biological signaling pathways. Science. 1999; 283:381–387. https://doi.org/10.1126/science.283.5400.381 PMID: 9888852
11. Shifman JM, Choi MH, Mihalas S, Mayo SL, Kennedy MB. Ca2+/calmodulin-dependent protein kinase II (CaMKII) is activated by calmodulin with two bound calciums. Proceedings of the National Academy of Sciences. 2006; 103(38):13968–13973. https://doi.org/10.1073/pnas.0606433103 PMID: 16966599
12. Pepke S, Kinzer-Ursem T, Mihalas S, Kennedy MB. A dynamic model of interactions of Ca2+, calmodulin, and catalytic subunits of Ca2+/calmodulin-dependent protein kinase II. PLoS Computational Biology. 2010; 6. https://doi.org/10.1371/journal.pcbi.1000675 PMID: 20168991
13. Byrne MJ, Putkey JA, Waxham MN, Kubota Y. Dissecting cooperative calmodulin binding to CaM kinase II: A detailed stochastic model. Journal of Computational Neuroscience. 2009; 27:621–638. https://doi.org/10.1007/s10827-009-0173-3 PMID: 19609660
14. Stefan MI, Edelstein SJ, Novère NL. An allosteric model of calmodulin explains differential activation of PP2B and CaMKII. Proceedings of the National Academy of Sciences. 2008; 105(31):10768–10773. https://doi.org/10.1073/pnas.0804672105 PMID: 18669651
15. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics); 2009.
16. Deng L. The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine. 2012; 29(6):141–142. https://doi.org/10.1109/MSP.2012.2211477
17. Gutenkunst RN, Waterfall JJ, Casey FP, Brown KS, Myers CR, Sethna JP. Universally sloppy parameter sensitivities in systems biology models. PLoS Computational Biology. 2007; 3:1871–1878. https://doi.org/10.1371/journal.pcbi.0030189 PMID: 17922568
18. Pharris MC, Patel NM, Kinzer-Ursem TL. Competitive Tuning Among Ca2+/Calmodulin-Dependent Proteins: Analysis of In Silico Model Robustness and Parameter Variability. Cellular and Molecular Bioengineering. 2018; 11:353–365. https://doi.org/10.1007/s12195-018-0549-4 PMID: 31105797
19. Romano DR, Pharris MC, Patel NM, Kinzer-Ursem TL. Competitive tuning: Competition's role in setting the frequency-dependence of Ca2+-dependent proteins. PLoS Computational Biology. 2017; 13. https://doi.org/10.1371/journal.pcbi.1005820 PMID: 29107982

20. Ordyan M, Bartol T, Kennedy M, Rangamani P, Sejnowski T. Interactions between calmodulin and neurogranin govern the dynamics of CaMKII as a leaky integrator. PLoS Computational Biology. 2020; 16. https://doi.org/10.1371/journal.pcbi.1008015 PMID: 32678848

21. Kim M, Park AJ, Havekes R, Chay A, Guercio LA, Oliveira RF, et al. Colocalization of protein kinase a with adenylyl cyclase enhances protein kinase a activity during induction of Long-Lasting Long-Term-Potentiation. PLoS Computational Biology. 2011; 7. https://doi.org/10.1371/journal.pcbi.1002084 PMID: 21738458

22. Nair AG, Bhalla US, Kotaleski JH. Role of DARPP-32 and ARPP-21 in the Emergence of Temporal Constraints on Striatal Calcium and Dopamine Integration. PLoS Computational Biology. 2016; 12. https://doi.org/10.1371/journal.pcbi.1005080 PMID: 27584878

23. Jedrzejewska-Szmek J, Luczak V, Abel T, Blackwell KT. beta-adrenergic signaling broadly contributes to LTP induction. PLoS Computational Biology. 2017; 13:1–32. https://doi.org/10.1371/journal.pcbi.1005657 PMID: 28742159

24. Rackauckas C, Ma Y, Noack A, Dixit V, Mogensen PK, Elrod C, et al. Accelerated Predictive Healthcare Analytics with Pumas, A High Performance Pharmaceutical Modeling and Simulation Platform. bioRxiv. 2022.

25. Lindstrom MJ, Bates DM. Nonlinear Mixed Effects Models for Repeated Measures Data. Biometrics. 1990; 46(3):673–687. https://doi.org/10.2307/2532087 PMID: 2242409

26. Akaike H. A new look at the statistical model identification. IEEE Transactions on Automatic Control. 1974; 19(6):716–723. https://doi.org/10.1109/TAC.1974.1100705

27. OGAWA Y, TANOKURA M. Calcium Binding to Calmodulin: Effects of Ionic Strength, Mg 2+, pH and Temperature 1. The Journal of Biochemistry. 1984; 95(1):19–28. https://doi.org/10.1093/oxfordjournals.jbchem.a134584 PMID: 6706907

28. Sivakumaran S, Hariharaputran S, Mishra J, Bhalla US. The Database of Quantitative Cellular Signaling: management and analysis of chemical kinetic models of signaling networks. Bioinformatics. 2003; 19(3):408–415. https://doi.org/10.1093/bioinformatics/btf860 PMID: 12584128

29. Malik-Sheriff RS, Glont M, Nguyen TVN, Tiwari K, Roberts MG, Xavier A, et al. BioModels—15 years of sharing computational models in life science. Nucleic Acids Research. 2020; 48(D1):D407–D415. https://doi.org/10.1093/nar/gkz1055 PMID: 31701150

30. Glont M, Nguyen TVN, Graesslin M, Hälke R, Ali R, Schramm J, et al. BioModels: expanding horizons to include more modelling approaches and formats. Nucleic Acids Research. 2018; 46(D1):D1248–D1253. https://doi.org/10.1093/nar/gkx1023 PMID: 29106614

31. Putkey JA, Waxham MN, Gaertner TR, Brewer KJ, Goldsmith M, Kubota Y, et al. Acidic/IQ motif regulator of calmodulin. Journal of Biological Chemistry. 2008; 283:1401–1410. https://doi.org/10.1074/jbc.M703831200 PMID: 17991744

32. Duchesne R, Guillemin A, Gandrillon O, Crauste F. Practical identifiability in the frame of nonlinear mixed effects models: the example of the in vitro erythropoiesis. BMC Bioinformatics. 2021; p. 1–21. https://doi.org/10.1186/s12859-021-04373-4 PMID: 34607573

33. Lee SY. Bayesian nonlinear models for repeated measurement data: an overview, implementation, and applications. Mathematics. 2022; 10(6). https://doi.org/10.3390/math10060898

34. Linkevicius D, Chadwick A, C Faas G, Stefan MI, C Sterratt D. Fitting and comparison of calcium-calmodulin kinetic schemes to a common data set using non-linear mixed effects modelling; 2025. Available from: https://figshare.com/collections/Fitting_and_comparison_of_calcium-calmodulin_kinetic_schemes_to_a_common_data_set_using_non-linear_mixed_effects_modelling/7614467/1.

35. Rackauckas C, Nie Q. Differentialequations.jl–a performant and feature-rich ecosystem for solving differential equations in julia. Journal of Open Research Software. 2017; 5(1):15. https://doi.org/10.5334/jors.151

36. Wang Y. Derivation of various NONMEM estimation methods. Journal of Pharmacokinetics and Pharmacodynamics. 2007; 34:575–593. https://doi.org/10.1007/s10928-007-9060-6 PMID: 17620001

37. Ding J, Tarokh V, Yang Y. Model Selection Techniques: An Overview. IEEE Signal Processing Magazine. 2018; 35:16–34. https://doi.org/10.1109/MSP.2018.2867638

38. Sawilowsky SS. Very large and huge effect sizes. Journal of Modern Applied Statistical Methods. 2009; 8:597–599. https://doi.org/10.22237/jmasm/1257035100

39. Bayazitov IT, Richardson RJ, Fricke RG, Zakharenko SS. Slow presynaptic and fast postsynaptic components of compound long-term potentiation. Journal of Neuroscience. 2007; 27:11510–11521. https://doi.org/10.1523/JNEUROSCI.3077-07.2007 PMID: 17959794

40. Sabatini BL, Oertner TG, Svoboda K. The life cycle of Ca2+ ions in dendritic spines. Neuron. 2002; 33:439–452. https://doi.org/10.1016/S0896-6273(02)00573-1 PMID: 11832230

41. Kakiuchi S, Yasuda S, Yamazaki R, Teshima Y, Kanda K, Kakiuchi R, et al. Quantitative Determinations of Calmodulin in the Supernatant and Particulate Fractions of Mammalian Tissues1. The Journal of Biochemistry. 1982; 92(4):1041–1048. https://doi.org/10.1093/oxfordjournals.jbchem.a134019 PMID: 7174634

42. Feinberg M. Foundations of Chemical Reaction Network Theory; 2019.

43. Hayer A, Bhalla US. Molecular switches at the synapse emerge from receptor and kinase traffic. PLoS Computational Biology. 2005; 1:0137–0154. https://doi.org/10.1371/journal.pcbi.0010020 PMID: 16110334

44. Quintana AR, Wang D, Forbes JE, Waxham MN. Kinetics of calmodulin binding to calcineurin. Biochemical and Biophysical Research Communications. 2005; 334:674–680. https://doi.org/10.1016/j.bbrc.2005.06.152 PMID: 16009337

45. Goraya TA, Cooper DMF. Ca2+-calmodulin-dependent phosphodiesterase (PDE1): Current perspectives; 2005.

46. Masada N, Schaks S, Jackson SE, Sinz A, Cooper DMF. Distinct mechanisms of calmodulin binding and regulation of adenylyl cyclases 1 and 8. Biochemistry. 2012; 51:7917–7929. https://doi.org/10.1021/bi300646y PMID: 22971080

47. Zhong L, Cherry T, Bies CE, Florence MA, Gerges NZ. Neurogranin enhances synaptic strength through its interaction with calmodulin. EMBO Journal. 2009; 28:3027–3039. https://doi.org/10.1038/emboj.2009.236 PMID: 19713936

48. Zhong L, Gerges NZ. Neurogranin targets calmodulin and lowers the threshold for the induction of long-term potentiation. PLoS ONE. 2012; 7. https://doi.org/10.1371/journal.pone.0041275 PMID: 22848456

49. Kim BH, Hawes SL, Gillani F, Wallace LJ, Blackwell KT. Signaling Pathways Involved in Striatal Synaptic Plasticity are Sensitive to Temporal Pattern and Exhibit Spatial Specificity. PLoS Computational Biology. 2013; 9. https://doi.org/10.1371/journal.pcbi.1002953 PMID: 23516346

50. Nair AG, Gutierrez-Arenas O, Eriksson O, Jauhiainen A, Blackwell KT, Kotaleski JH. Modeling Intracellular Signaling Underlying Striatal Function in Health and Disease. vol. 123. 1st ed. Elsevier Inc.; 2014. Available from: https://doi.org/10.1016/B978-0-12-397897-4.00013-9

51. Chay A, Zamparo I, Koschinski A, Zaccolo M, Blackwell KT. Control of betaAR- and N-methyl-D-aspartate (NMDA) Receptor-Dependent cAMP Dynamics in Hippocampal Neurons. PLoS Computational Biology. 2016; 12. https://doi.org/10.1371/journal.pcbi.1004735 PMID: 26901880

52. Pharris MC, Patel NM, VanDyk TG, Bartol TM, Sejnowski TJ, Kennedy MB, et al. A multi-state model of the CaMKII dodecamer suggests a role for calmodulin in maintenance of autophosphorylation. PLOS Computational Biology. 2019; 15(12):1–26. https://doi.org/10.1371/journal.pcbi.1006941 PMID: 31869343

53. Vanscyoc WS, Sorensen BR, Rusinova E, Laws WR, Ross JBA, Shea MA. Calcium Binding to Calmodulin Mutants Monitored by Domain-Specific Intrinsic Phenylalanine and Tyrosine Fluorescence; 2002. https://doi.org/10.1016/S0006-3495(02)75286-7 PMID: 12414709

54. Piazza M, Taiakina V, Dieckmann T, Guillemette JG. Structural Consequences of Calmodulin EF Hand Mutations. Biochemistry. 2017; 56:944–956. https://doi.org/10.1021/acs.biochem.6b01296 PMID: 28121131

55. Viswan NA, HarshaRani GV, Stefan MI, Bhalla US. FindSim: a framework for integrating neuronal data and signaling models. Frontiers in neuroinformatics. 2018; 12:38. https://doi.org/10.3389/fninf.2018.00038 PMID: 29997492

56. Danos V, Feret J, Fontana W, Harmer R, Krivine J. Rule-Based Modelling, Symmetries, Refinements. In: Fisher J, editor. Formal Methods in Systems Biology. Berlin, Heidelberg: Springer Berlin Heidelberg; 2008. p. 103–122.

57. Eriksson O, Jauhiainen A, Sasane SM, Kramer A, Nair AG, Sartorius C, et al. Uncertainty quantification, propagation and characterization by Bayesian analysis combined with global sensitivity analysis applied to dynamical intracellular pathway models. Bioinformatics. 2019; 35:284–292. https://doi.org/10.1093/bioinformatics/bty607 PMID: 30010712

58. Raue A, Steiert B, Schelker M, Kreutz C, Maiwald T, Hass H, et al. Data2Dynamics: A modeling environment tailored to parameter estimation in dynamical systems. Bioinformatics. 2015; 31:3558–3560. https://doi.org/10.1093/bioinformatics/btv405 PMID: 26142188

59. Ribeiro AH, Tiels K, Umenberger J, Schön TB, Aguirre LA. On the smoothness of nonlinear system identification. Automatica. 2020; 121. https://doi.org/10.1016/j.automatica.2020.109158

S1 APPENDIX. SUBSET OF DATA USED AND ITS SPLITTING INTO TRAINING, VALIDATION AND TESTING DATA SETS

The subset of data that we use from Faas et al. (2011) contains 94 recorded time series across 7 groups of initial conditions with varying numbers of time series between the initial conditions. Each time series contains data on fluorescence change divided by basal fluorescence $\Delta F/F_0$. Each time series is a record of either 35.204 ms (for group A) or 39.604 ms (all other groups) of $\Delta F/F_0$ after laser uncaging. Each time series is made up of either 258 (group A) or 260 (all other groups) points that were sampled unevenly – most points were focused in the first 4 ms of the time series. The first recorded time point after laser uncaging was at 0.160 ms.

From the full 7 groups we removed two time series from group C because the data in them were numerically identical but showed as having two different Pockels cell density values. This left 92 time series for us to use. We split the 92 time series into training, validation and test data sets. The training data set consisted of 7 randomly selected (without replacement) samples from each of the different experimental conditions, leaving 45 time series. The validation data set consisted of time series taken from the remaining 47 by randomly selecting 2–4 time series (without replacement) per experimental condition (due to imbalance in the group sizes). The remaining data was left for final model testing. We repeated this splitting procedure for 20 different seeds in order to counteract the small data set size.

We sub-sampled the first 2.424 ms of each time-series due to heavy over-sampling of this time period. As shown in Ribeiro et al. (2020), the larger the number of points in a time series used when fitting parameters of dynamical systems, the more complex the loss surface. Therefore, sub-sampling was done to improve the gradient-based optimization done by `Pumas.jl`. Namely, the original set of points $\{d_i\}_{i=1}^{201}$ was uniformly sub-sampled to $\{d_{1+20i}\}_{i=0}^{10}$. The comparison of original and sub-sampled data is shown in Figure 5 below.

S2 APPENDIX. CONCENTRATIONS OF SPECIES OF DIFFERENT GROUPS OF SOLUTIONS USED IN FAAS ET AL. DATA

Table 1 below shows the initial concentrations for all groups. There are some differences from those provided in the supplemental information in Faas et al. (2011), which Guido Faas communicated in response to queries.

	A	B	C	D	E	F	G
DMn	5.56mM	3.64mM	3.64mM	3.64mM	3.64mM	3.64mM	3.64mM
OGB-5N	50µM	100µM	100µM	100µM	100µM	100µM	100µM
Ca^{2+}_{Free}	1.88µM	1.10µM	0.73µM	0.255µM	0.41µM	399µM	394µM
CaM	123µM	143µM	72µM	187µM	140µM	94µM	47µM

Table 1: Initial conditions for the 7 experimental groups from Faas et al. (2011).

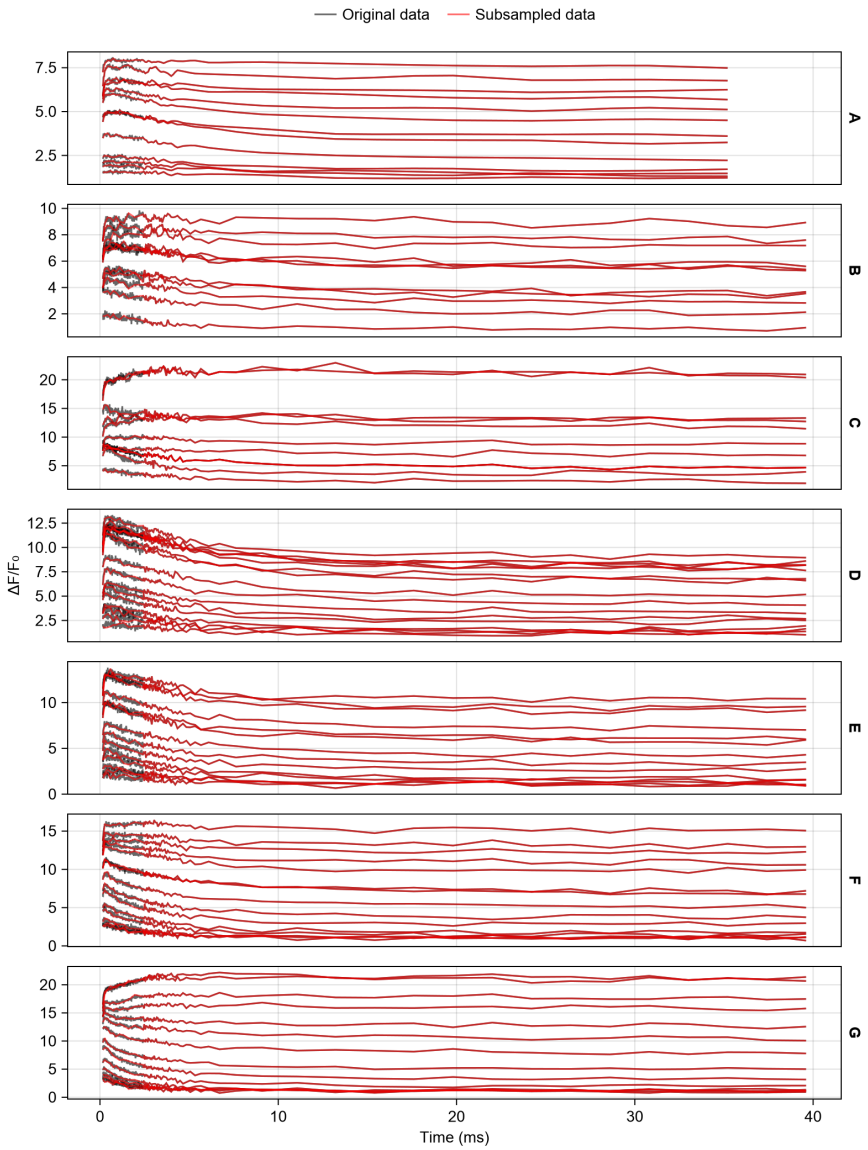


Figure 5: S1 Fig. Comparison of the original data set (black lines) and the data with subsampled initial period (red lines).

S4 APPENDIX. DERIVATION OF STEADY-STATE REACTION RATE CONSTANTS FOR SCHEME 4

Since we do not need to track the full CaM species (since it does not bind anything downstream) we consider lobes individually, namely

CaM0, CaM1x, CaM2x where x can be either the N or C lobe. This results in a three dimensional ODE system with 8 reaction rate constants (4 per lobe) $k_i{}^x$, $i \in (1, 2, 3, 4)$, $x \in (C, N)$. The resulting ODE system for a single lobe is

$$\frac{d[CaM0x]}{dt} = -k_1^x[CaM0x][Ca] + k_2^x[CaM1x] \tag{31}$$

$$\frac{d[CaM1x]}{dt} = k_1^x[CaM0x][Ca] - k_2^x[CaM1x] - k_3^x[CaM1x][Ca] + k_4^x[CaM2x] \tag{32}$$

$$\frac{d[CaM2x]}{dt} = k_3^x[CaM1x][Ca] - k_4^x[CaM2x] \tag{33}$$

The quasi-steady state approximation means $\frac{d[CaM1x]}{dt} = 0$, therefore we can express $[CaM1x]$ via other terms

$$0 = k_1^x[CaM0x][Ca] - k_2^x[CaM1x] \tag{34}$$

$$- k_3^x[CaM1x][Ca] + k_4^x[CaM2x] \tag{35}$$

$$k_2^x[CaM1x] + k_3^x[CaM1x][Ca] = k_1^x[CaM0x][Ca] + k_4^x[CaM2x] \tag{36}$$

$$[CaM1x](k_2^x + k_3^x[Ca]) = k_1^x[CaM0x][Ca] + k_4^x[CaM2x] \tag{37}$$

$$[CaM1x] = \frac{1}{k_2^x + k_3^x[Ca]}(k_1^x[CaM0x][Ca] + k_4^x[CaM2x]) \tag{38}$$

and then substitute back into equations for $\frac{d[CaM0x]}{dt}$ and $\frac{d[CaM2x]}{dt}$:

$$\frac{d[CaM0x]}{dt} = -k_1^x[CaM0x][Ca] \tag{39}$$

$$+ \frac{k_2^x}{k_2^x + k_3^x[Ca]}(k_1^x[CaM0x][Ca] + k_4^x[CaM2x]) \tag{40}$$

$$\frac{d[CaM2x]}{dt} = \frac{k_3^x[Ca]}{k_2^x + k_3^x[Ca]}(k_1^x[CaM0x][Ca] + k_4^x[CaM2x]) - k_4^x[CaM2x] \tag{41}$$

which after multiplying through, gathering the terms and some simplification becomes:

$$\frac{d[CaM0x]}{dt} = -\frac{k_1^x k_3^x}{k_2^x + k_3^x[Ca]}[CaM0x][Ca]^2 + \frac{k_2^x k_4^x}{k_2^x + k_3^x[Ca]}[CaM2x] \tag{42}$$

$$\frac{d[CaM2x]}{dt} = \frac{k_1^x k_3^x}{k_2^x + k_3^x[Ca]}[CaM0x][Ca]^2 - \frac{k_2^x k_4^x}{k_2^x + k_3^x[Ca]}[CaM2x] \tag{43}$$

which if we set

$$k_{ss}^{fx}([Ca]) = \frac{k_1^x k_3^x}{k_2^x + k_3^x[Ca]} \tag{44}$$

$$k_{ss}^{bx}([Ca]) = \frac{k_2^x k_4^x}{k_2^x + k_3^x[Ca]} \tag{45}$$

makes the reduced system

$$\frac{d[CaM0x]}{dt} = -k_{ss}^{fx}([Ca])[CaM0x][Ca]^2 + k_{ss}^{bx}([Ca])[CaM2x] \tag{46}$$

$$\frac{d[CaM2x]}{dt} = k_{ss}^{fx}([Ca])[CaM0x][Ca]^2 - k_{ss}^{bx}([Ca])[CaM2x] \tag{47}$$

easy to interpret as a $CaM0x + 2Ca \Longleftrightarrow CaM2x$ with Ca^{2+}-dependent reaction rates for Ca^{2+} binding to lobe $x \in (C, N)$.

S5 APPENDIX. PUBLISHED REACTION RATE CONSTANTS

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M \times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM + 2Ca \Longleftrightarrow CaM_2$	3.60	-2.10	-5.70
$CaM_2 + 2Ca \Longleftrightarrow CaM_4$	5.0	0.04	-4.96

Table 2: Original parameters from Putkey et al. (2008) for Scheme 1 that were used as a basis for Kim et al. (2010) reaction rate constants (in $\log_{10}$ scale).

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M \times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM + 2Ca \Longleftrightarrow CaM_2$	4.86	-1.42	-6.00
$CaM_2 + Ca \Longleftrightarrow CaM_3$	3.56	-2.00	-5.55
$CaM_3 + Ca \Longleftrightarrow CaM_4$	2.67	-2.01	-4.68

Table 3: Scheme 2 parameters used in Bhalla and Iyengar (1999) (in $\log_{10}$ scale).

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M \times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM + Ca \Longleftrightarrow CaM_1$	-	-	-5.10
$CaM_1 + Ca \Longleftrightarrow CaM_2$	-	-	-5.77
$CaM_2 + Ca \Longleftrightarrow CaM_3$	-	-	-4.46
$CaM_3 + Ca \Longleftrightarrow CaM_4$	-	-	-5.05

Table 4: Scheme 3 parameters from Shifman et al. (2006) (in $\log_{10}$ scale).

S6 APPENDIX. OUR PARAMETER FITS FOR ALL SCHEMES

All of our parameter fits can be downloaded as a spreadsheet at the github repository https://github.com/dom-linkevicius/FaasCalmodulin.jl/.

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M\times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM + Ca \Longleftrightarrow CaM_{1C}$	3.60	-1.40	-5.00
$CaM_{1C} + Ca \Longleftrightarrow CaM_{2C}$	4.00	-2.03	-6.03
$CaM + Ca \Longleftrightarrow CaM_{1N}$	5.00	0.40	-4.60
$CaM_{1N} + Ca \Longleftrightarrow CaM_{2N}$	5.18	-0.12	-5.30

Table 5: Scheme 4 and 5 parameters from Pepke et al. (2010) (in $\log_{10}$ scale).

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M\times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM + Ca \Longleftrightarrow CaM_{1C}$	4.90	0.30	-4.60
$CaM_{1C} + Ca \Longleftrightarrow CaM_{2C}$	4.40	-2.20	-6.60
$CaM + Ca \Longleftrightarrow CaM_{1N}$	5.90	2.20	-3.70
$CaM_{1N} + Ca \Longleftrightarrow CaM_{2N}$	7.50	1.40	-6.10

Table 6: Scheme 5 parameters from Faas et al. (2011) (in $\log_{10}$ scale).

Reaction	$\log_{10}(\mathbf{k_f}\frac{1}{M\times ms})$	$\log_{10}(\mathbf{k_f}\frac{1}{ms})$	$\log_{10}(\mathbf{K_d}M)$
$CaM_{0N} + Ca \Longleftrightarrow CaM_{1N}$	5.44	0.96	-4.48
$CaM_{0N} + Ca \Longleftrightarrow CaM_{2N}$	5.44	-1.80	-3.64
$CaM_{1N} + Ca \Longleftrightarrow CaM_{3N}$	5.71	0.24	-5.46
$CaM_{2N} + Ca \Longleftrightarrow CaM_{3N}$	5.70	-0.60	-6.30
$CaM_{0C} + Ca \Longleftrightarrow CaM_{1C}$	5.44	0.70	-4.73
$CaM_{0C} + Ca \Longleftrightarrow CaM_{2C}$	5.44	1.50	-3.94
$CaM_{1C} + Ca \Longleftrightarrow CaM_{3C}$	3.57	-2.85	-6.42
$CaM_{2C} + Ca \Longleftrightarrow CaM_{3C}$	5.07	-2.14	-7.22

Table 7: Scheme 6 parameters from Byrne et al. (2009) (in $\log_{10}$ scale).

i	k_1	K_{D_1}	k_2	K_{D_2}
Seed 1	8.49	-9.0	8.99	-9.0
Seed 2	2.02	-9.0	7.13	-4.99
Seed 3	6.72	-9.0	9.0	-8.98
Seed 4	8.1	-9.0	3.92	-9.0
Seed 5	8.28	-9.0	3.98	-9.0
Seed 6	8.37	-9.0	6.42	-9.0
Seed 7	8.2	-9.0	4.03	-9.0
Seed 8	8.66	-9.0	7.75	-9.0
Seed 9	8.2	-9.0	4.05	-9.0
Seed 10	8.09	-9.0	3.76	-9.0
Seed 11	8.19	-9.0	8.03	-5.54
Seed 13	3.07	-9.0	6.79	-8.92
Seed 14	8.1	-9.0	7.83	-5.79
Seed 15	8.03	-9.0	4.79	-9.0
Seed 16	8.14	-9.0	2.56	-8.94
Seed 17	8.24	-9.0	5.3	-9.0
Seed 18	8.88	-9.0	7.85	-9.0
Seed 19	8.05	-9.0	7.21	-5.16
Seed 20	8.22	-9.0	6.88	-7.17

Table 8: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 1 for all $i \in \{1 \dots 20\}$ seeds. Seed 12 was removed due to training failures.

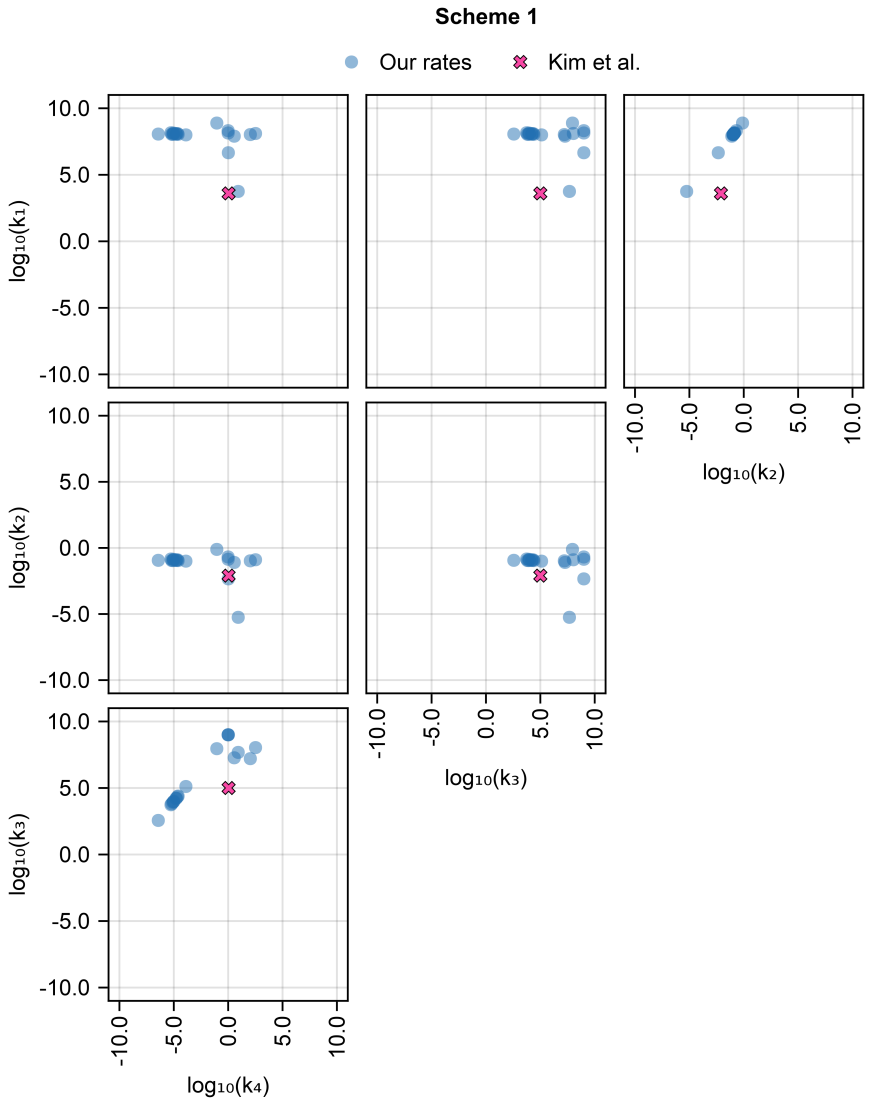


Figure 6: Pair plots for our reaction rate constants for Scheme 1 (blue dots) for all random seeds, along with the published reaction rate constants from Kim et al. (2010) (all in $\log_{10}$ scale).

i	k_1	K_{D_1}	k_2	K_{D_2}	k_3	K_{D_3}
Seed 1	8.18	-9.0	2.04	-8.53	4.67	-4.03
Seed 2	2.03	-9.0	6.6	-9.0	4.22	-6.21
Seed 3	6.43	-9.0	9.0	-9.0	4.38	-6.36
Seed 4	8.07	-9.0	2.07	-9.0	4.58	-6.42
Seed 5	8.84	-5.96	2.08	-8.87	4.46	-8.85
Seed 6	8.24	-9.0	2.41	-9.0	4.56	-6.41
Seed 7	6.52	-9.0	6.15	-9.0	4.41	-6.29
Seed 8	9.0	-8.99	7.69	-5.27	2.87	-9.0
Seed 9	8.38	-9.0	2.52	-8.98	2.92	-4.03
Seed 10	6.51	-9.0	6.35	-9.0	4.39	-6.37
Seed 11	6.72	-9.0	6.43	-9.0	4.5	-6.4
Seed 12	7.5	-9.0	5.99	-8.34	5.29	-9.0
Seed 13	6.44	-9.0	6.41	-9.0	4.35	-6.37
Seed 14	6.7	-9.0	7.36	-9.0	4.48	-6.37
Seed 15	6.62	-9.0	6.53	-9.0	4.42	-6.35
Seed 16	8.16	-9.0	2.07	-8.98	4.13	-4.1
Seed 17	7.97	-8.77	6.66	-8.8	8.4	-8.88
Seed 18	6.61	-9.0	6.5	-9.0	4.44	-6.39
Seed 19	8.09	-9.0	2.02	-6.18	8.75	-8.86
Seed 20	6.54	-9.0	6.29	-9.0	4.42	-6.36

Table 9: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 2 for all $i \in \{1 \dots 20\}$ seeds.

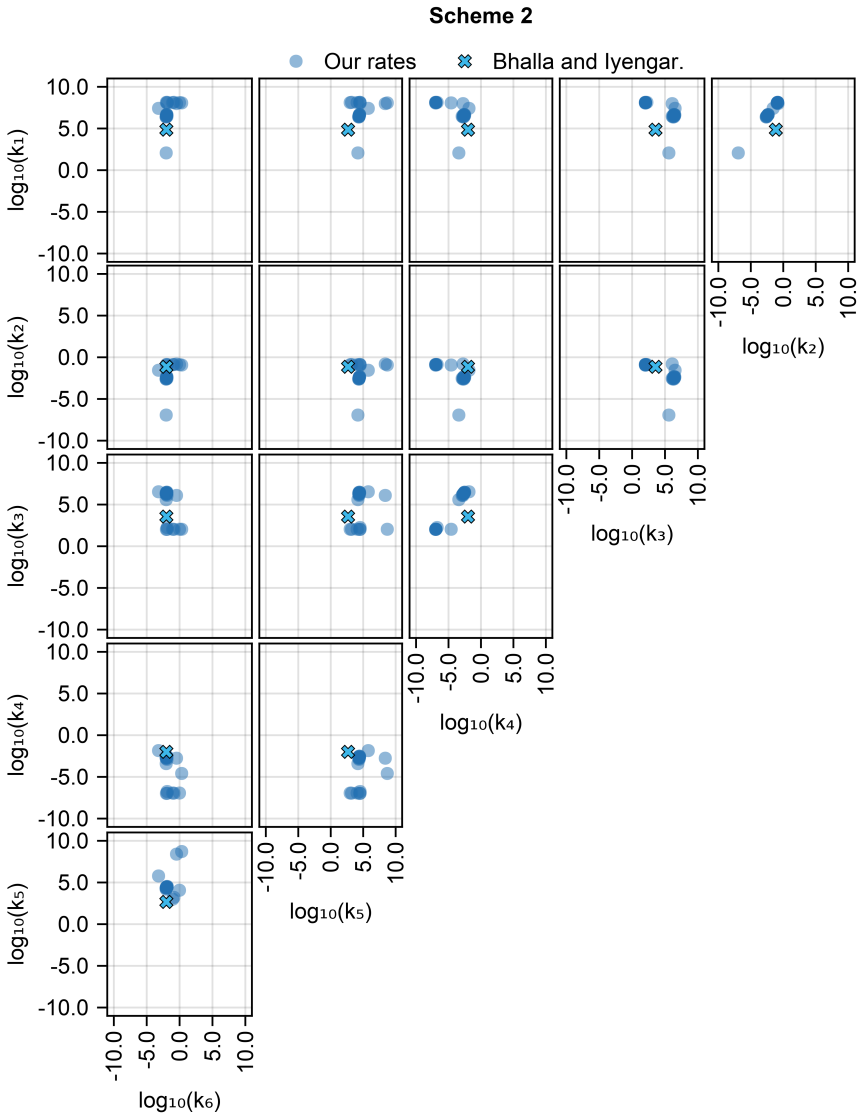


Figure 7: Pair plots for our reaction rate constants for Scheme 2 (blue dots) for all random seeds, along with the published reaction rate constants from Bhalla and Iyengar (1999) (all in $\log_{10}$ scale).

i	k_1	K_{D_1}	k_2	K_{D_2}	k_3	K_{D_3}	k_4	K_{D_4}
Seed 1	4.28	-6.26	5.81	-4.83	7.67	-3.61	7.91	-5.74
Seed 2	2.1	-3.23	3.97	-9.44	4.38	-6.27	5.4	-5.3
Seed 3	4.2	-6.46	5.47	-5.47	7.68	-2.8	8.53	-6.26
Seed 4	5.94	-4.83	6.0	-5.77	4.66	-6.05	3.17	-4.15
Seed 5	5.96	-7.85	4.21	-6.43	7.9	-3.09	6.94	-6.63
Seed 6	6.77	-4.7	5.78	-6.03	4.36	-5.69	6.78	-4.01
Seed 7	4.29	-6.26	5.67	-4.98	7.55	-4.21	7.37	-5.08
Seed 8	4.32	-6.62	5.23	-4.79	2.07	-3.38	6.36	-6.1
Seed 9	4.22	-6.33	5.54	-5.19	7.53	-3.4	7.84	-5.76
Seed 10	5.76	-5.84	4.5	-5.64	2.36	-6.83	5.51	-2.39
Seed 11	4.28	-6.2	5.83	-4.75	7.67	-3.3	8.2	-6.01
Seed 12	4.27	-6.32	7.71	-3.17	7.08	-6.08	2.18	-4.96
Seed 13	2.01	-6.41	5.64	-5.77	4.6	-6.94	3.28	-3.98
Seed 14	7.18	-3.36	6.97	-7.52	4.41	-5.94	6.87	-4.18
Seed 15	4.35	-6.64	5.31	-4.83	2.11	-3.26	6.53	-6.19
Seed 16	5.46	-5.09	2.59	-8.02	4.36	-6.1	5.53	-4.67
Seed 17	5.6	-7.06	4.21	-6.42	7.71	-3.06	7.05	-6.59
Seed 18	4.12	-6.34	5.59	-4.77	7.59	-3.3	7.92	-6.03
Seed 19	5.38	-4.89	2.74	-8.22	4.37	-6.1	5.57	-4.48
Seed 20	4.32	-6.29	7.73	-3.43	6.9	-5.89	2.29	-4.86

Table 10: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 3 for all $i \in \{1 \dots 20\}$ seeds.

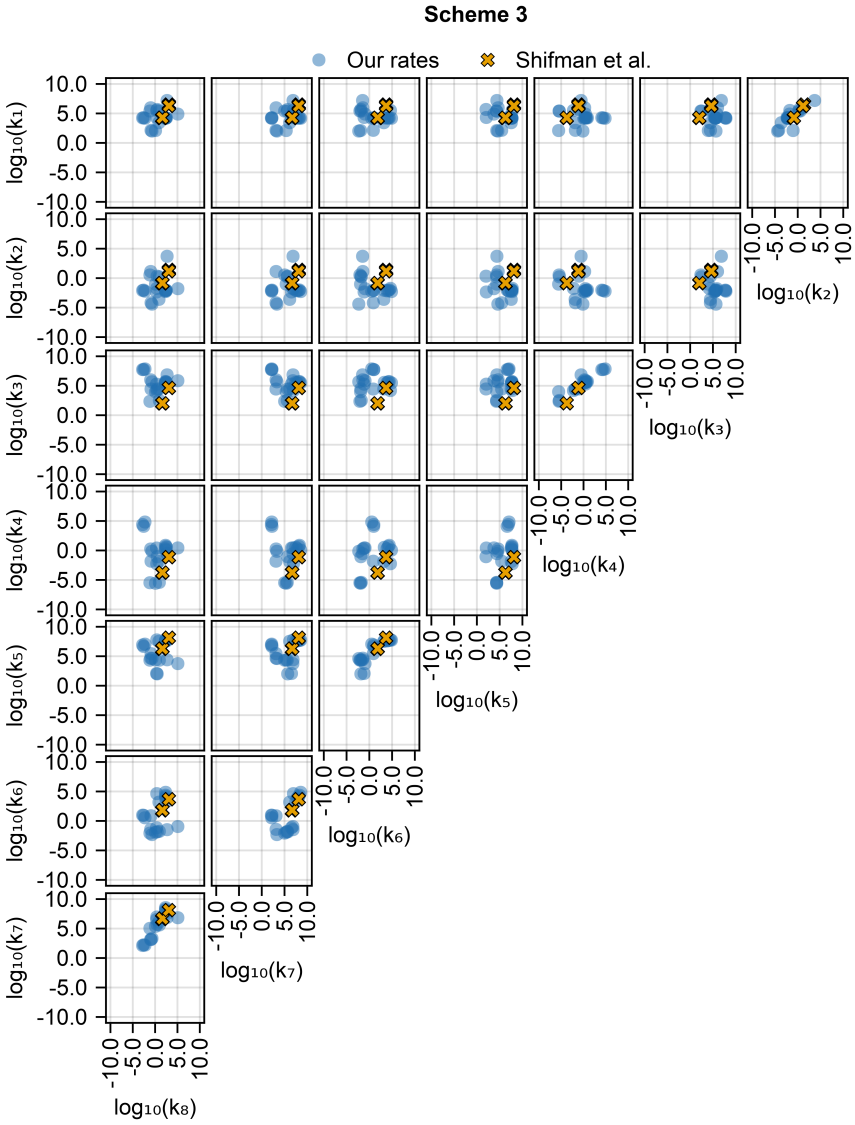


Figure 8: Pair plots for our reaction rate constants for Scheme 3 (blue dots) for all random seeds, along with the published reaction rate constants from Shifman et al. (2006) (all in $\log_{10}$ scale).

i	k_5	K_{D_5}	k_7	K_{D_7}	k_1	K_{D_1}	k_3	K_{D_3}
Seed 1	6.52	-4.18	6.4	-5.67	4.38	-5.28	5.5	-6.2
Seed 2	6.16	-4.13	6.3	-5.73	4.35	-5.21	5.26	-6.37
Seed 3	2.1	-4.06	7.51	-4.44	6.9	-5.32	4.02	-7.73
Seed 4	6.68	-4.07	6.31	-5.66	4.35	-5.1	5.19	-6.33
Seed 5	5.85	-4.27	6.46	-5.7	4.38	-5.37	5.48	-6.31
Seed 6	4.34	-5.57	7.5	-5.98	5.9	-4.69	5.89	-5.23
Seed 7	6.09	-4.17	6.37	-5.69	4.37	-5.22	5.42	-6.22
Seed 8	6.12	-4.13	6.31	-5.71	4.32	-5.34	5.36	-6.4
Seed 9	4.35	-5.51	7.36	-6.04	6.37	-4.62	5.8	-5.24
Seed 10	6.92	-3.19	5.5	-5.71	2.59	-5.2	5.64	-6.02
Seed 11	6.59	-4.08	6.3	-5.67	4.37	-5.16	5.25	-6.33
Seed 12	6.93	-4.08	6.29	-5.68	4.35	-5.28	5.26	-6.44
Seed 13	6.15	-2.65	4.77	-5.77	7.0	-5.42	3.9	-7.95
Seed 14	7.12	-4.08	6.24	-5.74	4.35	-5.44	5.66	-6.23
Seed 15	6.86	-4.12	6.3	-5.72	4.34	-5.56	5.92	-6.07
Seed 16	5.8	-4.25	6.57	-5.57	4.33	-5.25	5.51	-6.17
Seed 17	5.86	-4.21	6.41	-5.69	4.36	-5.4	5.55	-6.27
Seed 18	4.93	-6.7	7.46	-7.14	6.06	-3.38	4.42	-5.38
Seed 19	6.25	-4.06	6.35	-5.64	4.31	-5.16	5.44	-6.58
Seed 20	7.03	-4.08	6.27	-5.71	4.36	-5.4	5.63	-6.19

Table 11: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 4 for all $i \in \{1 \dots 20\}$ seeds. Note that parameters are not sorted according to dissociation constants, therefore parameters indicated as being for the C lobe may be representative of the N lobe.

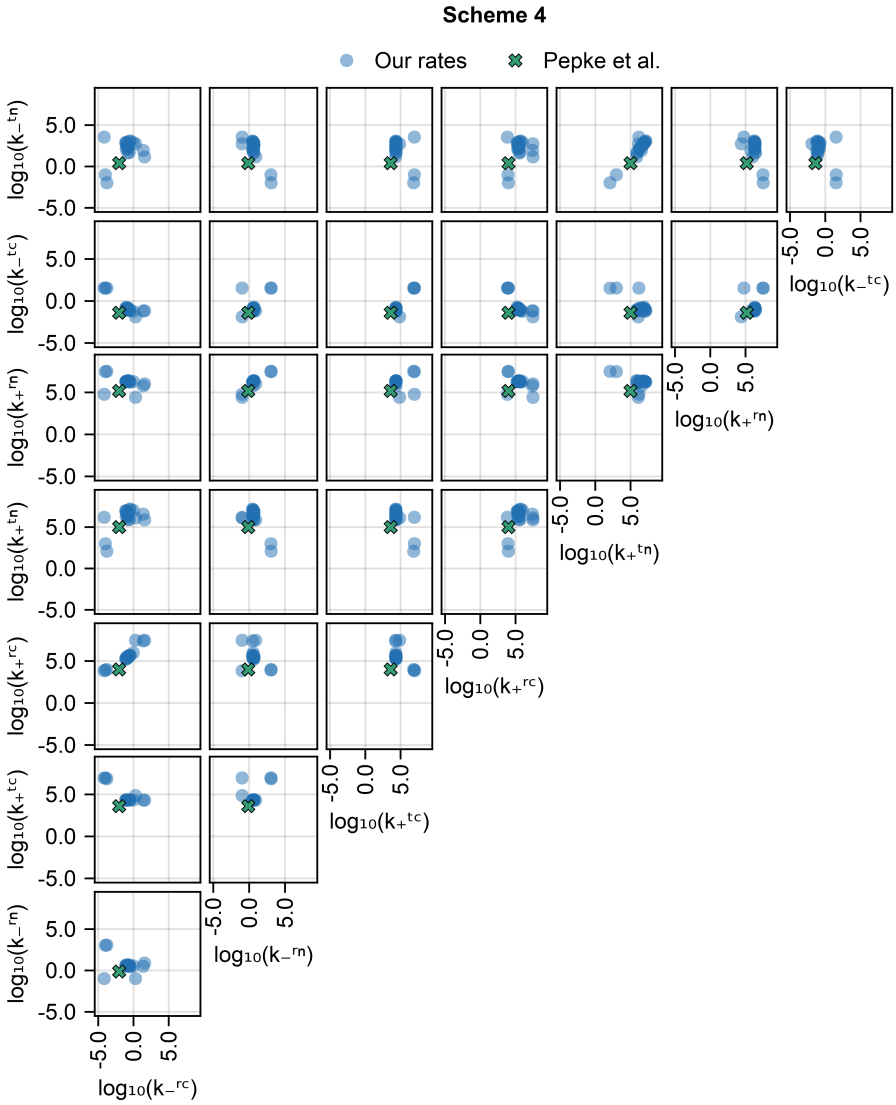


Figure 9: Pair plots for our reaction rate constants for Scheme 4 (blue dots) for all random seeds, along with the published reaction rate constants from Pepke et al. (2010) (all in $\log_{10}$ scale).

i	k_5	K_{D_5}	k_7	K_{D_7}	k_1	K_{D_1}	k_3	K_{D_3}
Seed 1	5.83	-2.82	7.51	-6.48	4.45	-5.21	4.06	-5.71
Seed 2	6.29	-2.69	7.19	-6.63	4.37	-5.1	4.09	-5.74
Seed 3	6.68	-3.05	7.03	-6.19	4.47	-5.19	4.07	-5.79
Seed 4	5.81	-2.51	7.45	-6.75	4.34	-5.18	3.99	-5.6
Seed 5	6.79	-2.44	7.34	-6.84	3.9	-5.17	5.62	-5.48
Seed 6	4.71	-5.9	5.89	-4.88	5.56	-4.93	4.2	-5.87
Seed 7	5.85	-2.69	7.29	-6.59	4.37	-5.13	4.03	-5.64
Seed 8	6.52	-2.63	7.26	-6.59	4.35	-5.07	4.02	-5.64
Seed 9	3.91	-5.16	5.47	-5.56	6.67	-2.38	7.4	-6.92
Seed 10	6.81	-2.29	7.53	-6.98	3.88	-5.23	5.4	-5.51
Seed 11	5.65	-3.0	7.1	-6.2	4.44	-5.24	4.04	-5.69
Seed 12	6.86	-2.26	7.63	-7.0	3.91	-5.35	5.26	-5.52
Seed 13	5.62	-2.93	7.52	-6.27	4.44	-5.11	4.02	-5.67
Seed 14	6.77	-2.6	7.34	-6.63	4.41	-5.24	4.01	-5.66
Seed 15	6.51	-2.64	7.38	-6.62	4.39	-5.19	3.99	-5.64
Seed 16	5.03	-6.53	5.66	-4.95	3.91	-5.27	2.32	-4.24
Seed 17	4.91	-6.54	5.7	-5.07	3.89	-5.4	1.98	-4.13
Seed 18	4.01	-5.6	7.51	-4.44	5.11	-4.4	2.42	-5.27
Seed 19	5.43	-1.77	8.53	-7.41	4.36	-5.24	3.94	-5.56
Seed 20	5.0	-6.37	5.81	-4.94	3.95	-5.44	2.29	-4.2

Table 12: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 5 for all $i \in \{1 \dots 20\}$ seeds. Note that parameters are not sorted according to dissociation constants, therefore parameters indicated as being for the C lobe may be more representative of the N lobe.

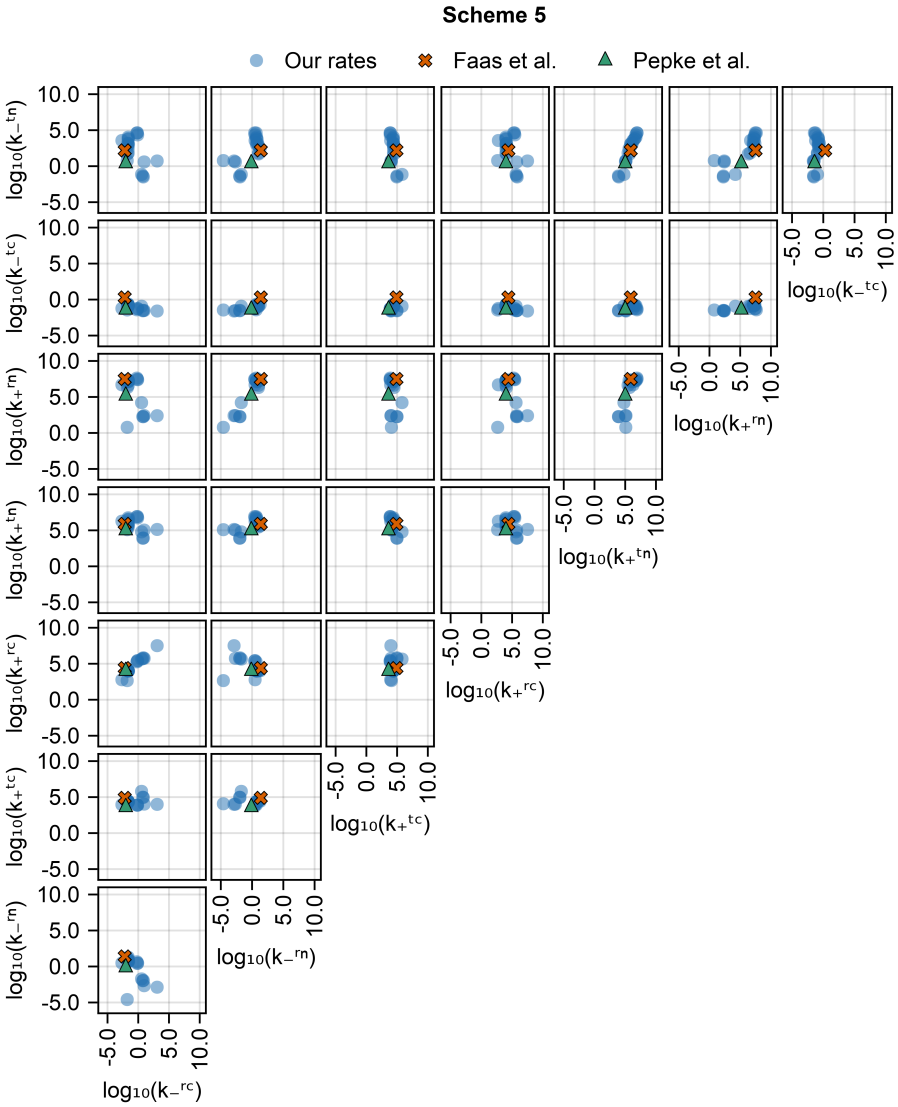


Figure 10: Pair plots for our reaction rate constants for Scheme 5 (blue dots) for all random seeds $i \in \{1 \ldots 20\}$ seeds, along with two sets of published reaction rate constants orange crosses (Faas et al., 2011) and green triangles (Pepke et al., 2010) (all in $\log_{10}$ scale).

i	k^n_{01}	k^n_{02}	k^n_{13}	k^n_{23}	$K^n_{D_{01}}$	$K^n_{D_{13}}$	$K^n_{D_{02}}$	k^c_{01}	k^c_{02}	k^c_{13}	k^c_{23}	$K^c_{D_{01}}$	$K^c_{D_{13}}$	$K^c_{D_{02}}$
1	3.29	6.26	5.25	7.63	-3.34	-5.91	-2.92	4.12	4.23	1.89	5.27	-4.46	-5.41	-6.32
2	6.55	4.37	7.07	5.78	-3.06	-6.2	-3.16	3.99	4.22	2.03	5.32	-4.34	-5.36	-6.43
3	3.72	6.73	3.39	7.3	-4.4	-4.78	-2.92	4.0	4.52	1.54	4.23	-5.45	-4.9	-5.24
4	5.33	6.21	5.86	7.47	-2.77	-6.47	-2.77	4.05	4.2	1.89	5.24	-4.44	-5.34	-6.3
5	6.81	3.86	7.23	3.47	-2.93	-6.36	-4.23	4.22	6.49	3.53	6.06	-5.32	-2.59	-5.23
6	3.9	3.87	5.32	5.47	-5.23	-5.42	-4.98	5.44	6.92	3.57	7.47	-2.39	-2.53	-6.93
7	3.42	6.66	5.23	7.35	-4.0	-5.27	-2.82	4.33	6.12	3.68	6.16	-5.4	-2.4	-5.21
8	4.29	6.65	5.42	7.24	-3.29	-5.92	-2.95	4.2	4.04	5.3	2.27	-5.31	-4.37	-5.38
9	3.63	6.74	5.12	7.33	-3.67	-5.58	-2.9	4.08	4.24	2.21	5.32	-4.41	-5.36	-6.39
10	3.13	6.42	3.33	7.31	-4.25	-4.95	-2.99	4.07	4.22	1.99	5.42	-4.36	-5.39	-6.32
11	5.97	5.36	7.28	5.72	-3.11	-6.06	-2.77	4.16	4.26	1.98	5.34	-4.49	-5.4	-6.37
12	5.32	6.69	5.87	7.46	-2.95	-6.23	-2.94	4.16	4.27	2.12	5.36	-4.42	-5.47	-6.47
13	5.25	6.61	5.98	7.26	-3.06	-6.18	-3.02	4.3	6.43	3.6	6.09	-5.34	-2.49	-5.26
14	4.16	4.26	2.53	5.34	-4.42	-6.42	-5.33	5.43	6.94	3.57	7.51	-2.59	-2.78	-6.63
15	3.07	6.68	5.2	7.56	-3.97	-5.31	-2.84	4.14	4.24	1.98	5.28	-4.4	-5.37	-6.39
16	4.03	4.18	2.67	5.34	-4.38	-6.27	-5.31	5.66	5.59	7.31	5.04	-3.26	-2.53	-5.96
17	3.24	6.68	3.51	7.39	-4.31	-4.99	-2.72	4.28	6.49	3.62	6.25	-5.3	-2.4	-5.25
18	5.67	5.4	7.33	5.61	-3.12	-6.12	-2.69	4.39	3.92	4.11	1.88	-5.12	-5.24	-5.54
19	3.8	4.16	4.61	6.49	-5.82	-4.01	-4.79	5.41	5.45	0.33	5.37	-4.62	-1.72	-5.11
20	4.17	4.28	2.52	5.36	-4.48	-6.41	-5.36	5.41	6.79	3.57	7.4	-2.72	-2.93	-6.5

Table 13: Our reaction rate fits (in $\log_{10}$ scale) for Scheme 6 for all $i \in \{1 \dots 20\}$ seeds. Note that parameters are not sorted according to dissociation constants, therefore parameters indicated as being for the C lobe may be more representative of the N lobe. Also, note that $K^c_{D_{23}}$ and $K^n_{D_{23}}$ are absent as they are not free parameters, but expressed via other dissociation constants due to microscopic reversibility constraints: $K^x_{D_{23}} = \frac{K^x_{D_{01}} K^x_{D_{13}}}{K^x_{D_{02}}}$.

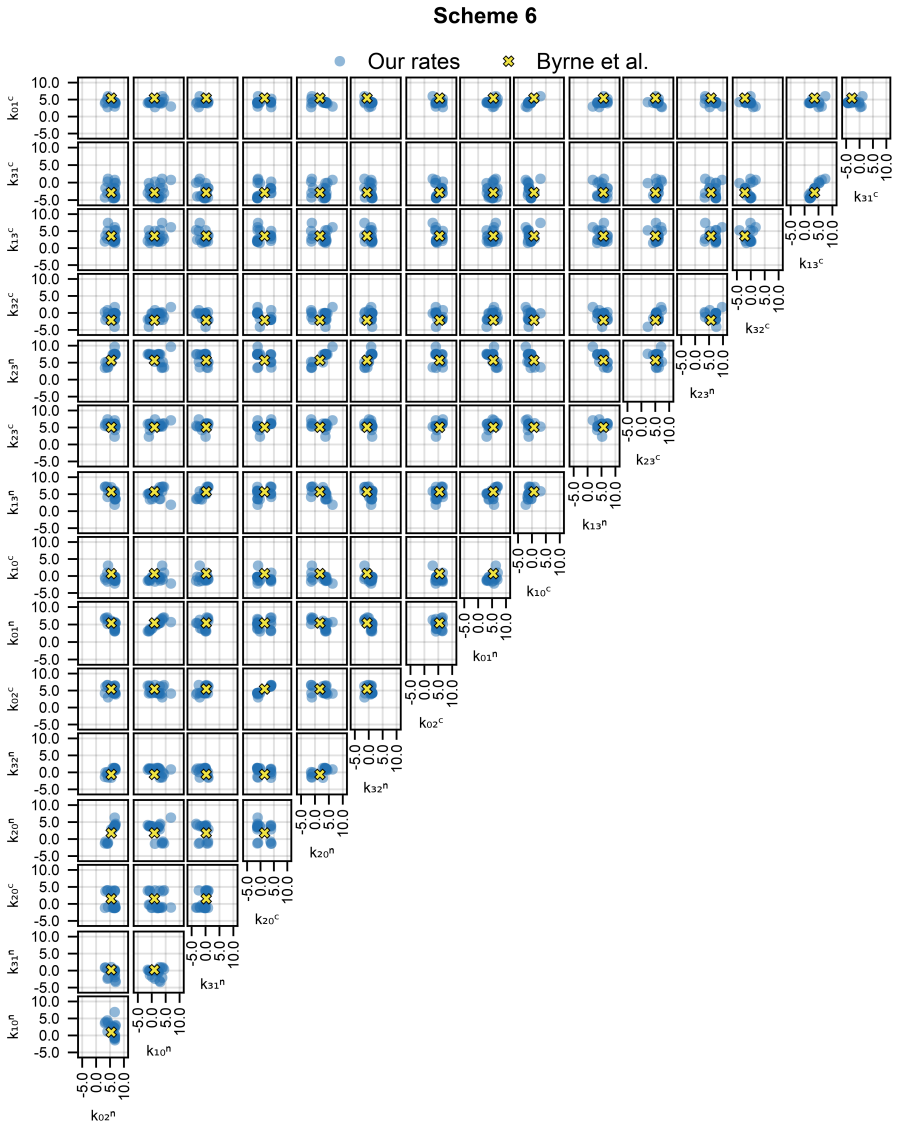


Figure 11: Pair plots for our reaction rate constants for Scheme 6 (blue dots) for all random seeds, along with the published reaction rate constants from Byrne et al. (2009) (all in $\log_{10}$ scale).

CONCLUSIONS

This chapter dealt with the challenge of fitting a chemical reaction network in the circumstances when the participant species and complexes are reasonably well constrained. The main challenge was finding reaction rate constants that would result in the best fit to the data given the structure of the chemical reaction network. Moreover, the data used in this chapter contained experimental conditions which showed significant variability and therefore required NLME modelling to account for.

Even with a relatively rich data set, the fitted parameters in some cases spanned 5 or more orders of magnitude between different seeds (Figures 6 to 11), implying that the data does not constrain the parameters sufficiently well. Therefore, additional data which would include genetic manipulations isolating the individual Calmodulin lobes may be necessary.

This chapter illustrates the benefit of comparing the chemical reaction models used in computational neuroscience, which is seldom done. The results showed that a number of previously used Ca^{2+}-Calmodulin schemes were not expressive enough to reproduce the experimental data (Faas et al., 2011; Shifman et al., 2006). Therefore, additional research into the downstream consequences of using a poorly performing Ca^{2+}-Calmodulin scheme may be informative, especially confirming the results of some publications which used them. One additional important Ca^{2+} signalling feature is its spatial distribution (Lee et al., 2009; Wang and Augustine, 2015). It is possible that models which include the spatial aspects of Ca^{2+} signalling may counteract some of the shortcomings of more simplified schemes. For example, see Figure 9 (in the publication) first column, where Scheme 1 does not produce any Ca^{2+} integration, whereas if the Ca^{2+} was more concentrated in a smaller nanodomain, the Ca^{2+} signal integration may have occurred.

Future work could build upon the results presented in this chapter, fitting Calmodulin-CaMKII interactions to data and comparing the performance of the newly fitted Ca^{2+}-Calmodulin-CaMKII network with respective schemes used in the literature. Finally, the work undertaken in this chapter bears implication for situations when chemical reaction networks are used as helpful abstractions – given that it is difficult to constrain a reasonably structurally well-known chemical reaction network if it only contains one observed entity, fitting a chemical reaction network model when the structure is not well-known is bound to pose an even greater challenge. In the next chapter on the modelling of the voltage-gated potassium channel gating, I delve deeper into the challenge of unknown underlying chemical reaction networks, and how it can be addressed with scientific machine learning.

4

MODELLING OF VOLTAGE-SENSITIVE POTASSIUM CHANNEL GATING

This chapter tackles the challenge of creating chemical reaction network models when there are few constraints of its states. This is particularly prevalent when the states are may be useful abstractions, not corresponding to real physical states. Specifically, chemical reaction networks of varying complexity are used to model the voltage gating of ion channels (Hodgkin and Huxley, 1952). Such models can contain multiple non-conducting, as well as open states, or gating particles, if a Hodgkin-Huxley (HH) formalism is used. However, only the dynamics of the open states are observable due to the conducted current. Therefore, the closed states, which are often the majority, can be difficult to constrain. Theoretical work shows that under certain assumptions (stationary scheme, low levels of noise), the underlying Markov schemes can be identified (Fredkin and Rice, 1985). However, these assumptions do not always hold and practical considerations need to be taken into account.

The work presented in this chapter showcases a way to address multiple different challenges in the modelling of voltage-gated ion channel gating. By using the recordings of 20 different K_v channels (Ranjan et al., 2019), scientific machine learning (SciML) and nonlinear mixed effects (NLME) modelling, I created K_v models that reproduced the data better than existing baselines. Moreover, the classical channel gating models require a single chemical reaction network per channel type. My results show that it is possible to use a single HH inspired model, with neural network-based equilibrium fraction and time constant functions, to create a unified SciML HH model that captures the gating dynamics of 20 different K_v channels. Importantly, the unified SciML HH model performs similarly to individual SciML HH Hodgkin-Huxley models fitted to individual channel data. Moreover, the unified SciML HH model exhibited significant imputation capabilities by accurately reproducing the data for K_v types that it was not trained on. These results are an important step forward in facilitating faster, more accurate, more general and streamlined creation of ion channel gating models.

The majority of this chapter is formed by the pre-print publication: Linkevicius D., Chadwick A., Stefan, M. I., & David C. Sterratt, D. C. (2025) 'One model to rule them all: unification of voltage-gated potassium channel models via deep non-linear mixed effects modelling' (https://www.biorxiv.org/content/10.1101/2025.04.24.650426v1). At the time of writing it has been submitted to a journal and is un-

dergoing peer-review. The publication is followed by a brief supplemental information as well as some additional concluding remarks.

One model to rule them all: unification of voltage-gated potassium channel models via deep non-linear mixed effects modelling

Domas Linkevicius[1], Angus Chadwick[1], Melanie I. Stefan[2], David C. Sterratt[1,*]

1 Institute for Adaptive and Neural Computation, School of Informatics, University of Edinburgh, Edinburgh, United Kingdom
2 Faculty of Medicine, Medical School Berlin, Berlin, Germany

* David.C.Sterratt@ed.ac.uk

Abstract

Ion channels are essential for signal processing and propagation in neural cells. Voltage-gated ion channels permeable to potassium (K_v) form one of the most prominent channel families. Techniques used to model the voltage-dependent gating of K_v channels date back to Hodgkin and Huxley (1952). Different K_v types can display radically different kinetic properties, requiring different mathematical models. However, the construction of Hodgkin-Huxley-like (HH-like) models is generally complex and time consuming due to the number of parameters, their tuning and having to choose functional forms to model gating.

In addition to the between-K_v type heterogeneity, there can be significant within-K_v type kinetic heterogeneity between different cells with genetically identical channels. Since HH-like models do not account for such variability, extensions to it are necessary. We use scientific machine learning (SciML), the integration of machine learning methodologies with existing scientific models, and non-linear mixed effects (NLME) modelling to bypass the limitations of HH-like modelling. NLME is a modelling methodology that takes into account both within- and between-subject variability. These tools allowed us to complement the HH-like modelling and construct a unified SciML HH-like model that fits the recordings from 20 different K_v types. The unified SciML HH-like model produced closer fits to the data compared to a set of seven previous HH-like models and was able to represent the highly heterogeneous data from different cells. Our model may be the first step in producing a SciML foundation model for ion channels that would be capable of modelling the gating kinetics of any ion channel type.

Author summary

Ion channels are complex molecules embedded in the membranes of neurons – the cells responsible for signal propagation and processing in the brain. Ion channels can open and close in response to various types of stimuli, in particular the voltage difference across the cell membrane . Computational modelling, usage of mathematical techniques to represent a system and algorithmically solve for its dynamics, has been previously used to understand the dynamics of voltage-gated ion channels. However, computational modelling of voltage-gated ion channels requires costly and complex optimization routines to fit their structure and parameters. We utilize two tools new to the modelling

of voltage-gated ion channels – scientific machine learning and non-linear mixed effects modelling – to bypass some limitations associated with the existing methods.

By using scientific machine learning and non-linear mixed effects modelling we were able to create a unified model capable of modelling the gating dynamics of 20 different ion channels. This is in stark contrast to the existing modelling approaches, where each channel requires its own model. Moreover, our unified model performed better than seven existing ion channel gating models. Therefore, the tools we used and the model we created is a significant step forward in facilitating the modelling of ion channel gating. Future work could include even more ion channels types within the scope of our unified model.

Introduction

Ion channels are ubiquitous in cellular membranes and essential for various forms of signal transduction. There are multiple different mechanisms by which ion channels can be activated, such as different ligands, mechanical stimulation, and voltage, whereas some channels are passively open [1]. Voltage-gated ion channels are important because they are one of the main means of signal conduction in the nervous system and neural cells [2,3]. An important feature of voltage-gated ion channels is their permeability to one dominant ion type, with sodium, potassium and calcium being the most prominent [1]. Moreover, even among voltage-gated channels that are permeable to the same ion, there is significant functional and structural heterogeneity [4]. For example, there are 40 different voltage-gated potassium channels expressed in the mammalian brain, divided into 12 subfamilies called K_v1-12 [5]. Different voltage-gated ion channel types show different voltage dependencies, making their role in electrical signalling functionally distinct, yet overlapping [6]. One way to understand the functional complexity resulting from the abundance of different voltage-gated ion channel types is computational modelling.

Mathematical models of ion channels found in the central nervous system can be used in compartmental models of neurons to elucidate the importance and interactions between various channel types, as well as the interaction between electrical and biochemical signalling [7,8]. One of the most prominent and widespread paradigms of voltage-gated ion channel modelling was first published in the pioneering work of Hodgkin and Huxley [2]. Hodgkin and Huxley used various experimental manipulations to characterize the voltage dependencies of the sodium and potassium channel gating in the squid giant axon. The essence of their method is computational modelling of hypothesised gating variables via differential equations. Adapting the notation in [9], two voltage and temperature-dependent functions describe the dynamics of the ith gating variable m_i: the steady-state probability of the gate being open $m_{i,\infty}(V,T)$ and the time-constant of the kinetics $\tau_i(V,T)$, where V is the membrane potential and T is the temperature. These functions are used in an ordinary differential equation system which can be solved numerically to obtain the dynamics of the gating variables over time $m_i(t)$. The gating dynamics can then be used to obtain the current conducted over time $I(t)$. There are many possible functional forms used for $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ (see various models in `https://modeldb.science/` [7] for examples).

Even though the HH modelling paradigm was a breakthrough, its application can be complex: 1) finding and tuning the parameters of the often numerically unstable functional forms of $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ that can easily exhibit extreme behaviours; 2) setting up an appropriate m_i structure often requiring either expensive discrete optimization routines or manual hand-tuning; 3) assumption of independence between the gating variables that may make it difficult to fit the data. The first two difficulties are not unique to the HH paradigm, they also apply if Markov models, which are a

generalization of the kinetic schemes used by Hodgking and Huxley [2], are used. Markov models offer more flexibility, but can often result in parameter and kinetic scheme structure unidentifiablity [10]. The difficulties of the Hodgkin-Huxley model were highlighted with the publication of a data set described in Ranjan et al. [4].

The data recorded by Ranjan et al. [4] (see `https://channelpedia.epfl.ch/expdata`) contains automated voltage-clamp recordings of 40 different K_v channels, one at a time expressed in CHO cells. They used a set of voltage-clamping protocols to probe the activation, inactivation, deactivation and recovery of K_v channels, as well as their response to ramp and action potential-like stimuli. One of the key insights from the Ranjan et al. [4] data is the inherent kinetic heterogeneity of some K_v subtypes: even though the K_v channels expressed in the cells were genetically the same, the recordings showed significant variability for some channel types, e.g. for $K_v1.3$ or $K_v3.4$. The data showed variability in what using the HH paradigm would be modelled as $m_{i,\infty}$, as well as τ_i.

The challenge posed by the data in Ranjan et al. [4] to the HH and Markov paradigms is that the basic models have one set of parameters and are not built for handling heterogeneity, i.e., different functional forms for $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ and number of gates m_i with different functions may be required for different cells. Fitting a model to a single current recording is a challenging task; fitting many such models via the classical HH paradigm may be prohibitively expensive. Although there are purpose-built approaches to identifying the Markov gating structure of a given current recording [11], to the best of our knowledge, no approach currently used in computational neuroscience can efficiently handle the challenge of data heterogeneity.

To address the challenge of heterogeneous channel behaviour, we apply modelling tools used in pharmacokinetics and pharmacodynamics (PK/PD), where heterogeneous populations of patients and complex dynamics are commonplace. The nonlinear mixed effects (NLME) modelling [12] framework is common in PK/PD literature and is well-suited to model complex dynamics in heterogeneous populations. NLME is a hierarchical framework with the population level parameters (fixed effects) at the highest level and individualized parameters (random effects) at the lowest level (see Figure 2). This structure allows to both individualize predictions in the presence of significant individual differences (for example, people in clinical trials), as well as extract average, population level models useful for future predictions. Moreover, the NLME approach allows for arbitrary nonlinear functions to be used, so long as the gradient vectors with respect to both fixed and random effects can be calculated. NLME modelling has been applied widely in PK/PD (e.g. see [13]), and infectious disease modelling [14], but it is also gaining popularity in other domains, for example, individualized predictions of mood [15], repeated measurements of individualized gaze estimation [16] and many other applications where the hierarchical structure is more faithful to reality than a single level structure.

Given nonlinear functions powerful enough to model any $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ present in the data and enough gates m_i, the NLME approach would serve as a suitable tool to handle the heterogeneity observed in the Ranjan et al. [4] data. We use NLME and neural networks, which are universal function approximators [17], to represent $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ and fit a single HH-like model for all K_v channels for which Ranjan et al. [4] provide data. We first fit a unified K_v channel model by pooling data from different channel types into a single data set, relying on the power of neural networks to provide functional flexibility and the random effects to capture the heterogeneity present in the data. We then utilize the channel type and recording temperature information of individual recordings to predict a portion of variability inherent to the channel type and temperature that was initially captured by the random effects. This step also allows the derivation of average models for different K_v types.

We then compare our unified model performance against multiple existing single channel model baselines. We show that the predictions of our unified model result in lower RMSE values than ones from models used in existing publications, even if random effects are added to their parameters. This comparison proves that our unified model that uses neural networks to represent $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ learns representations that can model the K_v dynamics in response to protocols presented in the data more accurately than the bespoke models . Moreover, the benefits of having a unified model are shown via the superior performance of the unified model against multiple separate models using the same neural network-based approach. We hypothesize that the unified approach outperforms the multiple single channel modelling approach due to the sharing of information between similar K_v types, especially when data is more limited, and finding more universal $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ forms. Finally, we analyse the $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ functions, and the temperature dependence of different K_v types.

Our results are practically useful in many ways, for example, by providing a significantly simpler and more efficient way to represent different K_v models, by providing a workflow of voltage-gated ion channel model development that is significantly faster and more powerful than the previously used methods and a set of heterogeneous K_v channel fits that are usable in traditional computational neuroscience simulators, such as NEURON [18]. Finally, our results point to a set of new higher level possibilities, such as a unified channel model representing both potassium and sodium channels, as well as being able to more easily integrate information of how different cellular factors affect channel dynamics.

Methods

Raw data

We downloaded the data for all cells that were classified as active by Ranjan et al. [4] for further processing locally. For a full description of the data and the recording conditions see the original publication. The raw current recording data used in this study has been downloaded in the neurodata without borders (`.nwb`) format from `https://channelpedia.epfl.ch/expdata` for the following 20 channel types: $K_v1.1$, $K_v1.2$, $K_v1.3$, $K_v1.4$, $K_v1.5$, $K_v1.6$, $K_v1.8$, $K_v2.1$, $K_v2.2$, $K_v3.1$, $K_v3.1$, $K_v3.3$, $K_v3.4$, $K_v4.1$, $K_v4.2$, $K_v4.3$, $K_v10.1$, $K_v10.2$, $K_v12.1$, and $K_v12.3$. These channel types were chosen based on a preliminary inspection of the number of cell recordings that showed a good signal to noise ratio in a sufficient number of recordings over different temperatures (15°C, 25°C and 35°C).

Voltage-clamp protocols used in Ranjan et al.

The current recordings in Ranjan et al. [4] were obtained by applying (often multiple times) six voltage-clamp protocols. Each protocol began with a 100ms baseline period, where in the first 40ms each cell was held at −80mV, then the holding voltage was set to −90mV for 10ms and from 50ms to 100ms the voltage was set back to −80mV. After this initial baseline period, each cell was exposed to one of six voltage clamp protocols – activation, deactivation, inactivation, recovery, ramp and action potential-like stimulation – meant to probe different K_v channel properties. Each protocol consisted of a specific number of sweeps where a single feature of the protocol was varied. After each sweep, the cell was held at −80mV for the final 100ms. Generally, after the initial activation protocol, the protocols in Ranjan et al. [4] were applied in a consistent order (potentially multiple times): activation, ramp, deactivation, AP, inactivation. After the

repetition of these five protocols, the recovery protocol was applied, potentially multiple times. Due to the limitations of `Pumas.jl` (its usage is described below), which disallows continuously applied time-varying negative stimuli, we only used the first four protocols – activation, deactivation, inactivation and recovery, described briefly below; for full description, see Ranjan et al. [4].

Activation

The activation protocol consists of 18 sweeps at different holding voltages meant to measure the steady state activation levels of K_v channels. Specifically, after the baseline period, cells were held between −90mV and +80mV (10mV increments) for 500ms.

Deactivation

The deactivation protocol consists of 12 sweeps meant to probe the deactivation kinetics of K_v channels. Each sweep includes two different voltage levels: during the first 300ms the cells were clamped at +70mV, whereas for the subsequent 200ms the cells were held at voltages between −80mV to +30mV (10mV increments).

Inactivation

The inactivation protocol consists of 12 sweeps meant to probe the inactivation kinetics of K_v channels. Each sweep includes two different voltage levels. During the first 1500ms the cells were held at voltages between −40mV to +70mV (10mV increments). Afterwards, the holding voltage is switched to +30mV for 100ms.

Recovery

The recovery protocol consists of 16 sweeps intended to investigate the kinetics of K_v recovery after activation. For the first 1500ms the cells were held at +50mV to induce channel activation and, potentially, inactivation. Then, each cell was held at −80mV for a time period ranging between 50ms to 2300ms (150ms increments) during which the channels recovered. Finally, the cells were held at +50mV for 200ms to measure the levels of channel recovery after the recovery period.

Data processing

In order to make the data in Ranjan et al. [4] suitable for model training we undertook a series of data processing steps (Figure 1). First, we filtered out inconsistent data. We then applied a time series smoothing algorithm to reduce the noise levels, followed by setting the current baseline to 0, normalization of the current, rescaling, exclusion of systematic data artifacts and downsampling. All data processing, modelling and analyses in this paper were performed using the `Julia v1.10.4` programming language [19].

Filtering out inconsistent data

We included only the second repetition of the activation protocol and the first repetition of each other protocol for further processing instead of averaging the different repetitions. We chose this approach due to frequent inconsistencies of current amplitudes or time constants between subsequent repetitions. We assumed that the earlier repetitions are of higher quality due to suffering from rundown less.

First, we visually inspected the second repetition of the activation protocol for all of the cells and if any systematic irregularities were detected, we manually excluded the

cells from further usage. This left a total of 2969 cells for further processing. For a more detailed summary of the number of sweeps left after the data processing see Table 1.

Since the activation sweeps were recorded first, we treated them as the standard reference point against which subsequent data is to be compared and scaled to. We compared the sweeps from other protocols to the relevant parts of the activation protocol using the mean absolute percentage error (MAPE), where m is the length of the time series being compared, x is the reference time series and y is the time series being evaluated:

$$\mathrm{MAPE}(x, y) = \frac{1}{m} \sum_{i=1}^{m} \left| \frac{x_i - y_i}{x_i} \right| \tag{1}$$

Using cell and sweep-specific references we then excluded any subsequent sweeps that were not consistent with the relevant activation sweep according to the following criteria:

- **Deactivation** – after the initial baseline period of 100ms, for the next 300ms in all sweeps in the deactivation protocol the cells are held at +70mV. If after setting the baseline to 0 (by subtracting the mean of the current during the first 40ms of the baseline period), and scaling of the peak current to match the activation protocol peak at +70mV, MAPE > 0.05, the sweep is not used.

- **Inactivation** – after the initial baseline period of 100ms, cells are held at voltages ranging from −40mV to +70mV, which is a subset of the voltages used in the activation sweeps. Therefore, we took the first 500ms post-baseline from the inactivation sweeps and the activation sweeps held at the same voltage. We then set both of their baselines to 0 (by subtracting the mean of the current during the first 40ms of the baseline period) and scaled both inactivation and activation sweeps by the ratio of the activation peak at +80mV divided by the peak of the respective sweep (inactivation or activation). If MAPE > 0.05 for the inactivation protocol compared against the matching activation protocol, the inactivation sweep was not used. We chose to scale both sweeps to the activation peak at +80mV, which introduces a systematic bias, instead of the peak at the matching activation voltage to avoid introducing additional noise which often resulted in inconsistencies in inactivation peaks, where clamping to lower voltages resulting in higher current peaks compared to higher voltages.

- **Recovery** – after the initial baseline period of 100ms, cells are held at +50mV for the first 1500ms. Therefore, we took the first 500ms of that 1500ms and compared against the 500ms of activation protocol at +50mV. We set the baseline of both sweeps to 0, scaled both sweeps by the ratio of activation peak at +80mV divided by the peak of the given sweep and calculated the MAPE values. Moreover, there were multiple cases where the recovery sweeps were internally inconsistent; some traces showed significantly different current values at the end the 1500ms. Therefore, we calculated the median current value at 1500ms post-baseline over all the sweeps of a given cell and we used recovery sweep only if MAPE < 0.05 and the current value at 1500ms post-baseline was within one standard deviation of the median.

These consistency checks were necessary to ensure that we kept only the sweeps that were internally consistent with the activation sweeps, for example there were no significant differences in time constants or large recording artifacts, etc. Usage of inconsistent data would have significantly hampered model training by requiring additional model mechanisms capable of accounting for the inconsistencies.

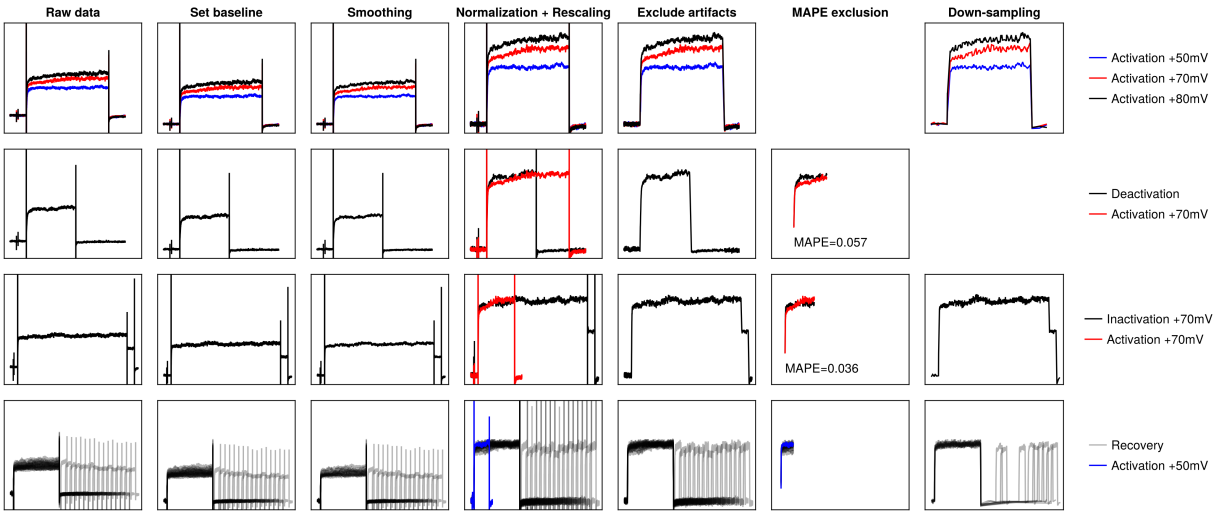


Fig 1. Visual representation of data processing pipeline. Each row represents the processing undertaken by some sample sweeps for each protocol: top row – activation, second row – deactivation, third row – inactivation, fourth row – recovery. The legends for each row are given in the last column. The plot for MAPE exclusion for the activation protocol is not shown because this step is not applied to the activation protocol. The down-sampling was not applied to the deactivation protocol because MAPE > 0.05. The normalization and rescaling column contains the relevant activation trace to which the other protocols were rescaled to.

Smoothing

After the filtering of the sweeps inconsistent with the second repetition of the activation protocol, we smoothed the data using the Julia `KissSmoothing.jl v1.0.8` package (see `https://github.com/francescoalemanno/KissSmoothing.jl`). It implements a smoothing function called `denoise` which transforms the signal using the discrete cosine transform and convolves it with an Gaussian function (similar to the method outlined in [20]). We use `denoise` with default parameters, except for setting the smoothing intensity to `factor=1.0`.

Setting the baseline

Since the raw current recordings were often offset from 0 at −80mV, when there should be no significant potassium current (due to the voltage being close to the reversal potential and channels generally being closed), we calculated the mean of the first 40ms of each sweep and subtracted it from the full sweep to set the mean during the baseline period to 0.

Normalization

Since we were only interested in the channel gating dynamics, we normalized each sweep by the current peak I_{max} of activation sweep when the cell was held at +80mV. This allows us to model only the gating dynamics, reducing the number of parameters that need to be fit.

Rescaling

For each sweep of each protocol other than activation, it was rescaled by ratio of the matching activation sweep peak current divided by the given sweep peak current before

Table 1. Number of sweeps of the data remaining after filtering of inconsistent data and irregularities.

Channel type	Total # of sweeps	15°C	25°C	35°C	Activation	Deactivation	Inactivation	Recovery
All types	87640	25788	45535	16317	52020	17867	10785	6968
$K_v1.1$	7024	2168	3417	1439	4716	1378	618	312
$K_v1.2$	6010	1848	3165	997	3600	1462	503	445
$K_v1.3$	8323	2042	4628	1653	6120	1403	493	307
$K_v1.4$	5040	1169	2806	1065	3942	936	145	17
$K_v1.5$	8788	1696	6234	858	4356	2030	1090	1312
$K_v1.6$	3837	1312	1703	822	2394	817	376	250
$K_v1.8$	1116	514	380	222	972	46	91	7
$K_v2.1$	4552	1578	2471	503	1800	1180	699	873
$K_v2.2$	4071	736	2464	871	2034	1134	475	428
$K_v3.1$	4943	1882	1929	1132	2034	1286	949	674
$K_v3.2$	6663	1678	4045	940	2700	1650	1259	1054
$K_v3.3$	8087	1626	4595	1866	4104	1993	1180	810
$K_v3.4$	3421	1275	1308	838	2430	527	364	100
$K_v4.1$	2028	922	870	236	1512	319	141	56
$K_v4.2$	1318	697	381	240	1134	53	117	14
$K_v4.3$	2557	1009	1322	226	1908	477	128	44
$K_v10.1$	3165	991	1547	627	1926	596	635	8
$K_v10.2$	1238	386	242	610	1026	2	204	6
$K_v12.1$	4161	1627	1638	896	2286	563	1061	251
$K_v12.3$	1298	632	390	276	1026	15	257	0

normalization by I_{max}. Concretely, each sweep of the deactivation protocol was rescaled by the activation sweep peak when the cell is held at +70mV. Each trace of the inactivation protocol was rescaled by the matching activation sweep. Finally, each sweep of the recovery protocol was rescaled by the activation sweep peak when the cell is held at +50mV.

Excluding systematic recording artifacts

A significant number of sweeps showed fast negative deflection artifacts whenever voltage-clamping level was changed. This behavior does not arise from channel gating dynamics, therefore, after taking the previously described data processing steps, we excluded any points that are more negative than -0.01. This is to avoid the model being penalized for not fitting the data it by construction can not reproduce.

Down-sampling of sweeps

In order to speed up model training we down-sampled all the sweeps by using the M4 down-sampler. The M4 down-sampler was chosen because of its simplicity, efficiency and preservation of extrema [21]. More specifically, for a given time-series x and number of bins b, the M4 down-sampling algorithm starts by dividing x into b non-overlapping bins and then takes the first and the last values in that bin, as well as the maximal and the minimal remaining values. Therefore, the M4 down-sampler reduces the length of a time-series from the initial length k to $4b$.

Each initial 100ms baseline recording was reduced from 1000 to 9 points: 3 points from the first 40ms, 3 points from 40ms to 50ms and 3 points from 50ms to 98ms, all spaced equidistantly within each of these periods. The final 100ms was also down-sampled to 3 equidistant points. For the activation protocol, the 4000 points

starting from 100ms to 600ms were down-sampled to 120 points. For the deactivation protocol, the 3000 points between 100ms to 400ms were down-sampled to 72 points, while the points between 400ms to 600ms were down-sampled to 48 points. For the inactivation protocol the 15000 points between 100ms and 1500ms were down-sampled to 360 points, whereas the 1000 points between 1600ms to 1700ms were down-sampled to 24 points. Finally, for the recovery protocol, the 15000 points between 100ms and 1600ms were down-sampled to 360 points, the period between the initial +50mV and the test +50mV that was of variable length was down-sampled to 3 points spaced equidistantly, and the second +50mV period of 150ms was down-sampled to 48 points. Down-sampling was the final step of our data processing pipeline. The data remaining at this point was split into training, validation and test data sets (described below). The full raw and the processed data are available to be downloaded at [22]. Next we move on to describe the general nonlinear mixed effects modelling approach.

Nonlinear mixed effects (NMLE) modelling

We adapt the definitions of NLME provided in [23, 24]. The NLME modelling framework comprises a two level hierarchical structure (Fig 2) with fixed effects Θ at the upper level which do not vary between recorded entities. Fixed effects can be broadly grouped into

- model parameters θ
- random effect prior distribution parameters Ω
- observation model noise parameters σ

The lower level consists of random effects η_n which account for the inter-individual variability of the observations by individualizing the model parameters θ for the nth individual. In this study the inter-individual variability is between the recorded currents y_n for the nth cell. Furthermore, a set of covariates Z_n (which are known at the outset) are associated with the nth cell, namely, the temperature of the recording (but see below), as well as the channel type.

These three sets of values (Θ, η_n and Z_n) are collated via the parameter model g (note the similar but distinct notation $\bar{g}$ for the maximal channel conductance) into the dynamical parameter vector p_n of the nth cell

$$p_n = g(\Theta, Z_n, \eta_n) \tag{2}$$

The dynamical parameters p_n are then fed into the structural model (e.g. an ordinary differential equation (ODE) system)

$$u_n' = f(u_n, p_n, t) \tag{3}$$

where u_n are the dynamical variables being solved for, e.g. the gating variables in the Hodgkin-Huxley formalism. The final step is to link the numerical solution of the ODE system to the experimentally observed quantities y_{nj}, where j denotes the number of different observed quantities for the nth individual. In this study the only observable quantity is the recorded normalized current defined in Equation 10. After numerically solving Equation 3, the appropriate variables are derived and passed through a Gaussian observational model to account for observational noise, giving y_{nj} (in this study there is only one observed quantity, therefore j is omitted)

$$p(y_n(t) \mid \Theta, \eta_n) = \mathcal{N}\left(\frac{I_n(t)}{I_{n_{max}}}, \sigma\right) \tag{4}$$

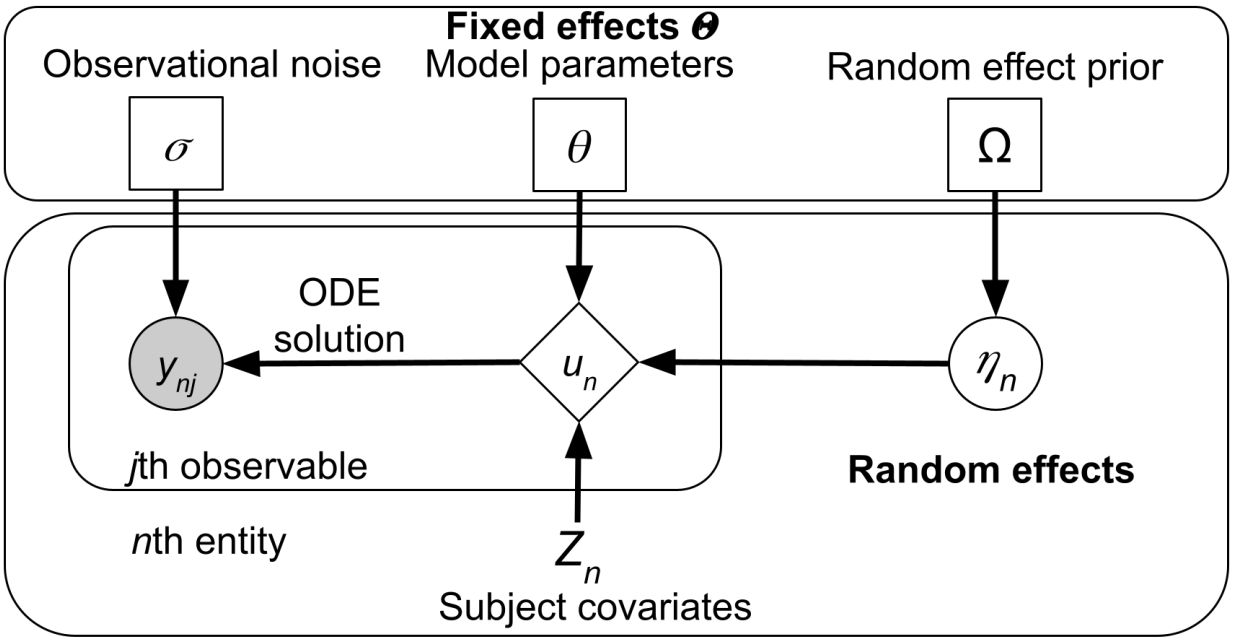


Fig 2. Visual representation of an NLME model, rectangle nodes in the top box denote parameters (fixed effects), circles denote random quantities which are either latent (unfilled) or observed (filled), diamonds are deterministic given the inputs, and nodes without a border are constant.

where $\frac{I_n(t)}{I_{n_{max}}}$ is defined in Equation 10. In all following equations the dependence of y_n on time will be omitted to reduce visual clutter.

There are many ways to fit NLME models, both frequentist and Bayesian [25]. In this study we used both the maximum a posteriori (MAP) conditional log-likelihood objective which can be stated as

$$\Theta^*, \eta^* = \arg\max_{\Theta,\eta} \left(p(\Theta) \cdot \prod_{i=1}^{N} p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta) \right) \tag{5}$$

as well as the maximum a posteriori (MAP) first order conditional estimate (FOCE) of the marginal likelihood [9]

$$\begin{aligned} \Theta^* &= \arg\max_{\Theta} \left(p(\Theta) \cdot \prod_{n=1}^{N} p(y_n \mid \Theta) \right) \\ &= \arg\max_{\Theta} \left(p(\Theta) \cdot \prod_{n=1}^{N} \int p(y_n \mid \Theta, \eta_n) \cdot p(\eta_n \mid \Theta)\, d\eta_n \right) \end{aligned} \tag{6}$$

with the random effects for individual cells set to their empirical Bayes estimate (EBE_n values defined as

$$\mathrm{EBE}_n = \eta_n^* = \arg\max_{\eta_n} \left(p(y_n \mid \Theta = \Theta^*, \eta_n) \cdot p(\eta_n \mid \Theta = \Theta^*) \right) \tag{7}$$

In both definitions Θ^* is the mode of the fixed effects and $p(\Theta)$ is the fixed effect prior distribution. We specify how we used both of these objectives below.

K_v channel models

In this study we use four types of models which we evaluate and fit to the Ranjan et al. [4] data, all of which follow the general HH equations for the gating variables m_i which form the function f in Equation 3:

$$\frac{dm_i}{dt} = \frac{m_{i,\infty}(V,T) - m_i}{\tau_i(V,T)} \tag{8}$$

These equations can be solved numerically to obtain the current $I(t)$ conducted by the channels over time. $I(t)$ is the product of the gating variables $m_i(t)$, the maximum channel conductance $\bar{g}$, the ionic reversal potential E and the gating variable power n_i

$$I(t) = \bar{g}(V(t) - E)\prod_i m_i^{n_i}(t) \tag{9}$$

Since we are only interested in the channel gating dynamics captured by m_i, instead we derive the normalized current $\frac{I(t)}{I_{max}}$

$$\frac{I(t)}{I_{max}} = \frac{\bar{g}(V(t) - E)\prod_i m_i^{n_i}(t)}{\bar{g}(V_{max} - E)\prod_i m_{i,max}^{n_i}} = \frac{1}{\tilde{V}}(V(t) - E)\prod_i m_i^{n_i}(t) \tag{10}$$

where $\tilde{V} = V_{max} - E$, $V_{max} = 80\text{mV}$ and $\prod_i m_{i,max}^{n_i} = 1$ by definition. We next describe the four types of HH models that we used, which differ in their scope, the functional forms used for $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$, the inclusion of the gating power n_i and the inclusion of random effects.

The first model type, which we call the classical HH models, is a set of K_v models from `https://modeldb.science/`. Since a systematic evaluation of existing K_v channel models is outside the scope of this work, we selected a number of models based on preliminary tests of their quality of fitting the Ranjan et al. [4] data for the K_v channel being modelled. The K_v channels for which we found reasonably good baseline models are $K_v1.1$ [4], $K_v1.2$ [26], $K_v1.5$ [27], $K_v3.1$ [28], $K_v3.3$, $K_v3.4$ and $K_v4.3$ which are all from [29] (see respective publications for specific model structure). The classical HH models serve as the exemplars of the class of models that are currently among the most prominent in computational neuroscience.

Since none of the classical HH models included random effects, in addition to the original models, we also analysed the same models with random effects added on their parameters. We used a standard multivariate Gaussian random effect prior and only fitted the EBE_n for these models using the FOCE objective (see Eq. 6). Addition of random effects allowed for a fairer comparison with the next two types of models and the baseline models, helping to isolate the impact of using alternative functional forms for $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$. We call this group of models the classical HH models with random effects.

Our third model type is the individual SciML HH models. For this model type, we use neural network-based $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ functions. The models follow the classical HH formalism from Equation 8, but in this case $m_{i,\infty}^{(n)}(V,T_n) = \text{NN}_{m_{i,\infty}}(V,\eta_{i,m}^{(n)},\theta_{i,\infty},T_n)$ and $\tau_i^{(n)}(V,T_n) = \text{NN}_{\tau_i}(V,\eta_{i,\tau}^{(n)},\theta_{i,\tau},T_n)$ where V is voltage, $\eta_{i,f}$ are the random effects for the appropriate function f, $\theta_{i,\infty}$ and $\theta_{i,\tau}$ are neural network weights and biases of the ith respective neural network, $\text{NN}_{m_{i,\infty}}$ and NN_{τ_i} are specific types of neural network architecture and T_n is the normalized temperature for the nth cell. Specifically, $\text{NN}_{m_{i,\infty}}$ is capped off with a sigmoid nonlinearity and outputs in the range of $(0,1)$, whereas NN_{τ_i} is capped off with a softplus nonlinearity and outputs in the range of $(0,\infty)$. For all models we used $i = 2$ gates, therefore there were four vectors of neural network weights per model for a K_v

type, where each neural network had two hidden layers of 5 units each with L1 regularization of the weights with $\lambda = 10^{-4}$. Moreover, we used 8 random effects, two for each function.

The fourth type of model we use is the unified SciML HH model. This model type is used to model all of the K_v types via the same set of fixed effects Θ, largely following the neural network-based $m_{i,\infty}^{(n)}(V,T)$ and $\tau_i^{(n)}(V,T)$ functions used in the individual SciML HH models (including the same hyper-parameters). However, in this case the random effects $\eta_{i,f}$ contain two components, the truly random effects η^* and the augmentation random effects $\eta^\dagger$ which are added element-wise, i.e. $\eta_{i,f}^{(n)} = \eta_n^* + \eta_n^\dagger$. The augmentation random effects for the nth cell are based on the channel type c_n, as well as the normalized temperature $\bar{T}_n$. The channel type c_n is encoded as a 19 dimensional one-hot vector, where all zeros would correspond to $K_v1.1$. A 19 dimensional, rather than 20 dimensional one-hot vector was chosen to slightly reduce the number of parameters in the neural network, as the expressivity of the one-hot encodings is identical. It was assumed that the normalized temperature $\bar{T}_n$ is a noisy estimate of the true temperature by adding a random effect $\eta_{temp} \sim \mathcal{N}(0,1)$. The observation that the temperature is a noisy estimate is based on the recordings of the electrode tip temperature available in the original data set. We then pass the channel type vector and the normalized temperature to an augmentation neural network $\eta_n^\dagger = \mathrm{NN}_{aug}(c_n, \bar{T}_n, \theta_{aug})$ where θ_{aug} is the set of neural network weights. This model structure allows the inherent heterogeneity present in the data to be accounted for, while capturing the portion of its variance attributable to different K_v types and temperatures via NN_{aug}.

Data splitting into training, validation and test data sets

For the individual SciML HH model training we split the processed data using a 60%/20%/20% fraction for training, validation and test. We took the appropriate fractions from each temperature for each channel type, rather than pooling the temperatures together. The validation and test data derived in this fashion were used for the validation and testing of all model types. However, the training data set for the unified SciML HH model, due to its size, required a different approach.

For the unified SciML HH model, we took seven cells for each of the three measured temperatures for each channel type, except for $K_v1.8$, $K_v4.2$, $K_v12.3$. We took seven cells because that was the maximal number of cells that we could take for each channel type such that the resulting data set would be balanced for each temperature and each channel type. This resulted in 357 cells in the training data. We left $K_v1.8$, $K_v4.2$, $K_v12.3$ out of the training data in order to establish whether the unified SciML model could generalize to channel types which were not in the training data set.

Model fitting

We use the `Pumas.jl` [23] and `DeepPumas.jl` [30] Julia packages to solve Equations 5 and 6. `Pumas.jl` contains efficient and powerful algorithms for NLME modelling, whereas `DeepPumas.jl` contains the code infrastructure necessary to incorporate neural networks into the NLME modelling. More specifically, it implements algorithms to solve equations 6, 5 and 7 and uses forward-model automatic differentiation to allow gradient-based optimization techniques. We used the BFGS optimization algorithm from `Optim.jl` with the gradient calculations handled by `Pumas.jl` and `DeepPumas.jl`.

Both the classical HH models with random effects and the individual SciML HH type models were fit to the individual K_v channel type data, whereas the unified SciML

HH model was fit to a unified data set described earlier. For the classical HH models with random effects, we fixed the θ and the Ω values and only fit the σ and the η_n values to data.

All fitting was done on the JuliaHub (`https://juliahub.com/`) cloud computing platform using nodes with 8 vCPUs and 64GB of memory. Fits took between three and twenty-four hours, depending on the K_v type and the amount of data which was used. All the code that was used to define the models, run the simulations and perform the analysis is accessible at
`https://github.com/dom-linkevicius/SciMLHHModels.jl.git`.

The first stage of fitting was shared between the individual and the unified SciML HH models. In the first step of the fitting we used the conditional likelihood objective (Eq. 5) to produce an initial fit of all of the fixed effects, except for the parameters of the random effect prior Ω which was initialized to be standard normal and was held constant throughout this step of the optimization.

We ran the optimization for 300 epochs, saving the results every 15 epochs, evaluating the model performance on the validation data and selecting the set of parameters that had the best performance on the validation data. Conditional likelihood is much more numerically efficient due to Θ and η_n being optimized jointly whereas, for example, marginal likelihood generally requires a two level optimization scheme. However, conditional likelihood requires appropriate handling of Ω to avoid overly broad random effect distributions which barely penalize extreme η_n values and effectively result in different individual models due to the learning being offloaded mostly to the random effects.

In the second stage, which was only applied to the unified SciML HH model, we held all of the fixed effect values constant, except for the random effect prior parameters Ω. Then we used the FOCE objective (Equation 6) to fit the Ω. We ran the optimization for 10 iterations. The objective changed very minimally during the 10 iterations (so the quality of the data fit did not change), but the values in Ω were adjusted appropriately. Since the data fitting quality does not change in this step, we opted to omit the individual SciML HH models from this step because our main focus is on the unified SciML HH model (see Results and Discussion).

This two stage optimization procedure utilizes the numerical efficiency of the conditional likelihood by first finding the neural network weights capable of representing the functions necessary to model the current conducted via the K_v channels. It also safeguards against the over-fitting of the random effects by not allowing the penalty coming from the random effect prior to become negligible via Ω assuming arbitrarily large values. The second step in the optimization pipeline only optimizes Ω using an approximation to the marginal likelihood. This step is meant to calibrate the spread of the random effects in order to properly account for the inherent heterogeneity present in the data that is not due to the differences in temperature or channel types. Under ideal circumstances of large enough compute capabilities, joint fitting of all the fixed effects using only marginal likelihood would be preferable, but was not feasible in the present study.

Results

As described in detail in the Methods, we fitted two novel types of HH models to voltage clamp data from 20 K_v channels collected by Ranjan et al. [4] and compared them against a set of classical HH models with and without random effects. The first set of seven classical HH models for different K_v types was taken from the literature (see Methods). The second set of models was produced by taking the models from the first set and adding random effects on their parameters. The third set of 20 models was

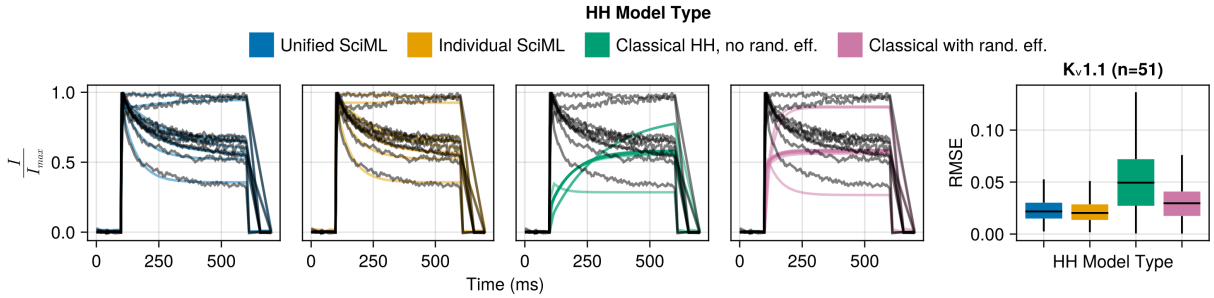


Fig 3. Comparison of the four different HH model types for the $K_v1.1$ channel. Left-most panel shows box plots of the test data RMSE values (at the sweep level, e.g. activation at +80mV). The next four plots show 10 activation sweeps at +80mV (from different cells) from the test data set (black) and the matching model predictions (coloured lines), providing a visual link between the RMSE values in the box plot and traces. The classical HH model without random effects (green line) displays variance because even though the model parameters do not have random effects on them, the temperature used in it (and the Q_{10} value which scales them) does.

produced by using neural networks to represent the voltage-dependent gating functions $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ and we call it the individual SciML HH models. The final model again used neural networks to represent $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$, but it was fitted to a joint data set of different K_v channels, we call this model the unified SciML HH model.

We present the results of both the individual and the unified SciML HH models that we fit, along with the performance of the classical HH models and the classical HH models with random effects. We only show the results of the best performing models found during the optimization. For more detailed results of model selection see S1 Appendix.

Fig 3 shows the comparison of the different types of HH models for one example channel, $K_v1.1$, chosen because Ranjan et al. [4] compare a model for this channel to their data. The four leftmost plots in Fig 3 show the normalised voltage clamp currents (see Methods) for 10 sweeps of the activation protocol at +80mV (black lines) along with the respective model predictions (blue – unified SciML HH, yellow – individual SciML HH, green – classical HH without random effects, pink – classical HH with random effects. Visually, there is a clear difference between the fits for the SciML HH models and the HH models, with the classical HH models performing poorly, the SciML HH models fitting the data reasonably well. The box plots (rightmost plot in Fig 3) of the root mean square error (RMSE) show that the SciML HH models significantly outperform the classical HH models with or without random effects (for statistical test results see Table 2). The individual SciML HH model significantly outperforms the unified SciML HH model, but the size of the difference is small (Table 2).

Fig 4 shows the (RMSE) box plots for different model predictions compared to normalised voltage clamp current at the protocol sweep level for individual channel types. The four types of models – unified SciML HH (blue), individual SciML HH (yellow), classical HH without random effects (green) and classical HH with random effects (pink) display a consistent pattern of performance across different K_v channel types.

Firstly, the comparisons between the classical HH models with and without random effects confirms the utility of the NLME approach in modelling the K_v channel data of Ranjan et al. [4]. In all seven cases ($K_v1.1$, $K_v1.2$, $K_v1.5$, $K_v3.1$, $K_v3.3$, $K_v3.4$ and $K_v4.3$) of the implemented classical HH models, the version of the model with random

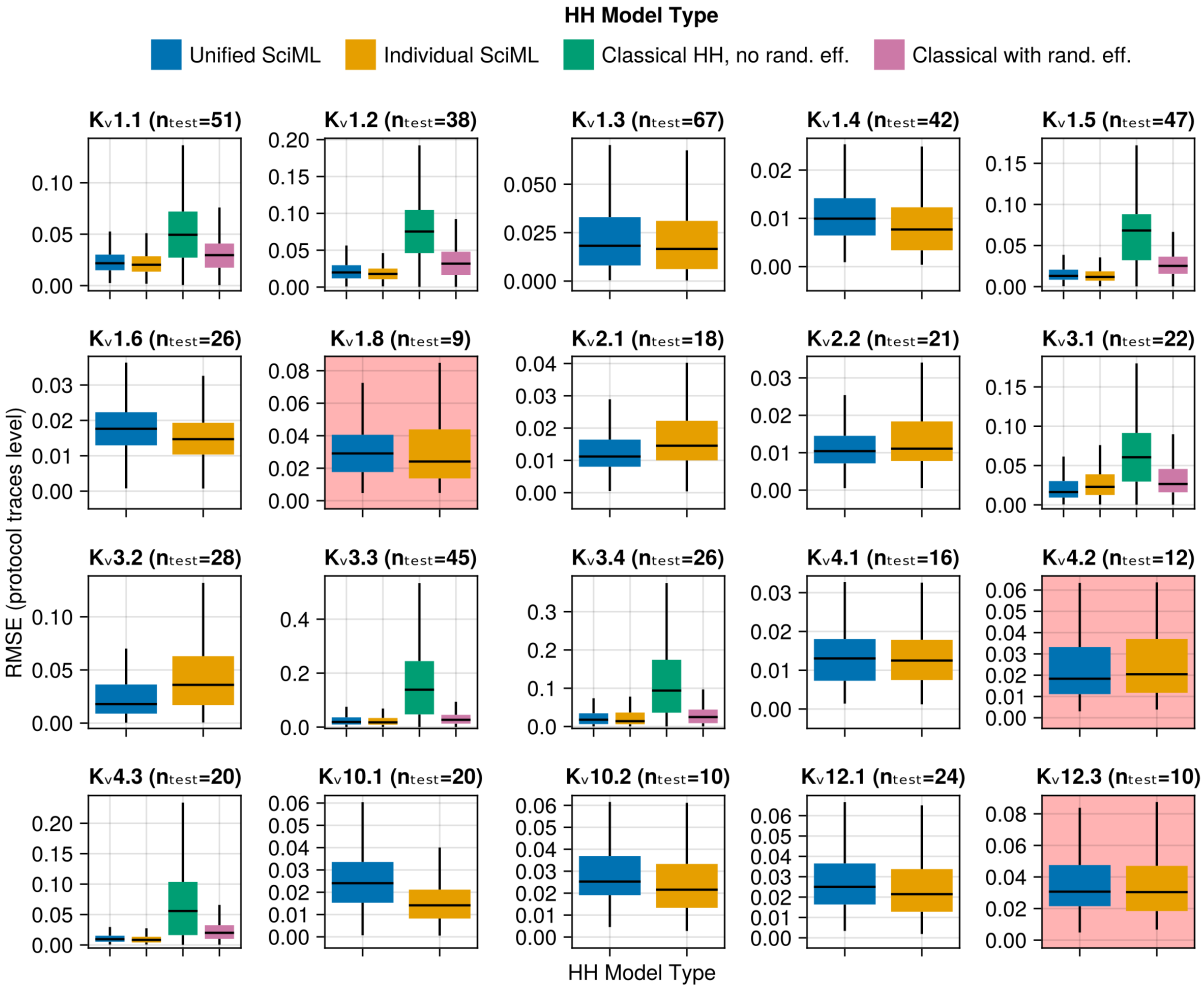


Fig 4. RMSE box plots obtained by comparing the data against model predictions at the sweep level, e.g. Activation protocol at −10mV. Different colours represent different HH model types: blue – unified SciML HH model, yellow – individual SciML HH models, green – classical HH models re-implemented from existing publications without random effects, pink – classical HH models re-implemented from existing publications but with random effects on their parameters. The number of different cells in the test data set is denoted by n_{test} and shown above each subplot. The channel types ($K_v1.8$, $K_v4.2$ and $K_v12.3$) not used in the training of the unified SciML HH model have a light red background to distinguish them from the rest. Note that outliers are not shown and the y axes are not identical between different subplots to enhance legibility.

Table 2. Comparison of performance between different models on the test data set.

Channel type	$\mathcal{M}_1$	$\mathcal{M}_2$	median($\mathcal{M}_1$ - $\mathcal{M}_2$)	W **statistic** (n = **sample size**)	**Cohen's d**‡
$K_v1.1$	uSciML	iSciML	0.0005	601 731* ($n = 1390$)	0.10
$K_v1.1$	uSciML	cHH	−0.0242	56 854* ($n = 1390$)	1.02
$K_v1.1$	uSciML	cHH with randeff.	−0.0054	201 466* ($n = 1390$)	0.47
$K_v1.1$	iSciML	cHH	−0.0253	39 686* ($n = 1390$)	1.06
$K_v1.1$	iSciML	cHH with randeff.	−0.0055	131 746* ($n = 1390$)	0.53
$K_v1.1$	cHH	cHH with randeff.	0.0145	890 562* ($n = 1390$)	0.63
$K_v1.2$	uSciML	iSciML	0.0010	527 946* ($n = 1269$)	0.16
$K_v1.2$	uSciML	cHH	−0.0463	25 109* ($n = 1269$)	1.01
$K_v1.2$	uSciML	cHH with randeff.	−0.0075	108 906* ($n = 1269$)	0.58
$K_v1.2$	iSciML	cHH	−0.0494	24 529* ($n = 1269$)	1.06
$K_v1.2$	iSciML	cHH with randeff.	−0.0107	91 870* ($n = 1269$)	0.72
$K_v1.2$	cHH	cHH with randeff.	0.0317	773 999* ($n = 1269$)	0.79
$K_v1.3$	uSciML	iSciML	0.0011	877 521* ($n = 1667$)	0.06
$K_v1.4$	uSciML	iSciML	0.0018	466 856* ($n = 1049$)	0.34
$K_v1.5$	uSciML	iSciML	0.0009	1 085 000* ($n = 1770$)	0.09
$K_v1.5$	uSciML	cHH	−0.0459	76 775* ($n = 1770$)	0.97
$K_v1.5$	uSciML	cHH with randeff.	−0.0095	178 057* ($n = 1770$)	0.59
$K_v1.5$	iSciML	cHH	−0.0475	52 527* ($n = 1770$)	0.98
$K_v1.5$	iSciML	cHH with randeff.	−0.0105	141 359* ($n = 1770$)	0.63
$K_v1.5$	cHH	cHH with randeff.	0.0300	1 438 910* ($n = 1770$)	0.70
$K_v1.6$	uSciML	iSciML	0.0017	257 406* ($n = 801$)	0.36
$K_v1.8$	uSciML	iSciML	0.0014	23 208 ($n = 282$)	0.06
$K_v2.1$	uSciML	iSciML	−0.0021	150 313* ($n = 1014$)	0.39
$K_v2.2$	uSciML	iSciML	−0.0007	136 530* ($n = 868$)	0.27
$K_v3.1$	uSciML	iSciML	−0.0006	206 663* ($n = 1077$)	0.24
$K_v3.1$	uSciML	cHH	−0.0356	21 131* ($n = 1077$)	1.15
$K_v3.1$	uSciML	cHH with randeff.	−0.0067	60 015* ($n = 1077$)	0.61
$K_v3.1$	iSciML	cHH	−0.0284	32 108* ($n = 1077$)	1.03
$K_v3.1$	iSciML	cHH with randeff.	−0.0035	159 664* ($n = 1077$)	0.43
$K_v3.1$	cHH	cHH with randeff.	0.0143	514 572* ($n = 1077$)	0.70
$K_v3.2$	uSciML	iSciML	−0.0122	71 679.0 ($n = 1517$)	0.82
$K_v3.3$	uSciML	iSciML	0.0001	704 918 ($n = 1645$)	0.01
$K_v3.3$	uSciML	cHH	−0.1001	21 429* ($n = 1645$)	1.42
$K_v3.3$	uSciML	cHH with randeff.	−0.0035	260 294* ($n = 1645$)	0.56
$K_v3.3$	iSciML	cHH	−0.1000	16 364* ($n = 1645$)	1.42
$K_v3.3$	iSciML	cHH with randeff.	−0.0027	267 241* ($n = 1645$)	0.56
$K_v3.3$	cHH	cHH with randeff.	0.0756	1 240 130* ($n = 1645$)	1.08
$K_v3.4$	uSciML	iSciML	0.0003	152 095 ($n = 737$)	0.04
$K_v3.4$	uSciML	cHH	−0.0839	10 639* ($n = 737$)	1.27
$K_v3.4$	uSciML	cHH with randeff.	−0.0023	60 688* ($n = 737$)	0.28
$K_v3.4$	iSciML	cHH	−0.0825	9386* ($n = 737$)	1.26
$K_v3.4$	iSciML	cHH with randeff.	−0.0022	62 971* ($n = 737$)	0.24
$K_v3.4$	cHH	cHH with randeff.	0.0705	264 187* ($n = 737$)	1.16
$K_v4.1$	uSciML	iSciML	0.0001	55 999 ($n = 447$)	0.04
$K_v4.2$	uSciML	iSciML	−0.0014	17 330* ($n = 317$)	0.10
$K_v4.3$	uSciML	iSciML	0.0007	108 508* ($n = 542$)	0.16
$K_v4.3$	uSciML	cHH	−0.0433	5932* ($n = 542$)	1.17
$K_v4.3$	uSciML	cHH with randeff.	−0.0090	9552* ($n = 542$)	0.82
$K_v4.3$	iSciML	cHH	−0.0456	3513* ($n = 542$)	1.20
$K_v4.3$	iSciML	cHH with randeff.	−0.0105	4681* ($n = 542$)	0.91
$K_v4.3$	cHH	cHH with randeff.	0.0303	140 229* ($n = 542$)	0.90
$K_v10.1$	uSciML	iSciML	0.0052	220 529* ($n = 709$)	0.52
$K_v10.2$	uSciML	iSciML	0.0044	40 733* ($n = 319$)	0.30
$K_v12.1$	uSciML	iSciML	0.0020	357 783* ($n = 943$)	0.21
$K_v12.3$	uSciML	iSciML	0.0021	35 206* ($n = 328$)	0.16

When more than two models are evaluated for a given channel type, the highlighted row contains the two best performing models. Abbreviations for models $\mathcal{M}_1$ and $\mathcal{M}_2$: cHH – classical Hodgkin-Huxley model, cHH with randeff. – classical Hodgkin Huxley model with random effects, iSciML – individual SciML Hodgkin-Huxley model, uSciML – unified SciML Hodgkin-Huxley model.

W - Wilcoxon rank-sum test statistic

‡ Effect sizes evaluated via Cohen's d follow the following rule of thumb [31]: $d > 0.2$ – small, $d > 0.5$ – medium, $d > 0.8$ – large, $d > 1.2$ – very large, $d > 2.0$ – huge.

* $p < 10^{-6}$

Bonferroni corrections were applied due to multiple comparisons.

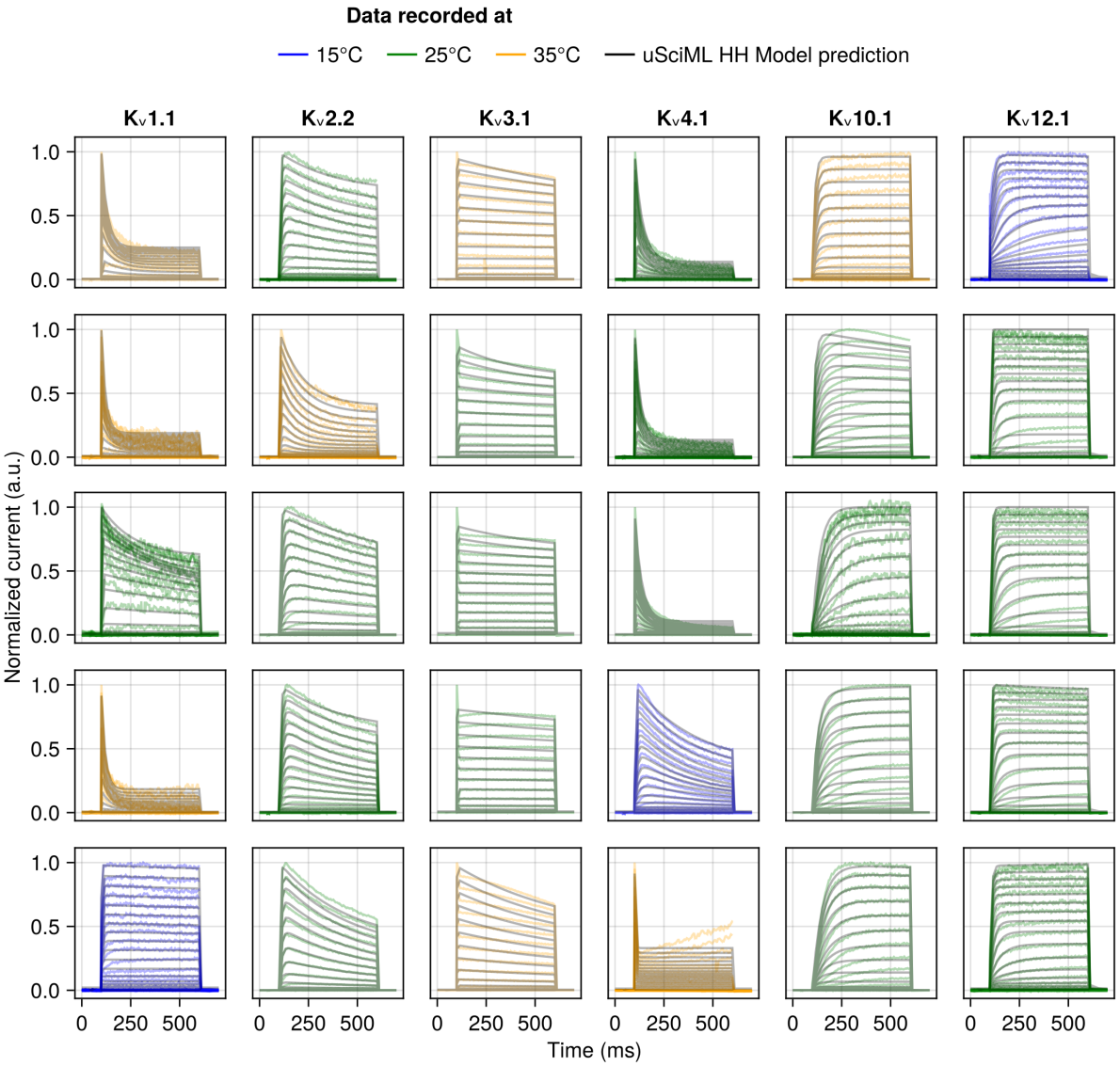


Fig 5. Samples of the unified SciML HH model predictions for test data for the activation protocol for a subset of channel types used in training: $K_v1.1$, $K_v2.2$, $K_v3.1$, $K_v4.1$, $K_v10.1$ and $K_v12.1$. The model predictions are plotted as black lines, whereas the empirical data is colour coded by the temperature at which it was recorded – blue for 15°C, green for 15°C and yellow for 35°C. Each subplot represents measurements from a different cell.

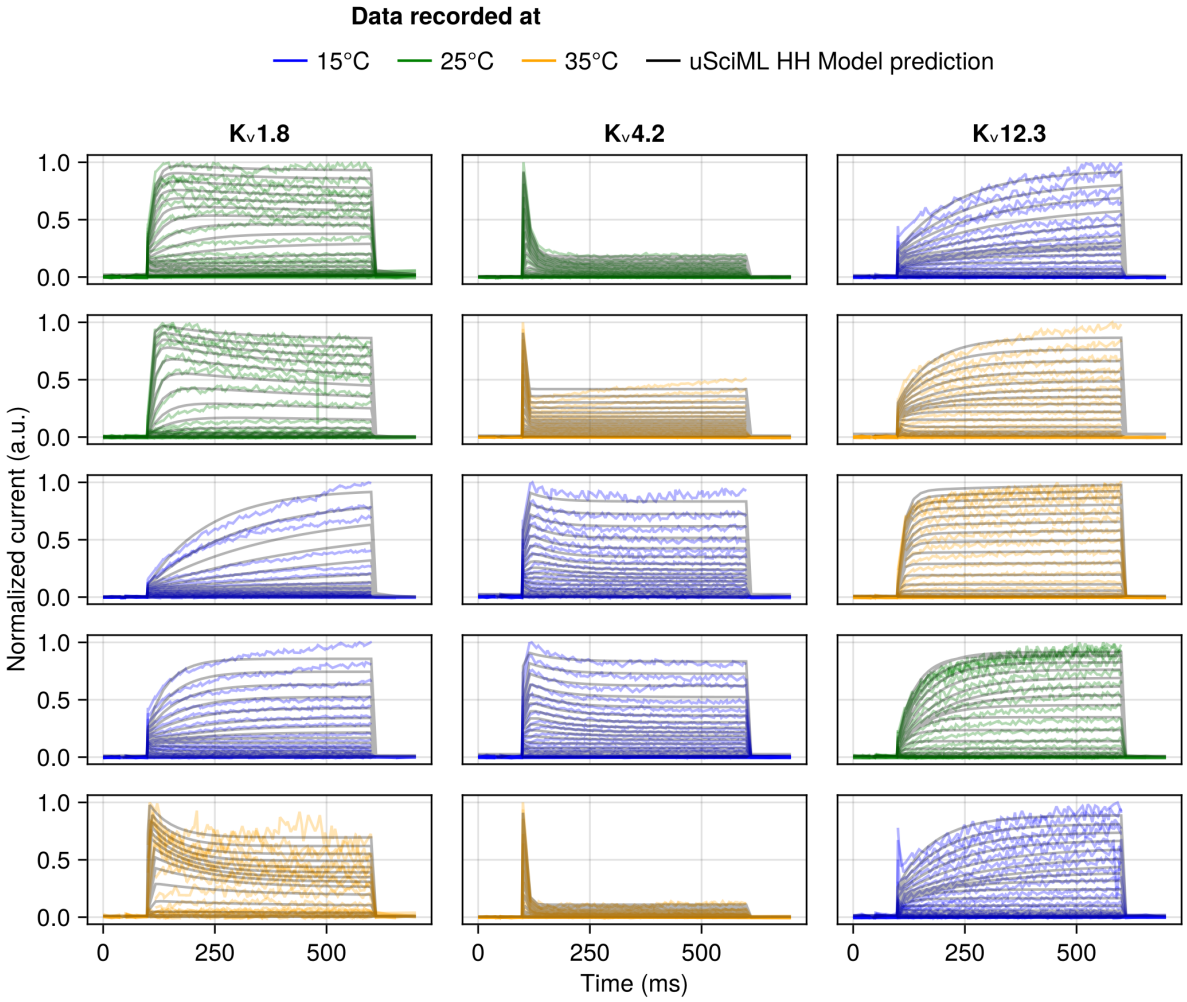


Fig 6. Samples of the unified SciML HH model predictions for the activation protocol of the test data for a subset of channel types not used in training ($K_v1.8$, $K_v4.2$, $K_v12.3$) and therefore represent the model's ability to generalize. The model predictions are plotted as black lines, whereas the empirical data is colour coded by the temperature at which it was recorded – blue for 15°C, green for 25°C and yellow for 35°C. Each subplot represents measurements from a different cell.

effects significantly outperformed the version without the random effect (see Table 2 for more details).

Secondly, comparing the individual SciML models against the classical HH models with random effects allows us to determine which model contains more appropriate functional forms for modelling the K_v currents. The functions used to model the gating variables in the classical HH models are constructed by hand and in some cases with biophysical support. We hypothesised that neural networks offer a more flexible and more powerful approach to constructing gating variable functions than the classical approach. As shown in Fig 4 and in Table 2, the individual SciML HH models fit the data significantly better than the classical HH models with random effects in all seven cases. This comparison indicates that the flexibility in the voltage-dependent gating functions of individual SciML HH models supported better fits to the Ranjan et al. data than the classical HH models.

Finally, we compare the unified SciML HH model and the individual SciML HH models. Under ideal circumstances, an individualised channel model would always outperform a more unified model due to its ability to specialize on a particular K_v type. However, given that the data is limited and there are similarities between the different K_v channel dynamics, a unified model may offer comparable or better performance along with the possibility of generalizing to channel types on which it was not trained. As shown in Fig 4 and in Table 2, the unified SciMLHH model outperformed the individual SciML HH models in five out of 20 cases, gets outperformed in 11 out of 20 cases and in four out of 20 cases there is no significant difference. Therefore, the individual SciML HH models outperform the unified SciML HH model on the majority of K_v channel types.

However, looking at the effect sizes of the differences (Table 2), the differences are generally small (Cohen's $d > 0.2$), except for $K_v10.1$ where the difference is medium (Cohen's $d > 0.5$). Importantly, the unified SciML HH model performs similarly well to the individual SciML HH models on data from channel types which were not in the training data ($K_v1.8$, $K_v4.2$, $K_v12.3$, see Fig 4 subplots with light red background). Therefore, despite the statistically significant better performance of the individual SciML HH models in a higher proportion of cases, since the effect sizes are generally small and the unified SciML HH model offers a much more parsimonious representation of the data with the ability to generalize to K_v channels on which it was not trained, we conclude that the unified SciML HH model is practically preferable to the individual SciML HH models. Therefore, we will restrict further analyses to the unified SciML HH model. We show some example unified SciML HH model predictions on the test data for channel types which were in the training data (Fig 5) and channel types which were not (Fig 6) to illustrate the model's ability to generalize well.

Unified SciML Hodgkin-Huxley model diagnostics

To evaluate the model performance in an absolute sense, rather than relative to other types of models, we examine a number of diagnostics. The first set of diagnostic plots investigates the distributions of pairs of Empirical Bayes Estimates of random effects (EBEs, defined in Equation 7) obtained by fitting them to the test data. As shown in the pair plots in Fig 7 (subplots below the diagonal), the EBEs have a varying range of values, e.g. η_5 or η_8 are relatively more concentrated than η_1 or η_2. However, visually there seem to be relatively few, if any, correlations between the different η_i. The subplots above the diagonal show the RMSE values associated with the pairs of EBEs, and could highlight regions of EBEs which would systematically have higher errors, but no such patterns emerged. We conclude that there is no significant mismatch between the random effect prior and the posterior.

The second set of diagnostic plots (Fig 8) are classical NLME residuals diagnostics.

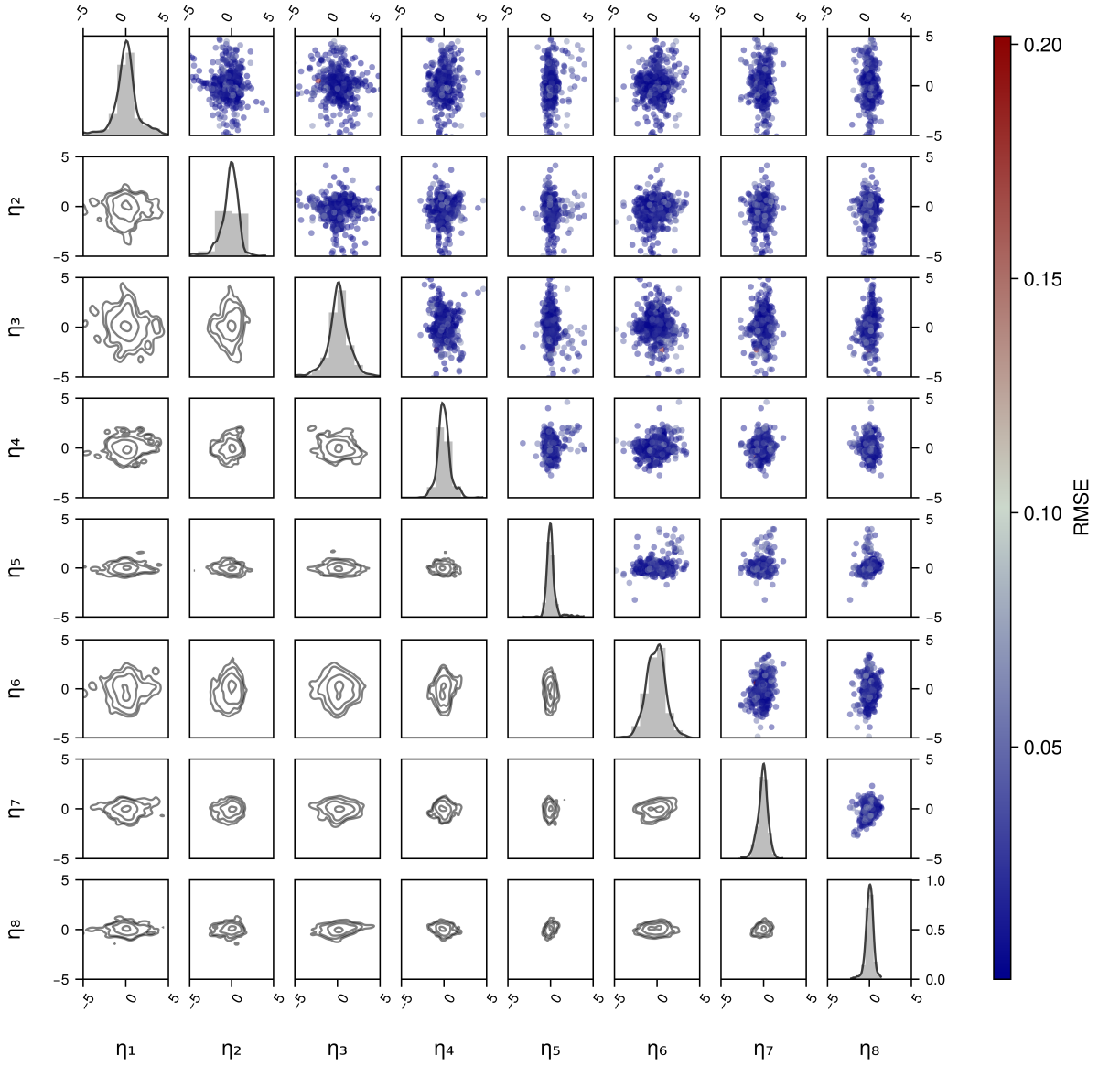


Fig 7. Pair plots of the Empirical Bayes estimates (EBEs) of random effects obtained on the cells in the test data set. The diagonal shows the histogram and a kernel density fit to the EBE distributions. Sub-diagonal subplots show the contour plots of the distribution of pairs of EBEs. Super-diagonal subplots show the scatter plot of the pairs of EBEs with points coloured by the RMSE at the sweep level to check if certain regions of the distributions contain high RMSE points.

Fig 8A shows the relationship between the model population level predictions (i.e. not for a particular cell, but for a channel type) and all the observations for that channel type. An ideal model would have a Gaussian error around the yellow $y = x$ line, which denotes a perfect relationship between the observed values and the values predicted by the model. The unified SciML HH model shows approximately this behaviour as both the local linear fits to the contour lines (LOESS, red line) and the global linear fit (OLS, green line) are very close to the ideal line of $y = x$ (yellow line). The error distribution exhibits some heteroscedasticity – the width of the error distribution decreases with increasing x values. However, overall, given that the heteroscedasticity is mild, we conclude that the model fits the data reasonably well at the population level.

Visually, the individual predictions (Fig 8B) are significantly more accurate than the population level predictions (Fig 8A). For the individual predictions, the global and local linear fits nearly coincide with the $y = x$ line and there is a high concentration of contour lines around the $y = x$ line with a relatively small standard deviation. Therefore, we conclude that the individual fits obtained via the unified SciML HH model are better than the population fits.

Fig 8C analyses the population level residuals weighted by the inverse standard deviation of the observation model and their dependence on time. Therefore, in this plot the residuals can be larger than 1 if a model has a small σ value in the observational model. A well-performing model would show no dependence of the residuals on time, which is nearly true of the unified SciML HH model – the local and the global linear fits (red and green lines) are virtually overlapping and are on the $y = 0$ line. The wider distribution of the contour plot of the residuals at earlier times might indicate that there is a higher spread of the residuals during those times. However, the length of the recording from a cell was proportional to the number of stimulation protocols and sweeps which were of high enough quality. Therefore, since a significant amount of data was filtered out, there are more shorter recordings than longer ones, and the more numerous shorter recordings will tend display larger variability than the fewer long ones. Therefore, the higher spread of the contour lines at earlier times is most likely a feature of the data used, rather than biased model performance.

Finally, Fig 8D plots individual predictions against weighted residuals of individual predictions. Under ideal circumstances the plot would show a $\mathcal{N}(0, 1)$ distribution around the $y = 0$ line. The model results show the correct mean behaviour both locally and globally (red and green lines basically overlapping), but not ideal spread that is skewed towards the negative values with a standard deviation larger than one (similar skewness can also be noted in the bottom left plot). This skewness is most likely due to the model structure and normalization of the data. The model cannot produce negative normalized currents, whereas, due to the noise inherent in the data, current values can be negative and they get up-weighted by a small σ in the observational model. Therefore, these diagnostics support the conclusion that the unified SciML model predictions are generally unbiased, both at the population and the individual cell levels.

The final set of diagnostics are the visual predictive checks (VPCs) shown in Fig 9. The VPCs are given for each sweep of each protocol to provide a more granular diagnostic of the model's performance. For a given model VPCs a produced by sampling k random effects which produces n populations. For each population of k samples we calculate the 95% confidence intervals for the 10%, 50% and 90% quantiles of the data (shaded areas) and then plot the actual observed quantiles. If the observed quantiles fall into the simulated 95% CI of the quantiles, the model is representative of the data to which it was fit.

Overall, the VPCs support the conclusion that the unified SciML HH model fits the population level data reasonably well. However, most likely due to the differing amounts of data available for each protocol, there are some differences between the

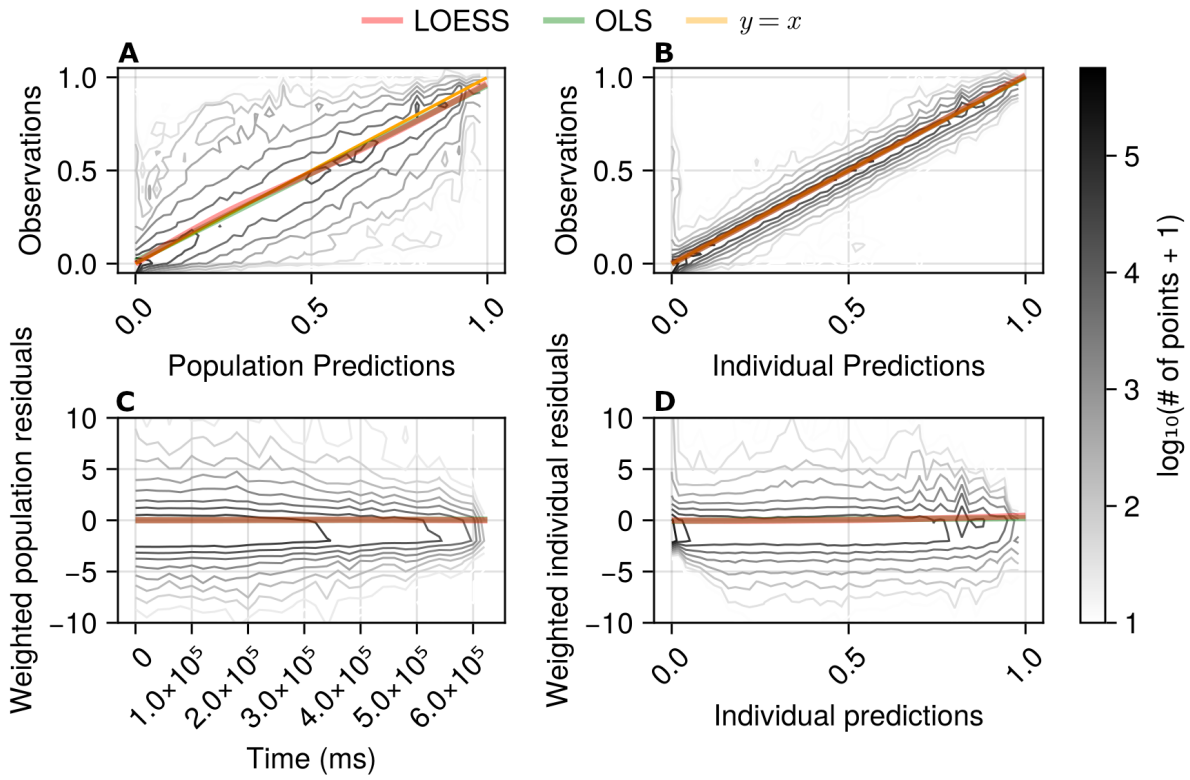


Fig 8. Goodness of fit visualizations for the unified SciML HH model on the test data set. Panels contain: A – observations plotted against population predictions, B – observations plotted against individual predictions, C – weighted population residuals plotted against time, D – weighted individual residuals plotted against individual predictions. Each subplot contains either two or three lines: the yellow $y = x$ line (where x is the prediction and y are the observed values), in the top two plots shows an ideal scenario, the green line represents a global ordinary least squares (OLS) fit and the red line represents the locally estimated scatter plot smoothing (LOESS) fit to the data shown in each subplot. The contour line levels represent the $\log_{10}(\#\text{ of points} + 1)$ (differentiated by the intensity of the contour and shown in the colour bar on the right) and are based on approximately 3 million observed data points present in the test data set.

protocols. More specifically, the activation and deactivation protocols show mostly good VPCs, whereas the inactivation and the recovery protocols show still mostly good, but somewhat worse VPCs. Specifically, for the inactivation protocol the model consistently under-predicts the 10% quantile relative to the data. Similarly for the recovery protocol, the model systematically under-predicts the 10% quantile. Therefore, the VPCs indicate that some caution is warranted when the unified SciML HH model is used to simulate channel dynamics involving inactivation or recovery, but otherwise provides general support to the quality of the population predictions of the model.

We have provided three model diagnostic plots for the unified SciML HH model, investigating different, if overlapping, aspects of its performance: whether the EBEs are correlated, the residuals and the VPCs. The diagnostics lend significant support to the conclusion that the model predictions for individual cells are highly accurate and the population level predictions are reasonably accurate.

Unified SciML Hodgkin-Huxley model gating function behaviour

Having established the quality of the unified SciML HH model, we now investigate its properties, starting by showing the fitted $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ functions. We plot the fitted functions for the individual cells for the test data set in Fig 10 and the population predictions in Fig 11. We provide both plots to illustrate the variability of the gating functions in the test data, and to show the population level predictions made for temperatures not included in the training data.

The individual fitted functions (Fig 10) exhibit several patterns. Firstly, $m_{2,\infty}(V,T)$ approximately follows the classical sigmoidal behaviour of an activation gate, increasing in value as the voltage increases, reaching the maximal value of 1 at different voltages for different K_v types. It is notable that $m_{2,\infty}(V,T)$ shows a small amount of temperature dependence.

The steady state behaviour of the first gating particle, $m_{1,\infty}(V,T)$, however, varies between cells, displaying three clusters of behaviours. Firstly, for some channel types (K_v10.1, K_v10.2, K_v12.1 and K_v12.3) $m_{1,\infty}(V,T)$ does not show significant dynamics, implying that for those channel types a single gating variable would have been sufficient. Next, a few channel types display a classical inactivating behaviour that may reach close to full (K_v1.3 and K_v3.4) or partial inactivation (K_v1.5, K_v3.1 andK_v4.3). For all remaining channels, $m_{1,\infty}(V,T)$ shows a mix of inactivating behaviour at low voltages and activating behaviour at high voltages (e.g. see K_v1.8 or K_v4.2, especially at 15°C). This behaviour illustrates the flexibility of the SciML approach. This behaviour may also indicate that adding a third gating variable may provide small improvements for the channel types showing this behaviour in $m_{1,\infty}(V,T)$.

Contrasting $\tau_1(V,T)$, the time constant associated with $m_{1,\infty}(V,T)$, and $\tau_2(V,T)$, the time constant associated with $m_{2,\infty}(V,T)$, it is clear that $\tau_1(V,T)$ is generally either only as fast or significantly slower than $\tau_2(V,T)$. Therefore, the main activation represented by $m_{2,\infty}(V,T)$ is fast, and the gating dynamics modelled via $m_{1,\infty}(V,T)$ are slow. Generally the $\tau_i(V,T)$ behave as expected when temperature is increased, in they get smaller (i.e. the equilibration is faster). Recall that $\tau_i(V,T) = \frac{1}{\alpha_i(V,T)+\beta_i(V,T)}$ – if the temperature is increased, reaction rates $\alpha_i(V,T)$ and $\beta_i(V,T)$ tend to increase, decreasing the time constant $\tau_i(V,T)$.

Contrary to the results shown in Fig 10 which are based on actual fits using η obtained from individual cells in the test data set, the plots in Fig 11 are obtained by setting $\eta = 0$ and setting temperature $T \in [15, 20, 25, 30, 35]$. Therefore, Fig 11 provides predictions in cases where no data is available, specifically at temperatures other than 15°C, 25°C and 15°C. For both $m_{1,\infty}(V,T)$ and $m_{2,\infty}(V,T)$ their dynamics across temperatures are monotonic, suggesting that the model may generalize reasonably to

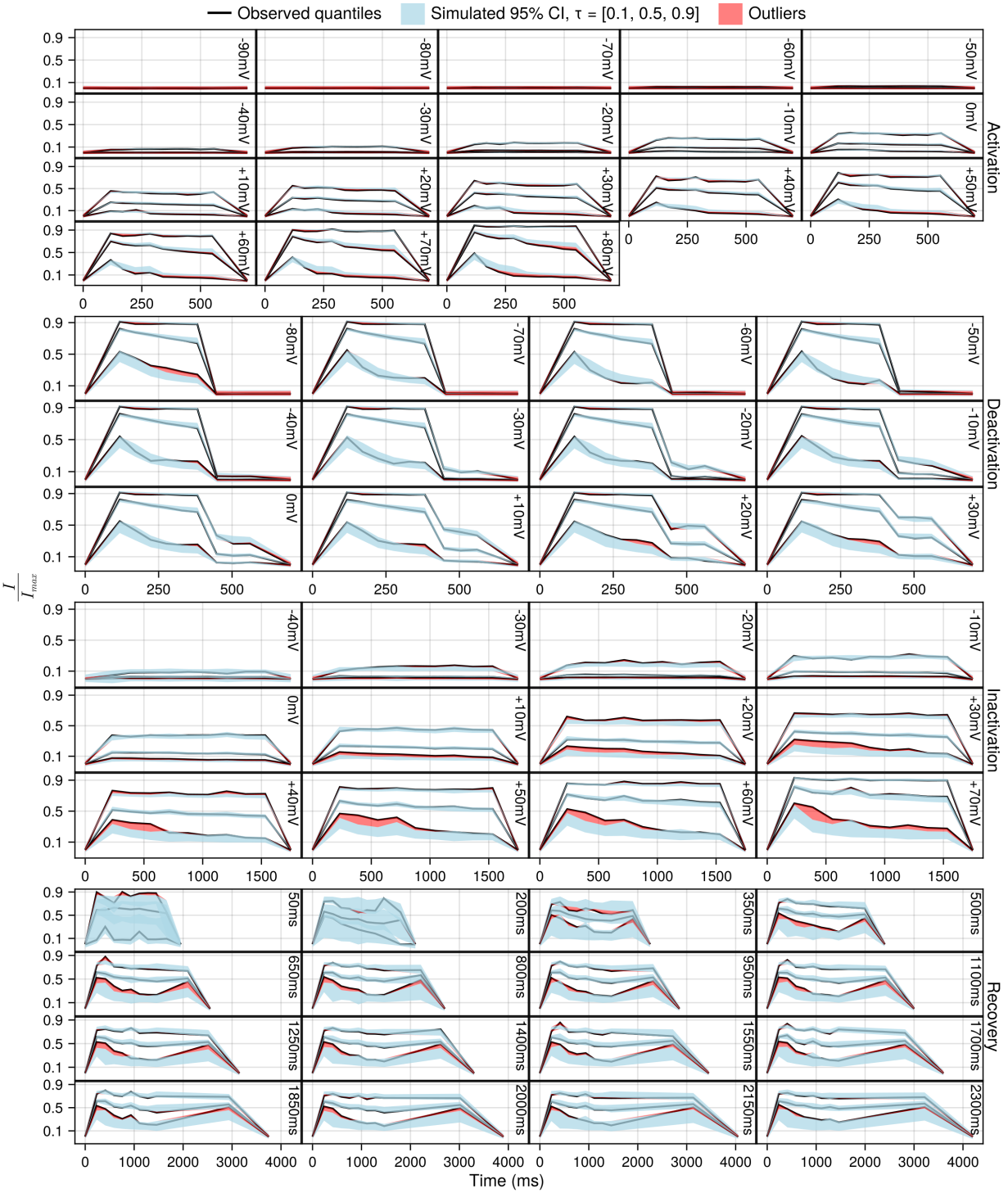


Fig 9. Visual predictive checks (VPCs) for the unified SciML HH model with one subplot for each sweep of each protocol. Each subplot contains three black lines representing the 10%, 50%, 90% quantiles of the data, as well as three light blue shaded areas for the simulated 95% confidence intervals for the 10%, 50%, 90% quantiles. The red shaded areas show where the observed quantile lines are outside the simulated quantiles.

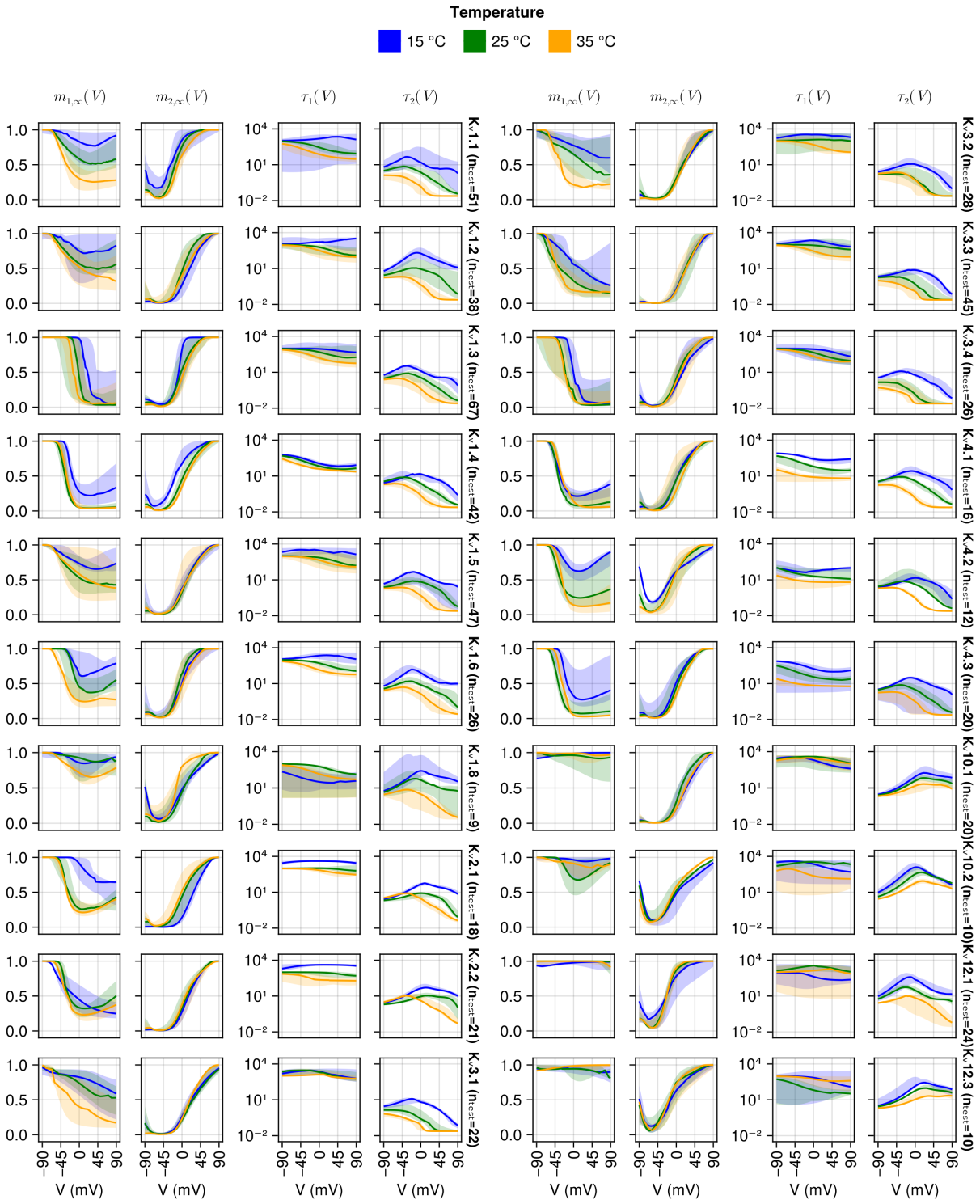


Fig 10. Individual predictions of the $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ functions with the EBEs for the cells in the test data. Each column represents one plots one function and its dependence on voltage and the plot is split into two groups, the left batch of K_v channels and the right batch of K_v channels, indicated by the names after the fourth and the eighth columns with the number of cells n in the test data for that channel type. Most cells had data in all three temperatures – 15°C (blue), 25°C (green) and 35°C (yellow). The solid lines represent the median and the shaded area is the 95% confidence interval.

other temperatures. It is notable that $\tau_1(V,T)$ can take various different functional forms (e.g. see $K_v1.8$ or $K_v3.4$) across temperatures. Finally, $\tau_2(V,T)$ shows a fairly consistent functional shape with the largest differences between channel types at higher temperatures.

Finally, Fig 12 shows the $Q_{10}(V)$ values for ratios of the gating time constant functions evaluated at 15°C and 25°C (blue lines), and the ratios of functions evaluated at 25°C and 35°C (orange lines). The most striking feature of these plots is the $Q_{10}(V)$ values taken for the activation time constant $\tau_2(V,T)$, e.g. 100 at +80mV for $K_v3.3$ and smaller (but still large) for other channel types. These $Q_{10}(V)$ values significantly exceed any previous Q_{10} values (e.g. 2-4, see [1]) used in models of ion channels, as well as empirical estimates provided in [4]. The $Q_{10}(V)$ values for $\tau_1(V,T)$ are better aligned with the literature. Importantly, most channel types show significant voltage dependencies for $Q_{10}(V)$ that can be qualitatively different between the two calculated ratios. Data measured at a set of intermediate temperatures between 15°C and 35°C may be necessary to better constrain the $Q_{10}(V)$ values of the unified SciML HH model.

Discussion

In this study, we employed scientific machine learning (SciML) and non-linear mixed effects modelling (NLME) to address the challenge of modelling the gating kinetics of a diverse set of voltage-gated potassium ion (K_v) channels, using data from Ranjan et al. [4]. To our knowledge, no previous work has modelled this data set fully, in a way that would fit individual cell recordings for a large number of cells and the majority of the recording protocols. Traditional Hodgkin-Huxley-like (HH) approaches to modelling ion channel gating require assumptions (or costly optimization techniques) to determine the appropriate number of gating particles, the functional forms and parameters describing the voltage dependencies of these particles, and the exponents applied to the gating particles. In contrast, we utilized the SciML framework, replacing the traditional functional forms of $m_{i,\infty}(V)$ and $\tau_i(V)$ with neural networks that were appropriately constrained. This allowed us to preserve the general HH framework while only assuming the number of gating particles. All other parameters were then fitted using efficient, gradient-based optimization techniques [23].

The data we fit the models to showed significant heterogeneity [4]. In addition to the expected differences in gating kinetics between distinct K_v channel types, variability was also observed within cells expressing the same K_v type. This within-type variability is challenging to capture with conventional methods for modelling voltage-gated ion channels. However, the NLME approach is specifically designed to account for both between-type and within-type variability by modelling the latter as random effects.

We explored two distinct SciML HH modelling approaches: (i) constructing individual SciML HH models for each K_v type, and (ii) developing a unified SciML HH model that accommodates all K_v types for which there was data, with channel type mapped onto the random effects through an additional neural network NN_{aug}. To assess the relative performance of the SciML HH K_v models, we compared them to a set of seven previously published HH models. Since the existing HH models do not account for within-K_v-type variability, we incorporated random effects into the parameters of the seven published K_v channels. This enabled us to isolate the contributions of the SciML approach – i.e. the data-driven learning of voltage-dependent functions – and evaluate the impact of incorporating random effects in fitting the Ranjan et al. [4] dataset.

Our comparisons demonstrated that both the addition of random effects and the use of the SciML approach independently improved the fit of the K_v models to the data (see Fig 4). The seven existing HH models fit significantly better when random effects were added. Furthermore, the HH models with random effects were significantly

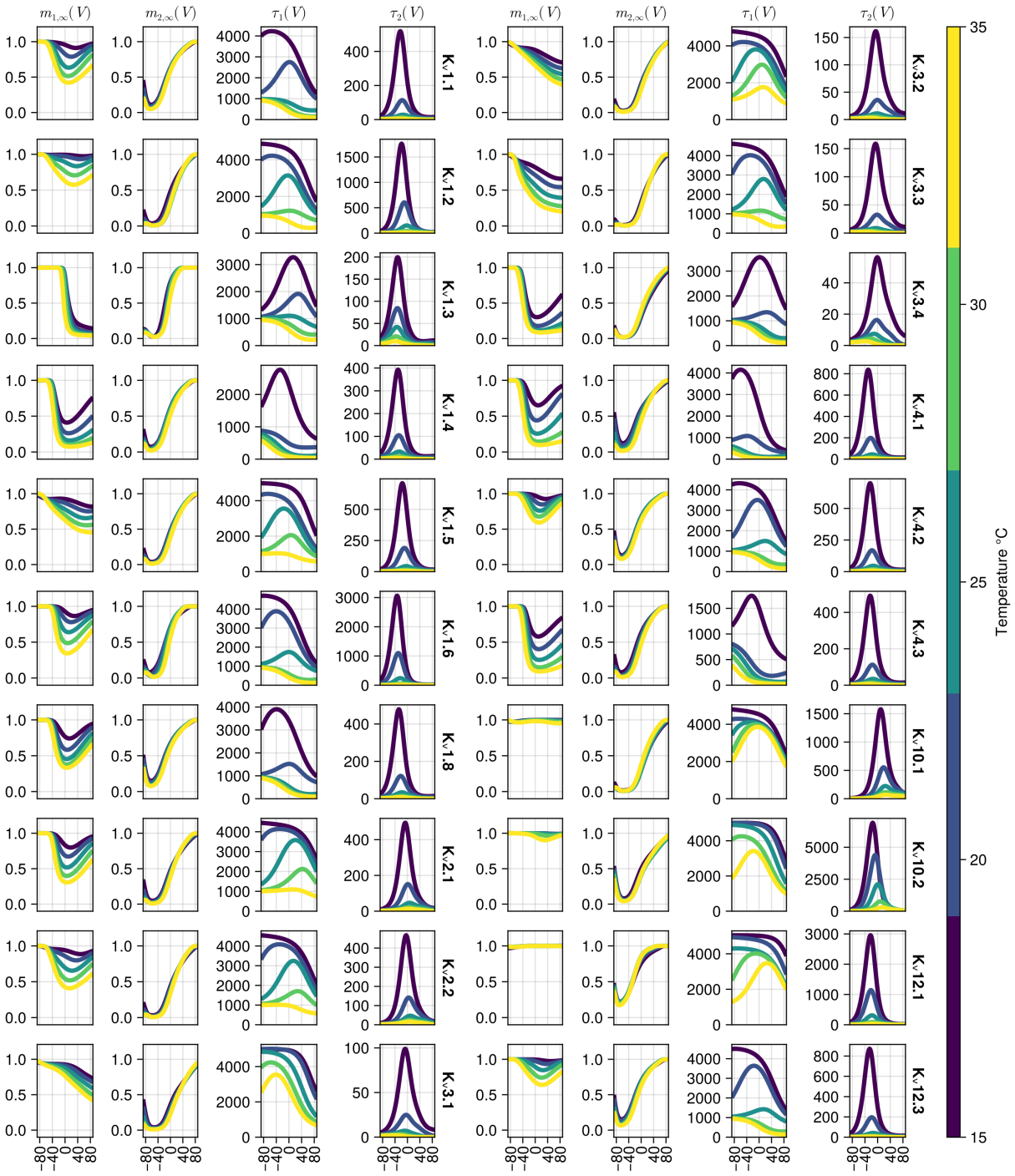


Fig 11. Population predictions of the $m_{i,\infty}(V,T)$ and $\tau_i(V,T)$ functions with the $\eta = 0$. Each column represents plots for one function and its dependence on voltage and the plot is split into two groups, the left batch of K_v channels and the right batch of K_v channels, indicated by the names after the fourth and the eighth columns. The colour bar on the right indicates the temperature at which the functions are evaluated.

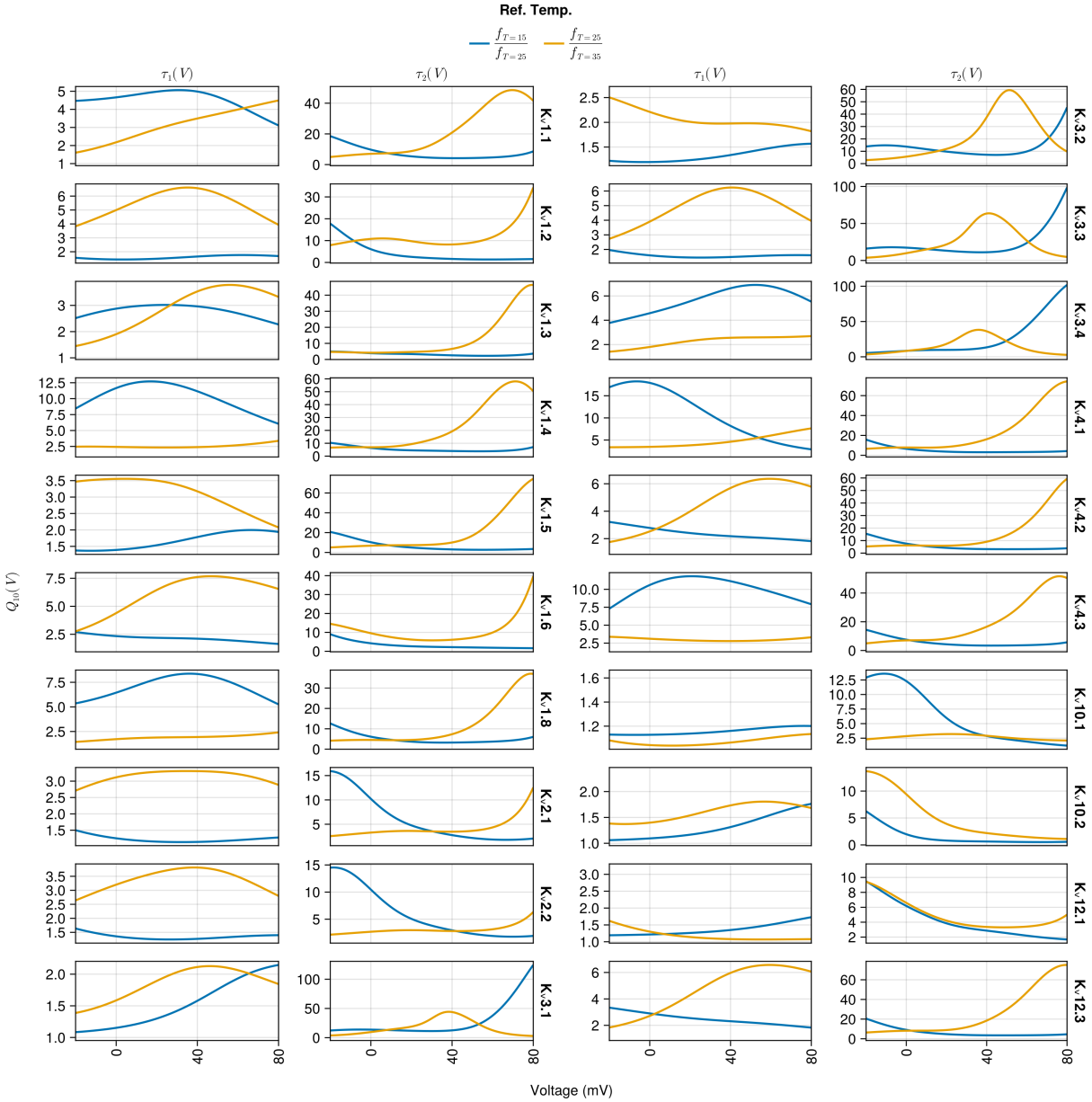


Fig 12. $Q_{10}(V)$ values obtained by evaluating the unified SciML HH model with $\eta = 0$ and calculating the ratios of the gating time constant functions $\tau_1(V,T)$, $\tau_2(V,T)$ values evaluated at 15°C and 25°C (blue line) as well as 25°C and 35°C (orange lines). The plot is split into two blocks (two columns each) and each row within a block is for an individual K_v channel type.

outperformed by the two SciML HH models, in which neural networks represented $m_{i,\infty}(V)$ and $\tau_i(V)$. The next question was whether the individual or unified SciML HH approach performed better. Comparing the two approaches revealed that the individual SciML HH models generally outperformed the unified model. This result is expected, as the neural network architectures used in both approaches were identical. The individual models could specialize more effectively to the specific gating dynamics of each K_v type, whereas the unified model had to account for all types. However, while the differences between the individual and unified models were statistically significant, they were generally small. Notably, the unified SciML HH model was able to generalize to three additional K_v types that it was not trained on. Given the small performance differences and the advantage of parsimony, we concluded that the unified SciML HH model is preferable. It is also highly extensible to many more voltage-gated channel types. Therefore, our primary contribution is the development, validation, and analysis of the unified SciML HH model, capable of modelling the gating dynamics of at least 20 different K_v channel types.

There are multiple ways to fit HH models to current recordings and it is worth comparing the software used in this study (`Pumas.jl` and `DeepPumas.jl`) to other available software packages. For example, `Data2Dynamics` (`https://github.com/Data2Dynamics/d2d`) is a MATLAB-based software package capable of optimizing the reaction rates of chemical reaction networks in an efficient manner [32]. This software suite was used in an attempt (`https://github.com/njohner/Kv-kinetic-models`) to model the K_v data from Ranjan et al [23]. Even though `Data2Dynamics` offers powerful capabilities in optimizing the reaction rates of chemical reaction networks, it does not include algorithms for the optimization of the structure of the networks. Therefore, while powerful in certain use cases, the problem tackled in this study requires a richer set of functionalities, notably usage of neural networks. Another option that is more specialized to computational neuroscience is the Python-based `BluePyOpt` package [33]. It has been used in studies to optimize the conductances of ion channels along the dendritic tree of a compartmental neuron model [34, 35]. It uses a set of evolutionary algorithms to fit various parameters, such as the maximal channel conductance, distribution of conductances along a dendrite, but it can also be used to optimize the parameters that determine the kinetic properties of individual channel types. However, `BluePyOpt` does not directly optimize the structure of the functions used for channel gating and therefore does not offer all the functionality necessary to tackle the current challenge. Without going into an extended review of the available tools for fitting chemical reaction networks to data, based on the two aforementioned examples, we conclude that `Pumas.jl` and `DeepPumas.jl` offer some of the currently most powerful and feature complete tools and were essential in modelling the different types of K_v channels. Therefore, our second contribution is the application of a novel set of approaches, SciML and NLME, to a challenge in computational neuroscience. These tools can be applied to other challenges where significant variance between observed entities is present, for example, modelling of synaptic signalling stemming from highly variable proteomic structures [36].

The creation of a unified model for 20 different K_v channels represents an important first step, but there are many potential directions for future development. The most immediate and promising future direction is extending the current unified model to include recently published data [4] on other major voltage-gated ion channels (K_{2P}, K_{ir}, K_{Ca}, Na_v, Ca_v, and HCN). While some families of voltage-gated channels, such as Kir and Nav, should integrate smoothly into the current unified model – potentially even falling within its existing predictive range – other families with more complex gating mechanisms (e.g. K_{Ca} and Ca_v) will require more sophisticated adaptations. For these, additional components, such as calcium dynamics, may need to be incorporated into the

model. Moreover, there is no theoretical limit to the number of temporally varying signals (such as neuromodulators, drugs, etc.) that could be fed as inputs to the neural networks representing the gating functions, depending on the available data. However, practical challenges arise when adding more dimensions or inputs to the neural networks, and alternative approaches would need to be explored to assess their feasibility.

One of the potential challenges for the unified SciML HH model lies in its use of one-hot encoding to represent different channel types. If this encoding is naively extended by increasing its dimension, it may lead to an augmentation neural network $\mathrm{NN}_{\mathrm{aug}}$ with too many parameters, making the model prohibitively expensive to fit. A more practical long-term solution could involve a dimensionality reduction approach, perhaps based on factors like the 3D structure of the channel or its amino acid sequence. The latent representations derived from such an encoding could then be combined with the random effects or used to replace the current one-hot vector altogether. This approach could enable the model to predict the impact of point mutations or other modifications on channel kinetics. Furthermore, encoding the 3D structure or amino acid sequence would facilitate the creation of species-agnostic ion channel models, allowing for the integration of data from different species. A similar dimensionality reduction approach could be applied to external factors affecting channel kinetics, such as neuromodulators or drugs, before passing these latent representations into the neural networks that model the kinetics. In essence, the use of a foundation model – currently popular in deep learning – could be highly beneficial for modelling ion channel gating [37]. It remains to be seen what predictive and explanatory power such a model could offer.

The HH approach to modelling ion channels – where channels are represented by a number of independent gating particles – is just one of several modelling paradigms. In fact, it is a subclass of the broader Markov models, where the gating particles may not be independent. For additional modelling approaches, see [38]. Recent studies have used single-channel patch-clamp recordings combined with deep learning to infer the most likely Markov schemes underlying the recorded data [11, 39]. Although their approaches may initially appear similar to ours, there are several key differences. Firstly, the studies by [11, 39] rely on single-channel patch-clamp recordings, which are more challenging to obtain compared to the whole-cell patch-clamp data used in Ranjan et al. [4]. Consequently, these approaches are suited to different types of questions. For instance, if the focus is on understanding the impact of phosphorylation on the gating of individual channels, the approach taken in [11, 39] would likely be more appropriate. However, if the goal is to model the aggregate dynamics of many channels within a large patch of membrane, our approach is likely to provide more accurate predictions. Moreover, our approach is better equipped to handle variability from different sources, such as differences in gating kinetics between cells expressing the same $\mathrm{K_v}$ type or across different $\mathrm{K_v}$ types. Additionally, extending the gating mechanisms to include Ca^{2+} or other ligands, in addition to voltage, may present more challenges for the Markov scheme approach due to the exponential growth in the number of states. Theoretically we could take different Markov schemes and represent the voltage-dependent gating functions as neural networks, but such an approach would require significant efforts to find an appropriate Markov scheme. Therefore, the approaches in [11, 39] and ours may be complementary. If the HH formalism proves insufficiently expressive for modelling a broader range of ion channel types, the approach of [11, 39] could provide a more universally appropriate Markov scheme. Our approach could then be applied to model the transition rates within it.

In addition to its explanatory and predictive benefits, a foundational approach to modelling ion channel gating could offer practical advantages for modellers. Current detailed compartmental models of neurons (written in e.g. `NEURON`) typically require

multiple files, each representing a single type of ion channel. There is an abundance of such files in various repositories and databases (notably `https://modeldb.science/` [7]), but these files often differ in their underlying models, even for the same ion channel type. Determining which model is most appropriate and provides the best fit to the data can be difficult and time-consuming. However, if a well-validated foundational model were available, this problem would be significantly alleviated. The burden of creating and maintaining a large number of files for each ion channel would be greatly reduced. Our unified SciML HH model could currently be used in a `NEURON .MOD` file by incorporating a pre-constructed lookup table for different K_v types. However, a significant challenge arises from the inability to use multiple lookup tables within the same `.MOD` file. Other data formats, such as NeuroML [40], may present their own challenges when attempting to port our unified SciML HH model to them. If foundation channel gating models are developed in the future, it would be important to collaborate with software developers of compartmental neural modelling simulators.

In conclusion, we successfully addressed a major challenge in neuroscience: the efficient and accurate modelling of voltage-gated potassium channel gating. By combining two novel approaches in computational neuroscience – scientific machine learning and non-linear mixed effects modelling – we developed a unified model capable of fitting the gating dynamics of 20 different K_v types. Our work demonstrates how these approaches can be seamlessly integrated into existing computational neuroscience approaches. Moreover, the produced unified K_v channel gating model may serve as the first step in creating a next generation foundation model that would be able to model all known voltage-gated channel families. The resulting gains in predictive and explanatory power, as well as computational efficiency could help to tackle complex challenges, such as investigating diseases related to ion channels [41].

Supporting information

S1 Appendix. Hyper-parameter optimization.

References

1. Hille B. Ionic Channels of Excitable Membranes. Sinauer; 2001. Available from: `https://books.google.co.uk/books?id=8Vk-QwAACAAJ`.
2. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. The Journal of Physiology. 1952;117(4):500–544. doi:https://doi.org/10.1113/jphysiol.1952.sp004764.
3. Thorpe SJ. Spike arrival times: A highly efficient coding scheme for neural networks; 1990.Available from: `https://api.semanticscholar.org/CorpusID:59901370`.
4. Ranjan R, Logette E, Marani M, Herzog M, Tâche V, Scantamburlo E, et al. A Kinetic Map of the Homomeric Voltage-Gated Potassium Channel (Kv) Family. Frontiers in Cellular Neuroscience. 2019;13. doi:10.3389/fncel.2019.00358.
5. Alexander SPH, Mathie AA, Peters JA, Veale EL, Striessnig J, Kelly E, et al. British Journal of Pharmacology. 2023;180(S2):S145–S222. doi:https://doi.org/10.1111/bph.16178.

6. Johnston J, Forsythe ID, Kopp-Scheinpflug C. SYMPOSIUM REVIEW: Going native: voltage-gated potassium channels controlling neuronal excitability. The Journal of Physiology. 2010;588(17):3187–3200. doi:https://doi.org/10.1113/jphysiol.2010.191973.

7. McDougal RA, Morse TM, Carnevale T, Marenco L, Wang R, Migliore M, et al.. Twenty years of ModelDB and beyond: building essential modeling tools for the future of neuroscience; 2017.

8. Sterratt DC, Sorokina O, Armstrong JD. Integration of Rule-Based Models and Compartmental Models of Neurons. In: Maler O, Halász Á, Dang T, Piazza C, editors. Hybrid Systems Biology. Cham: Springer International Publishing; 2015. p. 143–158.

9. Walch OJ, Eisenberg MC. Parameter identifiability and identifiable combinations in generalized Hodgkin–Huxley models. Neurocomputing. 2016;199:137–143. doi:https://doi.org/10.1016/j.neucom.2016.03.027.

10. Fink M, Noble D. Markov models for ion channels: Versatility versus identifiability and speed. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2009;367:2161–2179. doi:10.1098/rsta.2008.0301.

11. Oikonomou E, Kolan R, Kern L, Gruber T, Alzheimer C, Krauss P, et al.. A deep-learning approach to quasi-instantaneous Markov-modeling of ion channel gating; 2024. Presented as poster in FENS2024 conference. Available from: `https://fens2024.abstractserver.com/program/#/details/presentations/1099`.

12. Lindstrom MJ, Bates DM. Nonlinear Mixed Effects Models for Repeated Measures Data; 1990. Available from: `https://www.jstor.org/stable/2532087?seq=1&cid=pdf-`.

13. Lee JL, Mohamed Shah N, Makmor-Bakry M, Islahudin F, Alias H, Mohd Saffian S. A systematic review of population pharmacokinetic analyses of polyclonal immunoglobulin G therapy. International Immunopharmacology. 2021;97:107721. doi:https://doi.org/10.1016/j.intimp.2021.107721.

14. Adéoti OM, Agbla S, Diop A, Glèlè Kakaï R. Nonlinear mixed models and related approaches in infectious disease modeling: A systematic and critical review. Infectious Disease Modelling. 2025;10(1):110–128. doi:https://doi.org/10.1016/j.idm.2024.09.001.

15. Wörtwein T, Allen NB, Sheeber LB, Auerbach RP, Cohn JF, Morency LP. Neural Mixed Effects for Nonlinear Personalized Predictions. In: Proceedings of the 25th International Conference on Multimodal Interaction. ICMI '23. New York, NY, USA: Association for Computing Machinery; 2023. p. 445–454. Available from: `https://doi.org/10.1145/3577190.3614115`.

16. Xiong Y, Kim HJ, Singh V. Mixed Effects Neural Networks (MeNets) With Applications to Gaze Estimation. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2019. p. 7735–7744.

17. Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Networks. 1989;2(5):359–366. doi:https://doi.org/10.1016/0893-6080(89)90020-8.

18. Awile O, Kumbhar P, Cornu N, Dura-Bernal S, King JG, Lupton O, et al. Modernizing the NEURON Simulator for Sustainability, Portability, and Performance. Frontiers in Neuroinformatics. 2022;16. doi:10.3389/fninf.2022.884046.

19. Bezanson J, Edelman A, Karpinski S, Shah VB. Julia: A fresh approach to numerical computing. SIAM Review. 2017;59(1):65–98. doi:10.1137/141000671.

20. Savitzky A, Golay MJE. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. Analytical Chemistry. 1964;36(8):1627–1639. doi:10.1021/ac60214a047.

21. Jugel U, Jerzak Z, Hackenbroich G, Markl V. M4: a visualization-oriented time series data aggregation. Proc VLDB Endow. 2014;7(10):797–808. doi:10.14778/2732951.2732953.

22. Linkevicius D, Chadwick A, Stefan MI, C Sterratt D. Raw and processed data used in the publication "One model to rule them all: unification of voltage-gated potassium channel models via deep non-linear mixed effects modelling"; 2025. Available from: `addurl`.

23. Rackauckas C, Ma Y, Noack A, Dixit V, Mogensen PK, Elrod C, et al. Accelerated Predictive Healthcare Analytics with Pumas, A High Performance Pharmaceutical Modeling and Simulation Platform. bioRxiv. 2022;doi:10.1101/2020.11.28.402297.

24. Duchesne R, Guillemin A, Gandrillon O, Crauste F. Practical identifiability in the frame of nonlinear mixed effects models: the example of the in vitro erythropoiesis. BMC Bioinformatics. 2021; p. 1–21. doi:10.1186/s12859-021-04373-4.

25. Lee SY. Bayesian Nonlinear Models for Repeated Measurement Data: An Overview, Implementation, and Applications. Mathematics. 2022;10(6). doi:10.3390/math10060898.

26. Shen W, Hernandez-Lopez S, Tkatch T, Held JE, Surmeier DJ. Kv1.2-Containing K+ Channels Regulate Subthreshold Excitability of Striatal Medium Spiny Neurons. Journal of Neurophysiology. 2004;91(3):1337–1349. doi:10.1152/jn.00414.2003.

27. Masoli S, Solinas S, D'Angelo E. Action potential processing in a detailed Purkinje cell model reveals a critical role for axonal compartmentalization. Frontiers in Cellular Neuroscience. 2015;9. doi:10.3389/fncel.2015.00047.

28. Wang LY, Gan L, Forsythe ID, Kaczmarek LK. Contribution of the Kv3.1 potassium channel to high-frequency firing in mouse auditory neurones. The Journal of Physiology. 1998;509(1):183–194. doi:https://doi.org/10.1111/j.1469-7793.1998.183bo.x.

29. Beining M, Mongiat LA, Schwarzacher SW, Cuntz H, Jedlicka P. T2N as a new tool for robust electrophysiological modeling demonstrated for mature and adult-born dentate granule cells. eLife. 2017;6:e26517. doi:10.7554/eLife.26517.

30. Rackauskas Christopher IV. Neural-Embedded Nonlinear Mixed Effects Models (NENLME) in Pumas.jl. In: American Conference on Pharmacometrics; 2019.

31. Sawilowsky SS. Very large and huge effect sizes. Journal of Modern Applied Statistical Methods. 2009;8:597–599. doi:10.22237/jmasm/1257035100.

32. Raue A, Steiert B, Schelker M, Kreutz C, Maiwald T, Hass H, et al. Data2Dynamics: A modeling environment tailored to parameter estimation in dynamical systems. Bioinformatics. 2015;31:3558–3560. doi:10.1093/bioinformatics/btv405.

33. Geit WV, Gevaert M, Chindemi G, Rössert C, Courcol JD, Muller EB, et al. BluePyOpt: Leveraging open source software and cloud infrastructure to optimise model parameters in neuroscience. Frontiers in Neuroinformatics. 2016;10. doi:10.3389/fninf.2016.00017.

34. Masoli S, Rizza MF, Sgritta M, Geit WV, Schürmann F, D'Angelo E. Single neuron optimization as a basis for accurate biophysical modeling: The case of cerebellar granule cells. Frontiers in Cellular Neuroscience. 2017;11. doi:10.3389/fncel.2017.00071.

35. Migliore R, Lupascu CA, Bologna LL, Romani A, Courcol JD, Antonel S, et al. The physiological variability of channel density in hippocampal CA1 pyramidal cells and interneurons explored using a unified data-driven modeling workflow. PLoS Computational Biology. 2018;14. doi:10.1371/journal.pcbi.1006423.

36. Roy M, Sorokina O, McLean C, Tapia-González S, DeFelipe J, Armstrong JD, et al. Regional diversity in the postsynaptic proteome of the mouse brain. Proteomes. 2018;6:1–18. doi:10.3390/proteomes6030031.

37. Bommasani R, Hudson DA, Adeli E, Altman RB, Arora S, von Arx S, et al. On the Opportunities and Risks of Foundation Models. CoRR. 2021;abs/2108.07258.

38. Sansom MS, Ball FG, Kerry CJ, McGee R, Ramsey RL, Usherwood PN. Markov, fractal, diffusion, and related models of ion channel gating. A comparison with experimental data from two ion channels. Biophysical Journal. 1989;56(6):1229–1243. doi:https://doi.org/10.1016/S0006-3495(89)82770-5.

39. Oikonomou E, Gruber T, Chandra AR, Höller S, Alzheimer C, Wellein G, et al. 2D-dwell-time analysis with simulations of ion-channel gating using high-performance computing. Biophysical Journal. 2023;122(7):1287–1300. doi:https://doi.org/10.1016/j.bpj.2023.02.023.

40. Goddard NH, Hucka M, Howell F, Cornelis H, Shankar K, Beeman D. Towards NeuroML: Model Description Methods for Collaborative Modelling in Neuroscience. Philosophical Transactions: Biological Sciences. 2001;356(1412):1209–1228.

41. Musio C. Ion Channels and Neurological Disease. Life. 2024;14(6). doi:10.3390/life14060758.

S1 APPENDIX. HYPER-PARAMETER OPTIMIZATION

I have trained many different versions of the SciML HH model before settling on the one used in the pre-print publication, including a Markov scheme with 3 states connected in different ways, a single neural network that returned multiple $\tau_i(V)$ and another one that returned $m_{i,\infty}(V)$ and many more. The model I settled on was performing the best on the validation data. Due to the cost of training a single model we used the following approach:

1. Split off a smaller data set, taking 2 cells from each K_v type
2. Set up a **unified** model $\mathcal{M}$ from a number of different combinations of hyper-parameters
   a) number of gating particles $i \in [1, 2, 3]$
   b) number of layers $N_L \in [1, 2, 3]$
   c) number of hidden units $N_H \in [5, 10, 20]$
   d) activation function act $\in$ [tanh, σ, CeLU]
   e) regularization constant $\lambda \in [10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}]$
   f) regularization type: L_1 or L_2
   g) # of training epochs [100, 200, 300, 600]
3. Using conditional likelihood fit a model $\mathcal{M}$ and evaluate validation RMSE
4. Repeat setting up until options that make a significant different in training have been exhausted and select the best one

I opted to only do hyper-parameter optimization for the unified model reasoning that if an architecture would be powerful enough to fit a variety of different K_v types, it would also be powerful enough to fit individual K_v data since it is generally an easier task due to generally smaller variance within the data for individual channels. The final architecture I arrived at (comparison of different architectures not shown) was $i = 2$, $N_L = 2$, $N_H = 5$ with tanh nonlinearity using L_1 regularization with $\lambda = 10^{-4}$ and 300 training epochs.

Having settled on an architecture, I trained all the models using it. However, for some K_v types I had to use multi-start optimization, using multiple different seeds for the initial parameters and restarting optimization, because the individual K_v models suffered from bad local optima. The number of different seeds used ranged from 1 to 15. Having obtained models that trained properly, I investigated their results further by saving model parameters every 15 iterations and evaluating the validation loss at those points, selecting the set of parameters that showed the lowest validation error, essentially doing early stopping (see Fig 12).

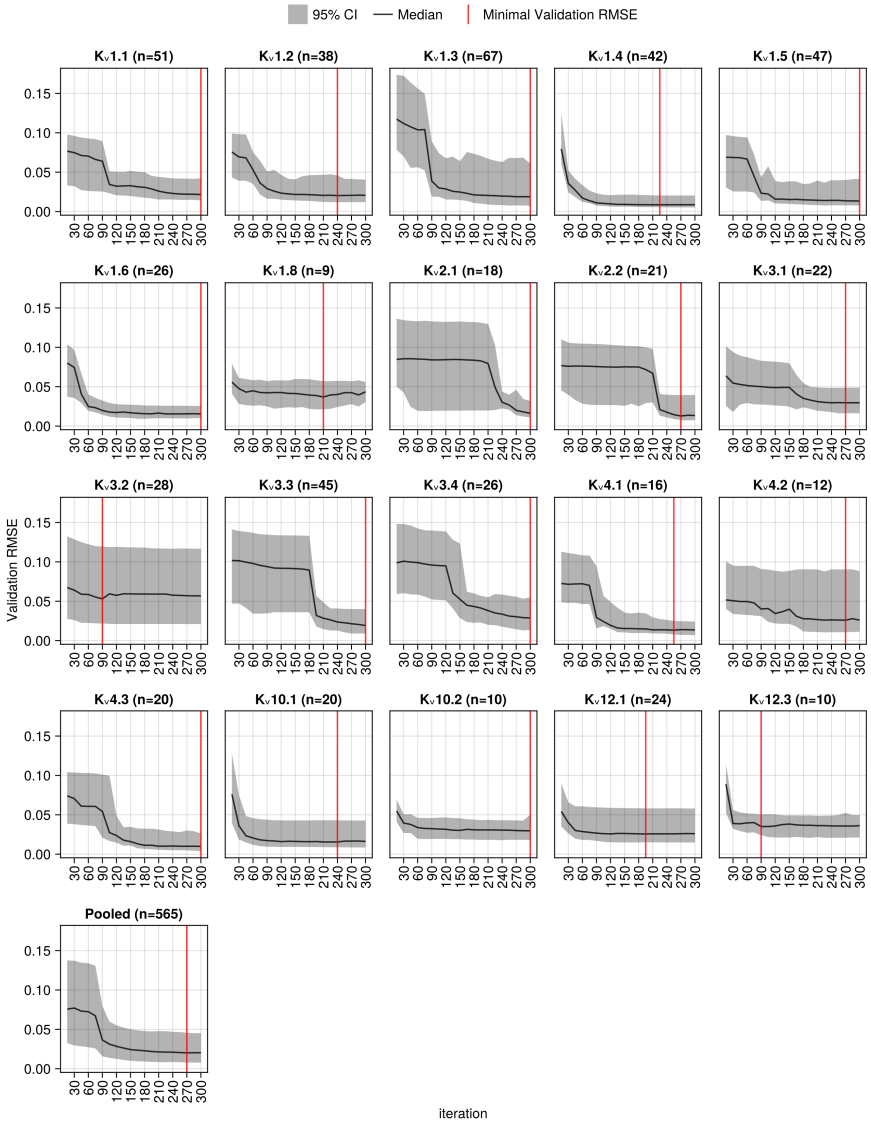


Figure 12: Early stopping evaluation of different individual and the unified K_V model. The red line denotes the iteration at which the validation RMSE was minimal, the black line is the median validation RMSE, the shaded area is the 95% confidence interval for the validation RMSE, n denotes the number of different cells in the validation data set.

CONCLUSIONS

This chapter dealt with the challenge of modelling chemical reaction networks when there is no data to constrain some of the states. In such instances it may be difficult to select between competing chemical reaction networks that reproduce the data similarly well. This concern is even more severe for voltage-gated ion channel models since each channel type requires its own chemical reaction network and there are many different channel types. I bypassed this challenge by using the HH formalism with neural networks that represented the equilibrium fraction and time constant functions with neural networks. This approach provided the necessary flexibility and expressive power required to fit the experimental recordings from Ranjan et al. (2019). In addition, the experimental data contained significant within-K_v type variability within a channel type, necessitating a NLME approach.

There are reasonable criticisms that could be made against the work done in this chapter. For example, it could be argued that the baselines used were not created to reproduce the channel recordings from Ranjan et al. (2019). While this criticism is fair, it reflects a shortcoming in the treatment of ion channel gating models. Similar to the different calmodulin schemes discussed in Chapter 3, there are often different gating models for the same ion channel type. These models are seldom compared and may be tuned to different sources of data. Even if the baseline models were not explicitly created to reproduce the Ranjan et al. (2019) data, I pre-screened a large number of K_v models from `https://modeldb.science/` that were nominally meant to capture currents of individual K_v types, using only those models that performed the best. Moreover, the `https://channelpedia.epfl.ch/` data is currently the richest source of ion channel data, making it an excellent benchmark data set for ion channel models.

This chapter provides a significant contribution for two reasons. Firstly, there are many exciting possible extensions directly following from the approach taken, e. g. extending the unified model to more K_v types, to different ion channel families, and to different gating modalities, allowing the combination of data from different species. However, the second and potentially more important contribution is the showcasing of the general DeepNLME approach to a task in computational neuroscience. The combination of domain expertise and DeepNLME is a very powerful tool that could be applied to many different challenges in computational neuroscience, especially where between subject variability and complex dynamics are involved, e. g. the heterogeneous spiking of stellate cells (Pastoll et al., 2020). In the next chapter I outline a research programme for one such challenge – the modelling of synaptic plasticity.

Part III

FUTURE DIRECTIONS

But thou, contracted to thine own bright eyes,
Feed'st thy light's flame with self-substantial fuel,
Making a famine where abundance lies,
Thyself thy foe, to thy sweet self too cruel.

Excerpt from "*Sonet 1*" by William Shakespeare, 1609

5

APPLICATION OF DEEPNLME IN SYNAPTIC PLASTICITY MODELLING

Following the successful fitting of different Ca^{2+}-CaM schemes using nonlinear mixed effects (NLME) modelling (Chapter 3) and the performant DeepNLME-based modelling of K_v channel gating (Chapter 4), in this chapter I discuss the application of DeepNLME to model synaptic plasticity. As mentioned in Chapter 2, due to the poorly constrained chemical reaction network and the partial observations of species, this challenge is the most complex discussed in this thesis. In the first section I provide a brief review of previous chemical reaction network-based modelling of synaptic plasticity, along with the outline of a chemical reaction network necessary for a specific form of long-term plasticity (LTP). I then, in the second section, build upon it by outlining a DeepNLME model that may be able to model this form of synaptic plasticity.

In section three I discuss ways of expanding the model presented in the second section to capture different forms of synaptic plasticity present in the Schaffer collateral synapse (Edelmann, Cepeda-Prado, and Leßmann, 2017). Finally, in the fourth section I further expand the model, building up to a model that may be able to capture different forms of synaptic plasticity in different types of synapses. Due to the nature of the DeepNLME modelling, the presented integration of the DeepNLME approach with the chemical reaction networks is not the only possible configuration. However, given the work I undertook in Chapter 3 and Chapter 4, the discussed integration seems to be the most promising. Models outlined in all three sections may be valuable tools in investigating synaptic plasticity, but, like all models, they would have some inherent limitations which are discussed in the final section of this chapter.

5.1 CHEMICAL REACTION NETWORKS IN SYNAPTIC PLASTICITY

There are many different forms of synaptic plasticity that depend on different parts of the synaptic chemical reaction network (Edelmann et al., 2015; Magee and Grienberger, 2020). I will focus on a particular synapse and a particular form of synaptic plasticity, which will provide the basis for a discussion of the application of SciML and NLME presented later in this chapter. I will present a brief overview of the real chemical reaction network of the Hippocampal CA1 pyramidal cell Schaffer collateral synapse and some of its models. Moreover, I will present an extension to the basic chemical reaction network the-

ory presented in Chapter 2 that can be used for simulating the synaptic chemical reaction network.

5.1.1 *CaMKII-dependent LTP induction and expression*

It has been empirically shown that three 1s trains of 50Hz stimulation applied to the CA3-CA1 fibres produce a robust, long lasting (on the order of 2h) and purely post-synaptic form of long-term potentiation (50Hz LTP) – an increase in the effectiveness of synaptic signalling (Bayazitov et al., 2007). LTP induction has been widely linked to intracellular calcium dynamics (Shouval, Bear, and Cooper, 2002). Regehr and Tank (1992) showed that 50Hz LTP is independent of L-type voltage-gated Ca^{2+} channels, leaving the N-methyl-D-aspartate type of glutamate receptors (NMDARs) as the primary candidate for the calcium entry necessary to induce 50Hz LTP.

There are many possible downstream targets for the Ca^{2+} entering the post-synaptic site. Along with intracellular calcium buffers (Brini and Carafoli, 2011), there is evidence that Calmodulin (CaM) and Calcium/Calmodulin-dependent Kinase II (CaMKII) are among the primary downstream targets activated by the 50Hz LTP induction protocol (Opazo et al., 2010). Moreover, once activated by Ca^{2+}-bound CaM, CaMKII can phosphorylate a considerable number of targets (Colbran, 2004). However, there is evidence that two CaMKII phosphorylation targets are the most relevant in the induction of synaptic plasticity: alpha-amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid glutamate receptors (AMPARs, Kristensen et al., 2011) and Stargazin (Sg, Heine et al., 2008). I will briefly describe both of these molecules.

AMPARs are the main transducers of electrical signals within synapses – upon binding glutamate they change their conformation and can conduct positively charged ions into the dendritic spine (Benke and Traynelis, 2019). AMPARs have four glutamate binding sites with four associated current conductance levels (Postlethwaite et al., 2007). Moreover, AMPARs have a large number of post-translational modification sites with Ser831 being of most relevance (Diering and Huganir, 2018). AMPAR Ser831 can be phosphorylated by CaMKII (Mao et al., 2014). When AMPAR Ser831 is phosphorylated, AMPARs have been shown to spend more time in higher conductance states, resulting in a larger inward current – a stronger synaptic signal (Kristensen et al., 2011). Therefore, AMPAR Ser831 phosphorylation provides one mechanism for 50Hz LTP induction.

In synapses *in vivo* AMPARs form macromolecular complexes with various auxiliary proteins, which have a wide range of functions (Payne, 2008). One of these proteins – Stargazin – plays a part in AMPAR trafficking between the extra-synaptic fraction, which does not get activated, and the synaptic fraction, which can be activated

(Sumioka, Yan, and Tomita, 2010). AMPARs bound to a phosphorylated Sg remain in the synaptic fraction for a longer period of time compared to unphosphorylated Sg, due to Sg binding the membrane-associated PSD-95 protein (Heine et al., 2008). It has been shown that Sg can be phosphorylated by CaMKII, increasing the number of AMPARs in the synaptic fraction (Opazo et al., 2010). Thus Sg phosphorylation is a second mechanism by which 50Hz LTP can be induced. The two pathways are different means to the same end – either increasing the current conducted by individual AMPARs, or increasing the number of AMPARs or some combination of the two would lead to a potentiated synapse.

Finally, phosphatase activity in the synapse plays a stabilizing role and dephosphorylates both AMPAR Ser831 and Sg. There is some uncertainty regarding which phosphatases play this role in a CA1 pyramidal cell synapses. However the most likely candidates are protein phosphatase 1 (PP1, Urakubo et al., 2014) and Calcineurin (CaN, Snyder et al., 2003). A simplified diagram of the early molecular events underlying 50Hz LTP is shown in Figure 13.

An additional study by Incontro et al. (2018) used a 90s 2Hz stimulation protocol of CA3-CA1 fibres with neurons held at 0mV to induce an NMDAR and CaMKII dependent form of synaptic plasticity which resulted in increased AMPAR postsynaptic currents (2Hz LTP). The authors used both pharmacological and genetic manipulations to show the importance of CaMKII. Therefore, there is strong evidence that LTP results shown in Incontro et al. (2018) and Bayazitov et al. (2007) operate (at least partially) through a common set of biochemical mechanisms, namely via the NMDAR-dependent Ca^{2+} influx, CaMKII-dependent phosphorylation and an increased AMPAR conductance which could be mediated by increasing the number of receptors via CaMKII-Sg phosphorylation dependent AMPAR trapping in the PSD or increasing single channel conductance due to CaMKII-AMPAR phosphorylation.

I have described (excluding the biochemistry of Ca^{2+} buffering) the minimal biochemical network necessary for 50Hz LTP and 2Hz LTP according to the currently available evidence. Having briefly described the empirical literature concerning a form of LTP, I will now summarise the published models of the CA3-CA1 synaptic chemical reaction network.

5.1.2 *Models of the synaptic chemical reaction networks*

The real chemical reaction networks present in synapses that underpin synaptic plasticity can, in principle, be modelled using the basic machinery defined in Chapter 2. However, they have a few distinguishing features that make it difficult to directly apply the basic theory. Firstly, the number of unique protein species in synapses is

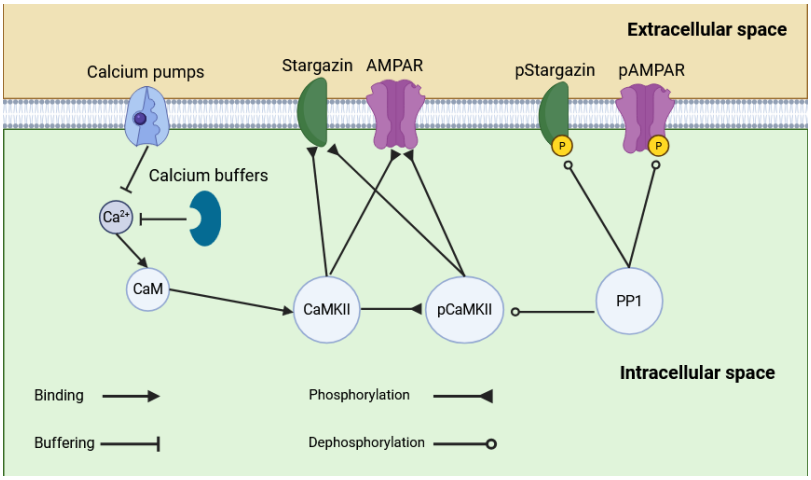


Figure 13: A simplified diagram of the chemical reaction network potentially underpinning 50Hz LTP (Bayazitov et al., 2007). Yellow circles indicate phosphorylation of Stargazin, CaMKII and AMPAR.

large, on the order of thousands (Roy et al., 2018; Sorokina et al., 2021). Moreover, although a significant amount of information on protein-protein interactions is available (Xenarios et al., 2002) in existing databases, the information currently available is unfortunately not sufficient to model the full chemical reaction networks present in synapses due to underconstrained reaction rate constants. Secondly, the numbers of biochemical species in synapses (whose volume is on the order of femtoliters), can be very small, on the order of tens of receptors in the membrane (Racca et al., 2000) or tens of free Ca^{2+} ions in the cytosol (Sabatini, Oertner, and Svoboda, 2002). While such low numbers of molecules can be modelled via mass-action kinetics, the obtained solution would be an average of potentially many stochastic simulations. Therefore, the mass-action solution may not be representative of the dynamics of individual simulations because the assumptions of the simulated solution being uniform and well-mixed do not hold. Despite these two challenges, there are published models of the synaptic chemical reaction network.

Current models of the synaptic chemical reaction networks contain at most 55 unique protein species (Heil et al., 2018). Therefore, compared to the size of the full synaptic proteome (on the order of thousands of unique protein species), existing works are only partial models of the full synaptic chemical reaction network. For example, the seminal work of Bhalla and Iyengar (1999) was the first study to include as many as 42 different biochemical species. In principle it included molecular species that would be sufficient to model a few different forms of LTP, but potentially not 50Hz/2Hz LTP as it does not include Sg (assuming it is necessary). Other work investigated the role of a CaM buffer and bistable biochemical switches participating in LTP (Hyun and Lisman, 2008; Zhabotinsky et al., 2006). Nair et

al. (2014) investigated the impact of neuromodulation by dopamine and acetylcholine on plasticity in the striatum. Sorokina, Sorokin, and Armstrong (2011) created a model with a very large number of unique proteins to investigate the size of the molecular aggregates in the synapse. Jedrzejewska-Szmek et al. (2017) investigated the impact of β-adrenergic neuromodulation on LTP. Kim et al. (2011) investigated the co-localization of molecular species participating in LTP. Bartol et al. (2015) used morphologically detailed dendritic spine information to produce one of the most detailed Ca^{2+} buffering models. De Schutter (2013) and Antunes and De Schutter (2012) investigated the importance of stochasticity in inducing LTD in cerebellar neurons. This is by no means an exhaustive list of the models of the synaptic plasticity chemical reaction network (for that see Heil et al., 2018). Rather, the models generally tend to include small, specific and overlapping parts of the synaptic chemical reaction network.

Real chemical reactions are discrete events, in particular, transitions between states of the same molecule and formations of bonds between different molecules. Mass-action kinetics provides a smooth approximation to these discrete events that is accurate when considered for a large number of molecules, resulting in the species-formation-rate function used in an ordinary differential equation (Definition 2.1.7). However, if the numbers of interacting species are low, as they can be in synapses, mass-action kinetics are not appropriate. Instead a stochastic solving approach that directly models individual reactions is necessary to avoid inaccuracies in mass-action solutions that may, for certain systems, qualitatively differ from a stochastic solution.

The simplest and most well-known algorithm used to simulate chemical reaction systems stochastically is Gillespie's method (Gillespie, 1977). Instead of simulating a continuous solution trajectory, Gillespie's algorithm simulates the occurrence of discrete events by drawing samples from the joint probability distribution (Gillespie, 2007)

$$p(\tau, j|\mathbf{x}, t) = a_j(\mathbf{x}) \exp(-a_0(\mathbf{x})\tau) \tag{48}$$

where τ is the time to the next discrete event, j denotes which of a set of unidirectional reactions occurs, $\mathbf{x}$ is the vector of current concentrations of species in the system, $a_j(\mathbf{x})$ is the propensity function of the jth reaction and $a_0(\mathbf{x}) = \sum_j a_j(\mathbf{x})$ is the total propensity of the system. The propensity function $a_j(\mathbf{x})$ depends on the jth reaction structure, for example for a first order reaction $A \xrightarrow{k_a} B$ the propensity function is $a_j(\mathbf{x}) = k_a A$, where A is the number of molecules of A. Propensities are generally equivalent to the velocities of reactions in the mass-action formulation. Note that the time to the next reaction is an exponential random variable and the probability of which reaction occurred depends on its propensity relative to the total propensity.

A Monte Carlo sample from Equation 48 provides a sample from the solution of the chemical master equation $p(\mathbf{x}, t|\mathbf{x}_0, t_0)$, which describes the probability of all combinations of molecules in the system over time. Due to the number of states and its complexity, analytical solutions to the chemical master equation are not generally tractable for anything but the simplest systems. Therefore, for systems where the mass-action assumptions do not hold, Gillespie's algorithm is an invaluable tool in simulating the dynamics of chemical reaction networks.

5.2 DEEPNLME MODELLING OF A SINGLE TYPE OF SYNAPTIC PLASTICITY IN A SINGLE TYPE OF SYNAPSE

Given the description of the chemical reaction network underlying 50Hz LTP in the previous section, an immediate question is whether the DeepNLME approach is necessary in the first place. Would it not be possible to assemble a model with the constituent species, tune some of its parameters as necessary and have a detailed model of 50Hz LTP? I have attempted this in a previous unpublished work (my MScR dissertation, Linkevicius, 2019) and it is not as straightforward as it might seem initially. Even though I was able to reproduce many different lower level phenomena, such as the size of the EPSP in the spine (Jayant et al., 2017) and spine Ca^{2+} buffering (Sabatini, Oertner, and Svoboda, 2002), by using realistic amounts of molecules and published models, the model did not reproduce 50Hz LTP. As outlined in Chapter 2 and as shown in 3, constraining the parameters of even intermediate sized chemical reaction networks is not a trivial task. One of the main issues with the model from Linkevicius (2019) was the lack of CaMKII activation and the difficulty of finding which parts of the model could be changed to increase it while preserving the rest of the model behaviour. This experience served as one of the main motivations for the approach used in this thesis, limiting the amount of guesswork necessary to produce complex models.

The 50Hz LTP DeepNLME model would contain both chemical reaction network parts and neural networks, acting as black box functions to glue together the constrained parts of the chemical reaction network. Given the empirical work on Ca^{2+} signalling and buffering in the Schaffer collaterals (Bloodgood and Sabatini, 2008; Sabatini, Oertner, and Svoboda, 2002; Scheuss et al., 2006), constraining a Ca^{2+} buffering model, along with assumptions about the size of the Ca^{2+} input resulting from the CA3-CA1 fibre stimulation, is feasible and has already been attempted (Bartol et al., 2015; Bell et al., 2019). The Ca^{2+}-CaM chemical reaction network would be need to be incorporated, potentially based on the work in Chapter 3.

The DeepNLME portion of the model would represent the Ca^{2+}-CaM-dependent CaMKII phosphorylation of AMPAR (Kristensen et

al., 2011) and Stargazin (Sg, Opazo et al., 2010). Instead of the simplified CaMKII autophosphorylation reactions used in many other modelling studies (Gallimore et al., 2018; Hayer and Bhalla, 2005; Jedrzejewska-Szmek et al., 2017), a neural network NN_{CaMKII} uses all of the Ca^{2+} bound CaM species, denoted as $[CaMN_nC_c]_{n,c\in[0,2]}$ (subscripts later omitted to reduce clutter) as inputs. NN_{CaMKII} would output a portion of the species-formation-rate function (Equation 6) that is independent of the substrate being phosphorylated. Moreover, NN_{CaMKII} would take random effects $\eta_i^{(CaMKII)}$ as inputs, allowing the neural network to be individualized to account for some of the between-synapse variability. The NN_{CaMKII}-based approach reduces the number of assumptions that have to be made, avoiding the poorly constrained parts of the chemical reaction network that are essential to the overall performance of the model.

Also, Ca^{2+}-CaM would activate CaN, using a full chemical reaction model (Quintana et al., 2005), which, along with PP1, would dephosphorylate AMPAR and Sg. The binding of Ca^{2+}-CaM-CaN and PP1 to AMPAR and Sg may require refinement if there are significant issues with the published reaction rates (Urakubo et al., 2008). Finally, if random effects are used on the total concentrations of the biochemical species, $NN_{CaMKII}^{(i)}$ (superscript i denotes the output dimension) may need to use the concentration of PP1 as an input, to be able to capture the CaMKII autophosphorylation dependence on PP1 (Ordyan et al., 2020).

The set of ODEs for the ith observed synapse (omitting the details where possible) is given in the boxed equations Equation 49–51 – each box represents one composable module. The readout of the model is based on AMPAR concentrations, linking their concentration and state to EPSP measurements (more on this below, Quintana et al., 2005). Here w is the multiplier modelling the effect of AMPAR phosphorylation. A visual representation of the model is shown in Figure 14.

$$
\begin{aligned}
X_{\text{CaMKII}} &= \left([\text{CaMN}_n\text{C}_c], [\text{PP1}], \eta_i^{(\text{CaMKII})}, L_1, L_2, L_3\right) \\
\frac{d[\text{Ca}^{2+}]}{dt} &= [\text{Ca}]^{2+} \text{ influx terms} + [\text{Ca}]^{2+} \text{ buffering terms} \\
&\quad + [\text{Ca}]^{2+}\text{-CaM interaction terms} \\
&= \text{ ... ODEs for the } [\text{Ca}]^{2+} \text{ buffers...} \\
\frac{d[\text{CaMN}_n\text{C}_c]}{dt} &= \text{ ... } [\text{Ca}]^{2+}\text{-[CaM] binding terms ...} \\
\frac{d[\text{CaMN}_n\text{C}_c\text{-CaN}]}{dt} &= \text{ ...} \\
\frac{d[\text{pSg}]}{dt} &= [\text{Sg}] \cdot \text{NN}^{(2)}_{\text{CaMKII}}(X_{\text{CaMKII}}) \\
&\quad - k_f[\text{pSg}][\text{CaM-CaN}] + \text{ ...} \\
&= \text{ ... ODEs for phosphatase activation...} \\
\frac{d[831\text{pAMPAR}]}{dt} &= [\text{AMPAR}] \cdot \text{NN}^{(1)}_{\text{CaMKII}}(X_{\text{CaMKII}}) + \text{ ...} \\
\text{EPSP} &\sim \mathcal{N}\left(\frac{\text{AMPAR}(t) + w \cdot 831\text{pAMPAR}(t)}{\text{AMPAR}_0 + 831\text{pAMPAR}_0}, \sigma\right)
\end{aligned}
\tag{49}
$$

$$
\begin{aligned}
X_{\text{PKA}} &= \left([\text{CaMN}_n\text{C}_c], [\text{AC}], [\text{cAMP}], \eta_i^{(\text{PKA})}, L_1, L_2, L_3\right) \\
&= \text{ ... GPCR activation terms ...} \\
\frac{d[\text{AC}]}{dt} &= \text{ ... CaM- and GPCR-dependent AC terms ...} \\
\frac{d[\text{cAMP}]}{dt} &= \text{ ... AC- and GPCR-dependent cAMP terms ...} \\
\frac{d[845\text{pAMPAR}]}{dt} &= [\text{AMPAR}] \cdot \text{NN}_{\text{PKA}}(X_{\text{PKA}}) + \text{ ...}
\end{aligned}
\tag{50}
$$

$$
\begin{aligned}
\frac{dL_1}{dt} &= \text{NN}^{(1)}_{\text{L}}([\text{CaMN}_n\text{C}_c], [\text{cAMP}], L_1, L_2, L_3, \eta^{\text{L}}_i) \\
\frac{dL_2}{dt} &= \text{NN}^{(2)}_{\text{L}}([\text{CaMN}_n\text{C}_c], [\text{cAMP}], L_1, L_2, L_3, \eta^{\text{L}}_i) \\
\frac{dL_3}{dt} &= \text{NN}^{(3)}_{\text{L}}([\text{CaMN}_n\text{C}_c], [\text{cAMP}], L_1, L_2, L_3, \eta^{\text{L}}_i)
\end{aligned}
\tag{51}
$$

Some of the modelling choices in Equation 49 require elaboration. It has been shown that partially bound Ca^{2+}-CaM may activate CaMKII (Shifman et al., 2006). Therefore, the neural network approach sidesteps the assumptions on how Ca^{2+}-CaM binds to CaMKII, the tuning of the added set of CaM-CaMKII binding rates (Meyer et al., 1992), the effect of CaM-CaMKII on binding Ca^{2+} (Gaertner, Putkey, and Waxham, 2004), the simplifications of the CaMKII dodecameric structure (Michalski and Loew, 2012) and the parametrisation of CaMKII binding to AMPAR/Sg. All of these steps require data that is either not available or not sufficient to properly constrain the parameters. Moreover, this approach also avoids the combinatorial explosion of

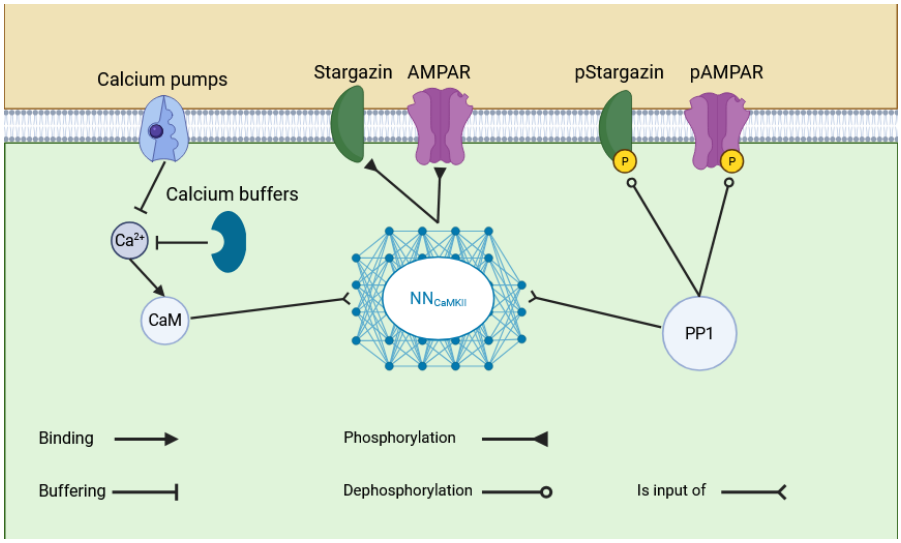


Figure 14: Visual representation of the model in Equation 49. Note that the symbol used for $\text{NN}_{\text{CaMKII}}$ is a simplification and does not represent the actual intended architecture.

the number of species in the ODE system arising from these complex interactions (Stefan et al., 2014).

As already mentioned, $\text{NN}_{\text{CaMKII}}$ takes all of the Ca^{2+} bound CaM species as inputs. Since CaMKII is not in the ODE system as a species which competes for Ca^{2+}-CaM, $\text{NN}_{\text{CaMKII}}$ must take all of the Ca^{2+}-bound CaM species as inputs, even those bound to CaN or other molecules, such as Neurogranin (Zhong et al., 2009; Zhong and Gerges, 2012). Using even the CaM species bound to other molecules is necessary to avoid underestimating the CaM levels that may have been bound to CaMKII if it were in the ODE system competing for it (Pharris, Patel, and Kinzer-Ursem, 2018). In order to prevent the explosion of the input dimensionality of $\text{NN}_{\text{CaMKII}}$, it may be sufficient to pass only the sums of CaM species bound to different amounts of Ca^{2+}. Finally, $\text{NN}_{\text{CaMKII}}$ takes random effects as inputs to account for any variability between synapses that affects the Ca^{2+}-CaM-CaMKII cascade, as well as any missing terms not present in the model chemical reaction system which may impact AMPAR/Sg phosphorylation.

The model outlined in Equation 49–Equation 51 posits that all other reactions, other than $\text{NN}_{\text{CaMKII}}$, do not exhibit variability as they do not have random effects on the reaction rates. The excitatory postsynaptic potentials (EPSPs) are most often the sole modality of observations in LTP induction experiments (Frey and Frey, 2009). Therefore, there are multiple ways in which the variability in EPSP recordings could be accounted for, offloading it to $\text{NN}_{\text{CaMKII}}$ being the primary candidate due to its flexibility. Offloading the variability to $\text{NN}_{\text{CaMKII}}$ would be unlikely to work when fitting multiple different types of observations consecutively, for example, if CaN activity is measured at the same time as LTP is induced. When such

measurements become available, random effects could be added to the relevant parts of the chemical reaction network to provide the necessary flexibility to fit the variability which is now informed by the additional measurements. However, until such measurements become available, offloading all of the between-synapse variability to $\mathrm{NN}_{\mathrm{CaMKII}}$ is the most practically feasible option.

The model in Equation 49 is loosely inspired by the pharmacometrics/pharmacodynamics models (Martensen et al., 2024), where certain entries of the ODE system relate to the drug trafficking between different compartments (parts of the body) and others to its effect on the observed quantities. Here Ca^{2+} is the drug and the $\mathrm{NN}_{\mathrm{CaMKII}}$ models the effect of the drug on the species that directly affect the observable quantities – EPSPs due to AMPAR opening. The final part of the model, shown in the last line of Equation 49, is the observation model which links the concentration of AMPARs to the normalized increases in EPSP strength and its dynamics over time (e. g. see Figure 4a or 8b in Bayazitov et al., 2007). Deriving such an observational model would require mechanistic assumptions of how AMPARs are related to the measured EPSP features (e. g. slope or amplitude). Alternatively, it could be assumed that the increases in the EPSP features is directly proportional to the number of AMPARs, making the observational model relatively easy to define. For example, it is possible to account for the increased conductance of phosphorylated AMPARs by taking their concentration and multiplying by 2 (Kristensen et al., 2011). This would produce multiple time-series of synapse specific normalized readings of synaptic strength, contrary to the pooled manner in which the data is generally presented in studies. Even though such an approach would be highly simplified, and further modelling refinements would be necessary, it may capture the essence of the EPSP dynamics in synaptic plasticity.

The data that would be necessary to train this model would be of the type in Bayazitov et al. (2007) Figure 8 – EPSP measurements over time. However, instead of pooling recordings from different cells, individual cell recordings should be used allowing the model to better disambiguate between different sources of variance. For example, different measured synapses might have somewhat different proteomic composition, leading to somewhat different plasticity outcomes (Roy et al., 2018). However, to the best of my knowledge, no such source of data is currently publicly available.

This concludes the outline of a DeepNLME model of the 50Hz LTP model for the Schaffer collateral synapse. The outline aims to integrate the parts of the synaptic chemical reaction network which are relatively well-described with the power of the machine learning in representing the parts which are currently too difficult to constrain. In the next section I will extend these idea to different forms of synap-

tic plasticity for the Schaffer collateral synapse (Edelmann, Cepeda-Prado, and Leßmann, 2017).

5.3 UNIFIED DEEPNLME MODELLING OF MULTIPLE DIFFERENT TYPES OF SYNAPTIC PLASTICITY IN A SINGLE TYPE OF SYNAPSE

Synapses are generally capable of displaying multiple forms of synaptic plasticity underpinned by different biochemical pathways which are activated by different modalities or types of stimuli (Edelmann, Cepeda-Prado, and Leßmann, 2017). Some modelling publications contain multiple different pathways which could result in synaptic plasticity (Gallimore et al., 2018; Hayer and Bhalla, 2005; Jedrzejewska-Szmek et al., 2017). For example, Jedrzejewska-Szmek et al. (2017) contains both CaMKII, which is activated by a Ca^{2+} influx, and protein kinase A (PKA), which is activated by cAMP released via a G protein coupled receptor, both phosphorylating AMPARs. Another example is Gallimore et al. (2018), whose model contains protein kinase C (PKC) and CaMKII, both of which can phosphorylate AMPARs, but are activated via different means. Moreover, both Jedrzejewska-Szmek et al. (2017) and Gallimore et al. (2018) contain complex chemical reaction networks that involve many different biochemical species providing multiple ways in which AMPARs can be dephosphorylated. Therefore, in order to model different forms of synaptic plasticity stemming from different, yet ill-constrained, parts of the chemical reaction network, the DeepNLME approach from the previous section needs to be generalized.

The most straightforward generalization is to simply have different neural networks representing the different pathways relevant to synaptic plasticity. For example, taking the pathways and dependencies from Jedrzejewska-Szmek et al. (2017), where either PKA (activated by cAMP and Ca^{2+}-CaM upstream) or CaMKII (activated by Ca^{2+}-CaM) can phosphorylate AMPARs (on different residues Ser845 and Ser831 respectively), a module extending the DeepNLME model (omitting most of the details that are not immediately relevant) is presented in Equation 50. A visual representation of this model is shown in Figure 15. Note that with modules Equation 49 and Equation 50 the model contains two different neural networks NN_{PKA} and NN_{CaMKII} which respectively model the input-output relationships of the respective species, based on the upstream species that activate them. Similar ideas could be extended to phosphatases whose activation is complex and/or under-constrained, or other species which integrate signals, for example DARPP-32 (Nair, Bhalla, and Kotaleski, 2016; Nair et al., 2014). Each neural network takes a set of random effects ($\eta_i^{(PKA)}$ and $\eta_i^{(CaMKII)}$), which capture the variability of the respective parts of the chemical reaction network. Using the cAMP concentration as the second input to NN_{PKA} has been chosen as a

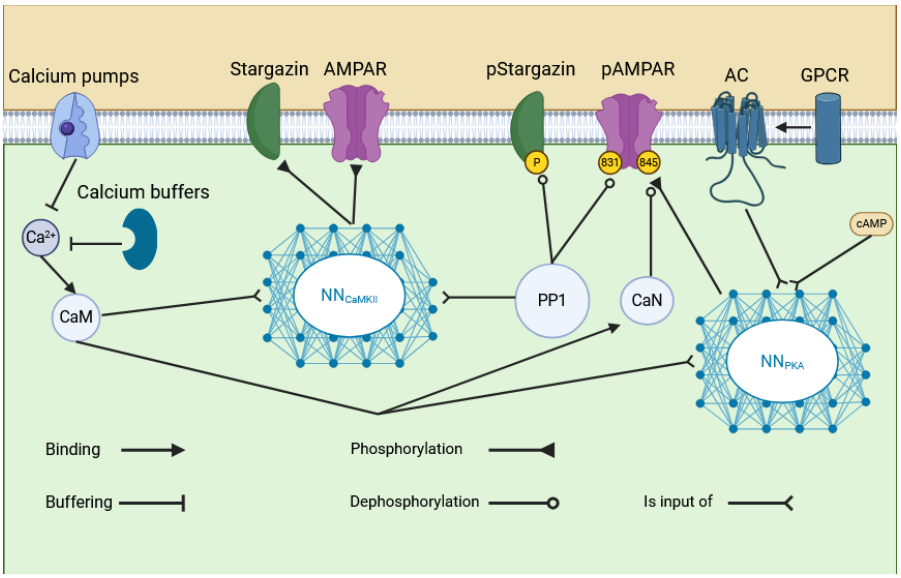


Figure 15: Visual representation of the model in Equation 49 and Equation 50 (without Equation 51). Note that the symbols used for NN_{CaMKII} and NN_{PKA} are a simplification and do not represent the actual intended architecture.

reasonable initial guess, assuming that the reactions describing its dynamics are constrained well-enough. If cAMP dynamics are not well constrained, it may be better to use more upstream species as inputs, whether it is [AC1] or G proteins. With increasing knowledge of the reaction rates and chemical reaction network structure, it would be possible to reduce the amount of variance captured by the neural networks in favour of the more interpretable chemical reactions.

It would be possible to replace both NN_{PKA} and NN_{CaMKII} by a single neural network NN_{Joint}, which would represent the combination of both pathways responsible for the activation of PKA and for the activation of CaMKII. Such an approach, while feasible, would require more data to train, as NN_{Joint} would have to be larger since it would represent a larger and more complex part of the chemical reaction network. Splitting it into two smaller networks NN_{PKA} and NN_{CaMKII} instead trades off domain knowledge coming from empirical studies against the flexibility offered by machine learning and reduces the amount of data necessary to train the model. However, the split structure, whether it is NN_{PKA} and NN_{CaMKII} or potentially more networks which would represent other pathways, relies on the assumption that their inputs are sufficient in explaining the variance in the observed data under given experimental conditions. For example, assume that Ca^{2+}-CaM-CaN was represented as a neural network NN_{CaN}. Now both NN_{CaN} and NN_{CaMKII} use Ca^{2+}-CaM as inputs, likely resulting in a more ill-constrained model that would suffer from non-identifiability issues, similar to trading off k_{on} and k_{off} in reaction rate constant fitting. More specifically, the same re-

sult may be achieved by either increasing the phoshporylation rate coming from NN_{CaMKII} or the dephosphorylation rate coming from NN_{CaN}. Therefore, using too many neural networks, especially if their inputs are the same or highly overlapping species, may result in DeepNLME models capable of fitting multiple forms of synaptic plasticity but incapable of generalizing well to different stimulation protocols.

One additional consideration is the potential feature of the synaptic chemical reaction network to hold information over time, e. g. in the form of bistable switches or other means (Miller et al., 2005). It may be that some parts of the chemical network which would be abstracted via neural networks maintain a form of memory via bistability or multistability. This bistability could not be captured within the neural network as it does not have states. However, it is possible to expand the ODE model with latent dimensions which would hold state information about the species not explicitly included in the chemical reaction network. They could simply be modelled via neural ODEs, using a neural network to model their derivatives (Chen et al., 2018). Building on top of the previous two modules, an additional neural ODE module can be included (see Equation 51).

The model with latent dimensions L_k ($k \in [1, 2, 3]$ in Equation 51) would now be able of maintaining temporal information about the species that are not explicitly included via the latent dimensions whose derivatives are modelled using NN_L. The architecture of NN_L would be relatively less constrained compared to NN_{CaMKII}, e. g.the outputs of NN_L would have to be capable of taking positive and negative values (unlike NN_{CaMII} or NN_{PKA}). Again note that NN_L takes random effects η_i^L in order to be able to account for some of the variability between the different recorded synapses. Moreover, it is necessary to link the rest of the chemical reaction network to NN_L by passing certain biochemical species as inputs, selecting those that are most likely to be informative based on domain knowledge; Ca^{2+}-CaM is taken as an example due to its multiple binding partners (Pharris, Patel, and Kinzer-Ursem, 2018). Moreover, any neural networks representing parts of the chemical reaction network (e. g. NN_{CaMKII} or NN_{PKA}) would have to take the latent states L_i as inputs. Such a model would contain a complete the information flow loop between the chemical reaction network into the latent dimensions and back from the latent dimensions into the chemical reaction network. Whether or not this extension is necessary depends on the specific type of synaptic plasticity being modelled and the complexity of the necessary chemical reaction network.

Concluding, in this section I provided an outline of a DeepNLME model that may be able to model different forms of synaptic plasticity at the same type of a synapse. I used the basic approach outlined in the Section 5.2, where parts of the chemical reaction network that are under-constrained are replaced by neural networks which are then fit

to data. The expanded model uses different neural networks to represent different parts of the chemical reaction network, using concentrations of different species as inputs whenever possible. Moreover it may include neural ODEs if latent dimensions are necessary for maintaining a latent representation about the species not included explicitly. In the next section I outline how the model from this section may be expanded to model synaptic plasticity phenomena from different synapse types, resulting in a unified DeepNLME model of synaptic plasticity.

5.4 UNIFIED DEEPNLME MODELLING OF MULTIPLE DIFFERENT TYPES OF SYNAPTIC PLASTICITY IN DIFFERENT TYPES OF SYNAPSES

In order to extend the model outlined in Equation 49–Equation 51 to different synapse types, a few assumptions about its structure must be made. I will elaborate on these assumptions by contrasting the unified synaptic plasticity model with the unified K_v model from Chapter 4. Due to making fewer assumptions, the unified K_v model is relatively more black box compared to the DeepNLME synaptic plasticity model outlined in the previous section. Due to its more abstract nature and flexibility, the K_v model should be able to capture the gating dynamics of different channel types, compared to the more concrete mechanisms assumed in the DeepNLME synaptic plasticity model. Therefore, any concrete reactions included in the unified DeepNLME synaptic plasticity model would either have to be present in all of the synapses modelled, or reactions not present in some synapse types would have to be switched off on a case-by-case basis, e. g. setting the concentrations of relevant species to 0. This is consideration is backed up by empirical research showing the diversity of the synaptic proteomes (Roy et al., 2018).

Recall that in the unified K_v gating model the K_v channel type was linked to the random effects via the neural network NN_{aug}. The random effect augmentation approach would have to be re-thought for the unified synaptic DeepNLME model. Synapses are much more complex and diverse objects than K_v channels (Grant and Fransén, 2020) and the information encoded in a one-hot vector (e. g. whether the synapse is a Schaffer collateral or a mossy fiber synapse) is likely to contain relatively little information that could inform the random effects. Instead, to better inform the random effects, more informative covariates Z_i, such as the the protein-protein interaction graphs of individual synapses, would have to be used as an input to NN_{aug}. More complex covariates would make NN_{aug} larger than used in the unified K_v model, making joint fitting no longer feasible requiring a sequential approach instead. Firstly, a unified model would have to be fitted to a pooled data set, obtaining a set of random effects

which would serve as the regression targets for NN_{aug}. Then, after the NN_{aug} is fitted to predict the random effects, the random effect prior Ω would have to be re-fitted, to account for the part of the variance in the data now explained by the covariates Z_i processed by NN_{aug}.

Therefore, the main difference compared with the DeepNLME synaptic plasticity model in the previous section would be the modelling of the random effects. Specifically, keeping the same model in Equation 49–Equation 51, the random effects $\eta_i^{(PKA)}$, $\eta_i^{(CaKMKII)}$ and $\eta_i^{(L)}$ would be replaced by the augmented random effects $\eta^\dagger$, defined in Equation 52, which are the sum of the output of the augmentation neural network $NN_{aug}(Z_i)$, where Z_i are the covariates for a given synapse i, and the random effects η_i:

$$\eta^\dagger = \begin{pmatrix} \eta_i^{(PKA)} \\ \eta_i^{(CaKMKII)} \\ \vdots \\ \eta_i^{(L)} \end{pmatrix} + NN_{aug}(Z_i) \tag{52}$$

Even though the step from the model of a single synapse capturing multiple forms of synaptic plasticity to a unified DeepNLME of all synapses is conceptually seemingly small, it would most likely pose significant practical difficulties, depending on the intended scope of the model. For example, there is nothing preventing a modeller from building a cross-species unified model of synaptic plasticity. The likely hurdles are practical, such as the unification of different experiments into a format where their data could be used in fitting a single model. The amount of data necessary to fit such a model would most likely be beyond the currently available data sets. Moreover, a cross-species DeepNLME model of synaptic plasticity could use many different types of covariates Z_i as inputs into NN_{aug} as there are many different factors that affect synaptic plasticity, such as species (Beed et al., 2020), age of the animal (Detoledo-Morrell, Geinisman, and Morrell, 1988), type of synapse (Larsen and Sjöström, 2015), type of experimental preparation (Glasgow et al., 2019) and even animal housing conditions (Bayat et al., 2015). All of these factors could be included as the covariates Z_i, necessitating a careful validation protocol to avoid over-fitting.

Similarly to the model outlined in the previous sections, EPSP measurements over time would be necessary to fit the model outlined here. More specifically, EPSP measurements recorded under induction protocols that include the relevant biochemical cascades (Edelmann, Cepeda-Prado, and Leßmann, 2017) would be necessary to properly fit the different neural networks. Again, instead of pooling recordings from different cells, individual cell recordings should be used. However, as in the previous case, it is impossible to extract such

data from publications and there are no public databases that contain such data.

This concludes the outline for a general unified DeepNLME model for synaptic plasticity. The outline did not provide all of the specific details for the model, as any model development is an iterative procedure, with challenges arising during model development. However, the basic idea was in part validated on voltage-gated K_v channels. A unified DeepNLME model would be a valuable tool furthering the understanding synaptic plasticity, uncovering points of commonality and difference between its different forms at the level of biochemical species, would provide predictions at the molecular level. Finally, it would serve as a repository of knowledge that could be incrementally refined and fine-tuned with increasing amounts of knowledge, eventually replacing the machine learning parts by chemical reactions or other mechanistic models. Even though it is not possible to predict all of the difficulties and limitations that would occur if the outlined model would be implemented, some limitations are clear at the outset, I discuss them in the next section.

5.5 CHALLENGES AND LIMITATIONS

Some of the main predictable difficulties and limitations of developing a unified DeepNLME model of synaptic plasticity are data related issues, incorrect model structure/reaction rates and lack of interpretability. Some of these challenges and limitations are a feature of the DeepNLME, rather than an issue to be resolved. Being aware of them could inform the model development process and preempt misallocation of time, resources and effort.

There are many factors affecting the induction and maintenance of synaptic plasticity (Bayat et al., 2015; Beed et al., 2020; Detoledo-Morrell, Geinisman, and Morrell, 1988; Glasgow et al., 2019; Larsen and Sjöström, 2015). To the best of my knowledge, no single source of data would have sufficient amount of details on all of the factors affecting synaptic plasticity for an extensive modelling of covariates Z_i, linking them to the random effects. Instead, different disparate sources of data would have to be combined, requiring unification of different sources of data which may be a non-trivial task. For example, very few, if any, synaptic plasticity studies would have both experiments on the induction of synaptic plasticity, as well as the proteome of the synapse being investigated. Moreover, there are some disparities in different studies on the synaptic protein-protein interactions (Sorokina et al., 2021). Disentangling which differences in the proteome are due to different experimental protocols, species, brain regions and other factors, is a complex task in itself, so it is unclear how much predictive signal on random effect it would provide. In some instances, if enough data for a particular covariate is deemed

to be inconsistent and contradictory, it may be best to leave it out, rather than try fitting a machine learning model to it. As long as the outputs of the augmentation neural network do not provide contradictory information, the portion of variance that is not captured by the covariates should be captured by the random effects. Even if there is a limited quantity of data on the covariates differentiating the synapses, it should not impede the initial fitting. However, absence of covariates differentiating synapses would prevent the model from making population level predictions.

Another potential challenge in developing the unified DeepNLME model is the usage of explicit reaction structures that are incorrect to the degree of precluding the overall model from reaching states that would be necessary for it to accurately fit the synaptic plasticity recordings. It is self-evident that having correct structural information facilitates model training, but there is a lack of publications investigating how incorporating incorrect elements affects model fitting. It is possible that addition of reactions that are incorrect, but only to a small degree, does not impede the training of SciML models too much, so long as the overall model is expressive enough. For example, if an incorrect Ca^{2+}-CaM model is used, but the difference between the true dynamics and the model is some constant, then NN_{CaMII} may be able to adjust to that without too many difficulties. This challenge should be diagnosable by using domain knowledge and inspecting the elements of the model participating in a specific type of synaptic plasticity, as well as validation under both control and experimental conditions. For example, if there is no Ca^{2+}-CaM activity in a set of simulations where CaMKII has been shown to be necessary for synaptic plasticity, the prime suspects would the Ca^{2+}-CaM model itself, along with overly stringent Ca^{2+} buffering, and overly strong phosphatase activity. Even though it may be possible to diagnose, the solutions to such challenges would be situation-specific and it would be difficult to prescribe any type of a remedy at the outset.

Finally, the third limitation that is guaranteed to be a feature of a unified DeepNLME model of synaptic plasticity is its dubious interpretability. Random effects that would be necessary to fit such a model are inherently uninterpretable by themselves as they capture the within-subject variability in some latent space. However, it is possible to link covariates to the random effects with additional modelling, providing some link between them. The selection of which covariates to use in order to improve a NLME model predictions is still an open question (Bonate, 2017) that is made more difficult by the usage of black box machine learning techniques. Therefore, even though a certain set of covariates could be used, it would be difficult to precisely determine which covariates are informative and beneficial in augmenting the random effects. Similarly, for the machine learning components used in the ODE system, since they are likely

to have inputs that are not one dimensional, interpreting the learned components may be difficult. Note that this problem is not present in Chapter 4 because the inputs into the neural networks are low dimensional – voltage and random effects, making the trained models easy to explore. Furthermore, if multiple machine learning components are used, interpreting the individual components may be misleading due to the inevitable sloppiness of the model, similar to Gutenkunst et al. (2007). More specifically, different neural networks may have opposite effects on some downstream effectors, e. g. phosphorylation and dephosphorylation of AMPARs, which may both be increased or decreased keeping the same steady-state activity. Therefore, while the unified DeepNLME model of synaptic plasticity would contain a lot of mechanistic details, it would have to be interpreted as a whole and with caution.

Undoubtedly, these three challenges and limitations are not the only ones that would arise in the process of constructing a unified DeepNLME model of synaptic plasticity. However, the DeepNLME framework offers solutions to more fundamental problems present in the current attempts to model synaptic plasticity, such as uncertainty in model structure and reaction rates. However, machine learning and replacing poorly constrained parts of the chemical reaction network with neural networks is not a silver bullet. With enough parameters it is possible to fit essentially anything (Hastie, Tibshirani, and Friedman, 2009). Therefore, machine learning approaches should be used sparingly and as minimally as possible, accompanied by a systematic validation protocol to guard against over-fitting. The practical challenges outlined in this section can be overcome by employing a principled validation procedure when training the model, as well as keeping in mind the limitations of the machine learning approach, whereas the more fundamental challenges in the current approaches are much more difficult to overcome. Therefore, applying the DeepNLME methodology would offer a step forward in understanding of synaptic plasticity.

5.6 CONCLUSIONS

Construction of a unified model of synaptic plasticity was the initial goal of my PhD studies. However, when I set out, I had neither the tools nor the data to do it. This chapter outlines an approach to a unified model of synaptic plasticity employing SciML and the NLME as essential tools. The outlined model circumvents a number of significant challenges which make the construction of a unified model very difficult: uncertainty in reaction rates, reaction structures and which biochemical species need to be included. Even though the usage of SciML and NLME introduces a different set of challenges, some of them may be overcome by approaches that are the standard

practice in machine learning. A unified model of synaptic plasticity would be a valuable tool capable of pushing the boundaries of understanding further, potentially providing some answers about how the brain learns and forms memories that have puzzled researchers for decades.

Part IV

DISCUSSION AND CONCLUSIONS

O lieb', solang du lieben kannst!
O lieb', solang du lieben magst!
Die Stunde kommt, die Stunde kommt,
Wo du an Gräbern stehst und klagst!

Excerpt from "*O lieb, so lang du lieben kannst*" by Ferdinand Freiligrath, 1829

6 DISCUSSION

The integration of scientific machine learning (SciML), nonlinear mixed effects (NLME) modelling and chemical reaction networks in neuroscience offers many interesting possibilities. As shown in Chapter 3 and Chapter 4, the usage of SciML and NLME lead to significant improvements on two challenges in computational neuroscience – modelling the dynamics of Ca^{2+}-Calmodulin binding and the modelling of K_v channel gating. Moreover, SciML and NLME show promise in allowing the construction of a unified model of synaptic plasticity, capable of modelling many different forms of it in many different synapses Chapter 5. Since each chapter contains its own discussion, in this chapter I will focus on possible future directions that could extend the work done in each chapter and the wider implications for the modelling of chemical reaction networks in computational neuroscience.

6.1 CALMODULIN MODELING

The main contribution of the Ca^{2+}-CaM binding modelling done in Chapter 3 is the comparison and evaluation of different schemes used throughout literature. The usage of many different schemes and parameter sets that are not compared between one another makes it difficult to compare models in which the schemes are used, creating more possible explanations of differences in model performance and potential reasons why a model may fail in reproducing some empirical data. The work in Chapter 3 is helpful in eliminating one of these possible reasons. Moreover, fitting the reaction rate constants of different schemes to the data from Faas et al. (2011) and Shifman et al. (2006) resulted in significantly better parameter sets for a number of more simplified schemes. However, inspecting the reaction rate constant fits across the different seeds shows that the parameters were still not well-constrained, at times spanning 5-10 orders of magnitude, showcasing the sloppy parameter sensitivities observed in many systems biology models (Gutenkunst et al., 2007). Therefore, future work will have to include even more data and constraints when fitting Ca^{2+}-CaM reaction rate constants to obtain higher precision in estimating these parameters.

There are many possible ways to derive additional constraints for the Ca^{2+}-CaM reaction rate constants. One of the most important factors to include in future work would be differential constraints on the C- and the N-lobe parameters, which would break the symme-

try of the model. Without such constraints the model is symmetric, which could result in model optimization being more difficult due to higher levels of sloppiness, more permitted parameter configurations (Gutenkunst et al., 2007). Both Faas et al. (2011, Table 1) and Shifman et al. (2006, Figure 3D) present information about CaM mutants, which are incapable of binding Ca^{2+} on either C- or the N-lobe. However, neither source of data was included in Chapter 3 because the data seems to be conflicting with some assumptions underpinning the Ca^{2+}-CaM schemes. Both studies show that when a single lobe is mutated to be incapable of binding, the binding equilibrium curves and parameters are very similar in their dissociation constants. However, the consensus in the literature is that C-lobe binds Ca^{2+} with higher affinity than the N-lobe in wild-type CaM (Lai et al., 2015; Linse, Helmersson, and Forsen, 1991; Porumb, 1994). Therefore, it is unclear whether the CaM mutants used in Faas et al. (2011) and Shifman et al., 2006 or any other study are representative of the wild-type CaM in terms of binding affinities. It has been shown that CaM mutations can affect the binding of CaM to its targets (Piazza et al., 2017), lending indirect support to this hypothesis. Therefore, alternative ways of constraining the Ca^{2+}-CaM reaction rate constants need to be considered.

Another initially appealing possibility to constrain the parameters of Ca^{2+}-CaM binding is by adding additional molecules that bind Ca^{2+}-CaM, such as CaMKII, Neurogranin, Calcineurin, etc. (Gaertner, Putkey, and Waxham, 2004; Quintana et al., 2005). For example, measurements of Ca^{2+} released from CaM after addition of Neurogranin or CaMKII (Gaertner, Putkey, and Waxham, 2004) do contain some information about the reaction rate constants of Ca^{2+}-CaM binding. However, the inclusion of additional molecules necessitates the addition of binding parameters between Ca^{2+}-CaM to them, as well as potentially a different set of Ca^{2+}-CaM binding reaction rate constants since CaM binding to its targets has been shown to affect its Ca^{2+} affinity (Gaertner, Putkey, and Waxham, 2004; Lai et al., 2015). Therefore, any additional information gained from including more molecular species that allow more types of measurements would be counteracted by the additional set of reaction rate constants that would need to be fitted as well, making the overall approach counterproductive for constraining Ca^{2+}-CaM reaction rate constants.

A potential modification of the idea of using the information obtained from the inclusion of additional molecules further leverages the NLME approach. Instead of fitting the additional reaction rate constants stemming from the Ca^{2+}-CaM binding to its targets and their effects on Ca^{2+}-CaM affinity, it may be possible to marginalize over them by treating them as random effects and fitting only the basic Ca^{2+}-CaM rates as fixed effects. This would focus on fitting the basic Ca^{2+}-CaM rates from all of the other parameters, while leverag-

ing the information obtained from inclusion of additional molecules. However, there may be issues due to the high dimensionality of the resulting random effect space, permitting overfitting via random effects, requiring priors on the random effects that somewhat constrain the random effects modelling the non-basic Ca^{2+}-CaM interactions.

Yet another different way to constrain the Ca^{2+}-CaM binding rates is inclusion of additional biophysical constraints that reduce the number of free parameters. For example, microscopic reversibility constrains the dissociation constants in a cyclical reaction scheme (Colquhoun et al., 2004). It may be possible to use other biophysical laws or results based on detailed CaM investigations to produce further restrictions, reducing the number of free parameters (Malmendal et al., 1999).

Creation of chemical reaction networks with well-constrained parameters ties into the modelling of synaptic plasticity discussed in Chapter 5. The detailed modelling of chemical reaction networks approaches the modelling of synaptic plasticity from the bottom-up – assembly of a large chemical reaction network from the smaller component networks. In contrast, Chapter 5 approaches the modelling of the chemical reaction networks underpinning synaptic plasticity more from the top-down perspective – focusing on fitting the overall output of a system including only those parts of the synaptic chemical reaction network that are well-validated and constrained. The interplay between these two approaches is important and beneficial. The top-down approach can reveal which parts of the large chemical reaction network are under-constrained and would benefit the overall model most if investigated deeper. The bottom-up approach deals with the constraining and the fitting of the parts of the synaptic chemical reaction network revealed via the top-down approach. The eventual meeting of these two approaches at the top would result in a detailed chemical reaction network capable of accurately reproducing various different forms of synaptic plasticity while providing mechanistic details and predictions. Therefore, further work on the components of the synaptic chemical reaction network, aided by NLME, may be important in the future.

6.2 ION CHANNEL MODELLING

The work in Chapter 4 presents a significant improvement in the modelling of ion channel gating. Specifically, it utilizes the SciML and the NLME approaches to produce a unified model capable of modelling the gating of 20 different K_v channels, outperforming other K_v models from the literature. The unified model uses the basic formalism from Hodgkin and Huxley (1952), while using neural networks for the time constant and equilibrium fraction functions. Moreover, by passing random effects into the neural networks the unified

model is capable of capturing the within-K_v type variability, as well as between-K_v type variability. Moreover, the channel belonging to its family was encoded as a one-hot vector which was passed through a neural network NN_{aug} and mapped onto the random effects, allowing to extract models of individual channel types from the unified model. The unified model presented in Chapter 4 serves as a fruitful starting point with many exciting directions in which it could be extendend and generalized.

First of all, structurally the model is not limited to K_v channels and it would require only small modifications to extended it to other channel types, such as Na_v or Ca_v. Notably, `channelpedia.epfl.ch`, in addition to the K_v data, contains experimental recordings for 13 K_{ir}, four HCN and eight Na_v channel types. Similarly to K_v, not all of the channels may have enough data to fit them, but there is likely going to be enough data to establish the feasibility of the unified modelling approach. Therefore, this future direction is immediately obvious and extends the unified model in a fruitful direction.

Secondly, another clear future extension to the basic unified model is the modification of NN_{aug} to take richer inputs. More specifically, the structural channel information derived from its sequence, e. g. via AlphaFold (Jumper et al., 2021), and reduced to a smaller latent space, to avoid needing a large network, could be passed into NN_{aug}. Structural channel information may be one of the most informative covariates that can be known *apriori*. There are many auxiliary subunits and post-translational modifications which modify channel gating dynamics (Haworth and Brackenbury, 2019). However, since these can be dynamic, and change in response to external stimuli, modelling them is complex. Nevertheless, improving the random effect augmentation based on detailed covariates is likely to improve model fitting as it would rely less on explaining the variance via random effects, avoiding the pitfalls such as random effect optimization failures, multimodality, and inaccuracies stemming from posterior mismatch from marginal likelihood approximation assumptions.

One important limitation and area for future development is construction of a unified voltage-gated ion channel model that uses more general Markov schemes, rather than the simplified ones used by Hodgkin and Huxley (1952). There are limitations in expressiveness when using the Hodgkin and Huxley (1952) type of chemical reaction networks, which may make fitting more complex ion channels unfeasible. General Markov schemes do not have this limitation, but they bring a different set of practical challenges. General Markov schemes for ion channel gating can have a varying number of states, requiring an equivalent number of variables in the ODE system. It is possible to limit the total number of states, as done in Oikonomou et al. (2024), but this may suffer from the same issues of expressiveness as the

schemes of the Hodgkin and Huxley (1952) type. Therefore, a much more complex approach may be necessary.

The theoretical analysis in Fredkin and Rice (1985) (used in Oikonomou et al., 2024) points out that under certain conditions it is possible to determine the structure of a Markov scheme from its dwell-time histogram for the open and closed states. Therefore, it may be possible to use a flexible representation of a distribution (for example a normalizing flow, Kobyzev, Prince, and Brubaker, 2019) to represent the dwell-time histograms of different types of voltage-gated channels. Then, inspired by Oikonomou et al. (2024), the histogram of a certain type of ion channel could be passed as an input to a neural ODE foundation model that would model the dynamics of Markov schemes in a latent space, avoiding the issue of varying dimensionality between different Markov schemes. Finally, the time series of the solution in the latent space would have to be passed through a decoding model that would link the latent states to an aggregate open state that would be conducting current. In such a setup the normalizing flow would also be taking random effects as inputs to account for the within-channel type variability (Ranjan et al., 2019). Specific parts of such a model would have to be trained separately, especially the foundation model of the latent dynamics, as training them all jointly in an NLME fitting routine would most likely be too computationally expensive.

A model of this type would be much more flexible and able to model gating dynamics not possible via Hodgkin-Huxley type of gating. However, it most likely would not be as interpretable as the model presented in Chapter 4. It is clear from the results of Oikonomou et al. (2024) that even a deep learning approach may not always be able to distinguish between different Markov scheme structures due to observational noise. Therefore, even if a normalizing flow could represent the dwell-time histograms associated with various different Markov schemes, the schemes may not be identifiable due to experimental noise. Moreover, abstraction obtained via a foundation model trades off expressiveness for interpretability, and even if a Markov scheme could be extracted from the dwell-time histogram, it may be complex to extract the specific voltage-dependent functions associated with the transitions between different states. Therefore, even though an extension of the unified model in this direction is possible, it is uncertain how beneficial it would be, necessitating further research in this area.

6.3 GENERAL LIMITATIONS

Above and beyond the limitations outlined in each individual chapter, the approaches of NLME and SciML have some more general limitations. The most significant current limitation, especially relevant

when considering the simulation of the chemical reaction networks of synapses, is the difficulty of using stochastic simulation approaches. As mentioned in Chapter 2, versions if the Gillespie stochastic simulation algorithm (Gillespie, 2007) are commonly used when simulating synaptic chemical reaction networks due to the small counts of molecules of some species. However, stochastic simulation makes it difficult to use gradient-based optimization techniques that are suitable for fitting SciML models. Therefore, even though it would be possible to use a fitted SciML/DeepNLME model in a stochastic simulator, fitting one in a stochastic setting is not feasible. In principle Markov Chain Monte Carlo (MCMC) approaches could be used to obtain samples of neural network weights from the posterior distribution. Unfortunately, an MCMC an approach is prohibitively expensive due to the usual size of the neural network parameter vectors. Therefore, further work would be necessary to make fitting of DeepNLME models in a stochastic setting possible.

7

CONCLUSIONS

The work done in this thesis presents the results of two applications of nonlinear mixed effects (NLME) modelling and scientific machine learning (SciML) to chemical reaction networks. Moreover, it discusses future projects that may have been difficult to consider as feasible 10 or 15 years ago. Computational neuroscience work has generally struggled in dealing with the heterogeneity of the various neural subjects of investigation, whether its neurons, synapses, ion channels, etc. The work done in this thesis presents a number of ideas on how this heterogeneity could be handled, while not losing the wood, a typical exemplar of the system, for the trees, the individual recordings. Wider systematic investigations of heterogeneous neural systems using the methods of SciML and NLME are the logical next step following the work done in this thesis. The methods applied to the chemical reaction networks in this thesis can, for example be easily be extended to models of whole neurons using the data of Pastoll et al. (2020). Undoubtedly there are many more potential usages of this methodology.

The philosopher of science, Thomas Kuhn, proposed that science does not progress linearly, but undergoes periodic revolutions called paradigm shifts (Kuhn, 1962). For example, the work of Hodgkin and Huxley (1952) was to a significant extent responsible for a paradigm shift in the modelling of ion channel gating. The development of NLME modelling resulted in another paradigm shift in how pharmacological interventions were investigated, same for the modern machine learning techniques that now pervade our lives. Even though it is still too soon to tell, the advent SciML may usher in another paradigm shift in the scientific computational modelling. Many fields that use computational techniques, such as computational biology, computational neuroscience, etc., may enter Kuhn's normal science stage, where scientists utilize the techniques of the new paradigm to produce new knowledge and expand the scope of scientific understanding. The work done in this thesis is a part of that new normal science, trying to spread the ideas of the new paradigm via hitherto unfeasibly complex use cases and interesting examples. It is exciting to consider what new insights about the brain may be discovered in this new paradigm.

BIBLIOGRAPHY

Abril-Pla, Oriol et al. (2023). "PyMC: a modern, and comprehensive probabilistic programming framework in Python." In: *PeerJ Computer Science* 9. ISSN: 23765992. DOI: 10.7717/peerj-cs.1516.

Adéoti, Olaiya Mathilde, Schadrac Agbla, Aliou Diop, and Romain Glèlè Kakaï (2025). "Nonlinear mixed models and related approaches in infectious disease modeling: A systematic and critical review." In: *Infectious Disease Modelling* 10.1, pp. 110–128. ISSN: 2468-0427. DOI: https://doi.org/10.1016/j.idm.2024.09.001. URL: https://www.sciencedirect.com/science/article/pii/S2468042724001003.

Ahern, Christopher A., Jian Payandeh, Frank Bosmans, and Baron Chanda (2015). "The hitchhiker's guide to the voltage-gated sodium channel galaxy." In: *Journal of General Physiology* 147.1, pp. 1–24. ISSN: 0022-1295. DOI: 10.1085/jgp.201511492. eprint: https://rupress.org/jgp/article-pdf/147/1/1/1795314/jgp_201511492.pdf. URL: https://doi.org/10.1085/jgp.201511492.

Alexander, SPH, A Mathie, and JA Peters (2011). "Ligand-gated ion channels." In: *British Journal of Pharmacology* 164.s1, S115–S135. DOI: https://doi.org/10.1111/j.1476-5381.2011.01649_4.x. eprint: https://bpspubs.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1476-5381.2011.01649_4.x. URL: https://bpspubs.onlinelibrary.wiley.com/doi/abs/10.1111/j.1476-5381.2011.01649_4.x.

Almanstötter, Marius, Roman Vetter, and Dagmar Iber (2025). *PINNverse: Accurate parameter estimation in differential equations from noisy data with constrained physics-informed neural networks*. arXiv: 2504.05248 [cs.LG]. URL: https://arxiv.org/abs/2504.05248.

Anand Krishna, R., R. V. S. Krishna Dutt, and P. Premchand (2023). "Automatic Differentiation Using Dual Numbers - Use Case." In: *Multi-disciplinary Trends in Artificial Intelligence*. Ed. by Raghava Morusupalli, Teja Santosh Dandibhotla, Vani Vathsala Atluri, David Windridge, Pawan Lingras, and Venkateswara Rao Komati. Cham: Springer Nature Switzerland, pp. 68–78. ISBN: 978-3-031-36402-0.

Anderson, David F. and Thomas G. Kurtz (2011). "Continuous Time Markov Chain Models for Chemical Reaction Networks." In: *Design and Analysis of Biomolecular Circuits: Engineering Approaches to Systems and Synthetic Biology*. Ed. by Heinz Koeppl, Gianluca Setti, Mario di Bernardo, and Douglas Densmore. New York, NY: Springer New York, pp. 3–42. ISBN: 978-1-4419-6766-4. DOI: 10.1007/978-1-4419-6766-4_1. URL: https://doi.org/10.1007/978-1-4419-6766-4_1.

Antunes, Gabriela and Erik De Schutter (2012). "A stochastic signaling network mediates the probabilistic induction of cerebellar long-term depression." In: *Journal of Neuroscience* 32.27, pp. 9288–9300. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.5976-11.2012.

Baker, Nathan, Frank Alexander, Timo Bremer, Aric Hagberg, Yannis Kevrekidis, Habib Najm, Manish Parashar, Abani Patra, James Sethian, Stefan Wild, et al. (2019). *Workshop Report on Basic Research Needs for Scientific Machine Learning: Core Technologies for Artificial Intelligence*. Tech. rep. USDOE Office of Science (SC), Washington, D.C. (United States). DOI: 10.2172/1478744. URL: https://www.osti.gov/biblio/1478744.

Bartol, Thomas M., Daniel X. Keller, Justin P. Kinney, Chandrajit L. Bajaj, Kristen M. Harris, Terrence J. Sejnowski, and Mary B. Kennedy (2015). "Computational reconstitution of spine calcium transients from individual proteins." In: *Frontiers in Synaptic Neuroscience* 7.OCT, pp. 1–24. ISSN: 16633563. DOI: 10.3389/fnsyn.2015.00017.

Bayat, Mahnaz, Mohammad Davood Sharifi, Masoud Haghani, and Mohammad Shabani (2015). "Enriched environment improves synaptic plasticity and cognitive deficiency in chronic cerebral hypoperfused rats." In: *Brain Research Bulletin* 119, pp. 34–40. ISSN: 0361-9230. DOI: https://doi.org/10.1016/j.brainresbull.2015.10.001. URL: https://www.sciencedirect.com/science/article/pii/S0361923015300411.

Bayazitov, Ildar T., Robert J. Richardson, Robert G. Fricke, and Stanislav S. Zakharenko (2007). "Slow Presynaptic and Fast Postsynaptic Components of Compound Long-Term Potentiation." In: *Journal of Neuroscience* 27.43, pp. 11510–11521. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.3077-07.2007. eprint: https://www.jneurosci.org/content/27/43/11510.full.pdf. URL: https://www.jneurosci.org/content/27/43/11510.

Beed, Prateep et al. (2020). "Species-specific differences in synaptic transmission and plasticity." In: *Scientific Reports* 10.1, pp. 1–9. ISSN: 20452322. DOI: 10.1038/s41598-020-73547-6. URL: https://doi.org/10.1038/s41598-020-73547-6.

Bell, Miriam, Tom Bartol, Terrence Sejnowski, and Padmini Rangamani (2019). "Dendritic spine geometry and spine apparatus organization govern the spatiotemporal dynamics of calcium." In: *Journal of General Physiology* 151 (8), pp. 1017–1034. ISSN: 15407748. DOI: 10.1085/jgp.201812261.

Benke, Tim and Stephen F. Traynelis (2019). "AMPA-Type Glutamate Receptor Conductance Changes and Plasticity: Still a Lot of Noise." In: *Neurochemical Research* 44.3, pp. 539–548. ISSN: 15736903. DOI: 10.1007/s11064-018-2491-1. URL: http://dx.doi.org/10.1007/s11064-018-2491-1.

Berridge, Michael J., Martin D. Bootman, and H. Llewelyn Roderick (2003). "Calcium signalling: Dynamics, homeostasis and remodelling." In: *Nature Reviews Molecular Cell Biology* 4 (7), pp. 517–529. ISSN: 14710072. DOI: 10.1038/nrm1155.

Bhalla, Upinder S. and Ravi Iyengar (1999). "Emergent properties of networks of biological signaling pathways." In: *Science* 283 (5400), pp. 381–387. ISSN: 00368075. DOI: 10.1126/science.283.5400.381.

Bibeau, Valérie, Daria Camilla Boffito, and Bruno Blais (2024). "Physics-informed Neural Network to predict kinetics of biodiesel production in microwave reactors." In: *Chemical Engineering and Processing - Process Intensification* 196, p. 109652. ISSN: 0255-2701. DOI: https://doi.org/10.1016/j.cep.2023.109652. URL: https://www.sciencedirect.com/science/article/pii/S0255270123003896.

Bloodgood, Brenda L. and Bernardo L. Sabatini (2008). "Regulation of synaptic signalling by postsynaptic, non-glutamate receptor ion channels." In: *Journal of Physiology*. Vol. 586, pp. 1475–1480. DOI: 10.1113/jphysiol.2007.148353.

Bonate, Peter L. (2017). "Effect of correlation on covariate selection in linear and nonlinear mixed effect models." In: *Pharmaceutical Statistics* 16.1, pp. 45–54. DOI: https://doi.org/10.1002/pst.1776. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/pst.1776. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/pst.1776.

Brini, M. and E. Carafoli (2011). "The Plasma Membrane Ca^{2+} ATPase and the NCX cooperate in the regulation of cell calcium." In: *Cold Spring Harbor Perspectives in Biology*, pp. 1–15.

Brunton, Steven L., Joshua L. Proctor, and J. Nathan Kutz (2016). "Discovering governing equations from data by sparse identification of nonlinear dynamical systems." In: *Proceedings of the National Academy of Sciences* 113.15, pp. 3932–3937. DOI: 10.1073/pnas.1517384113. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.1517384113. URL: https://www.pnas.org/doi/abs/10.1073/pnas.1517384113.

Byrne, Michael J., John A. Putkey, M. Neal Waxham, and Yoshihisa Kubota (2009). "Dissecting cooperative calmodulin binding to CaM kinase II: A detailed stochastic model." In: *Journal of Computational Neuroscience* 27 (3), pp. 621–638. ISSN: 09295313. DOI: 10.1007/s10827-009-0173-3.

Carpenter, Bob, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus A. Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell (2017). "Stan: A probabilistic programming language." In: *Journal of Statistical Software* 76 (1). ISSN: 15487660. DOI: 10.18637/jss.v076.i01.

Cava, William La, Patryk Orzechowski, Bogdan Burlacu, Fabrício Olivetti de França, Marco Virgolin, Ying Jin, Michael Kommenda, and Ja-

son H. Moore (2021). *Contemporary Symbolic Regression Methods and their Relative Performance*. arXiv: 2107.14351 [cs.NE]. URL: https://arxiv.org/abs/2107.14351.

Chen, Ricky T.Q., Yulia Rubanova, Jesse Bettencourt, and David Duvenaud (2018). "Neural ordinary differential equations." In: *Advances in Neural Information Processing Systems* 2018-Decem, pp. 6571–6583. ISSN: 10495258.

Colbran, Roger J. (2004). "Targeting of calcium/calmodulin-dependent protein kinase II." In: *Biochemical Journal* 378.1, pp. 1–16. ISSN: 02646021. DOI: 10.1042/BJ20031547.

Colquhoun, David, Kathryn A. Dowsland, Marco Beato, and Andrew J.R. Plested (2004). "How to impose microscopic reversibility in complex reaction mechanisms." In: *Biophysical Journal* 86 (6), pp. 3510–3518. ISSN: 00063495. DOI: 10.1529/biophysj.103.038679.

Craciun, Gheorghe and Casian Pantea (2008). "Identifiability of chemical reaction networks." In: *Journal of Mathematical Chemistry* 44 (1), pp. 244–259. ISSN: 02599791. DOI: 10.1007/s10910-007-9307-x.

Cranmer, Kyle, Johann Brehmer, and Gilles Louppe (Dec. 2020). "The frontier of simulation-based inference." In: *Proceedings of the National Academy of Sciences of the United States of America* 117 (48), pp. 30055–30062. ISSN: 10916490. DOI: 10.1073/pnas.1912789117.

Dandekar, Raj, Chris Rackauckas, and George Barbastathis (2020). "A Machine Learning-Aided Global Diagnostic and Comparative Tool to Assess Effect of Quarantine Control in COVID-19 Spread." In: *Patterns* 1 (9). ISSN: 26663899. DOI: 10.1016/j.patter.2020.100145.

Daniels, Bryan C., Yan Jiun Chen, James P. Sethna, Ryan N. Gutenkunst, and Christopher R. Myers (2008). "Sloppiness, robustness, and evolvability in systems biology." In: *Current Opinion in Biotechnology* 19 (4), pp. 389–395. ISSN: 09581669. DOI: 10.1016/j.copbio.2008.06.008.

Dax, Maximilian, Stephen R. Green, Jonathan Gair, Nihar Gupte, Michael Pürrer, Vivien Raymond, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf (Mar. 2025). "Real-time inference for binary neutron star mergers using machine learning." In: *Nature* 639.8053, 49–53. ISSN: 1476-4687. DOI: 10.1038/s41586-025-08593-z. URL: http://dx.doi.org/10.1038/s41586-025-08593-z.

De Oliveira, Júlia Toffoli, Letícia Reggiane De Carvalho Costa, Cardoso Estumano Diego, and Liliana Amaral Féris (2025). "Applying Bayesian statistics and MCMC to ozone reaction kinetics: Implications for water treatment models." In: *Chemosphere* 373, p. 144164. ISSN: 0045-6535. DOI: https://doi.org/10.1016/j.chemosphere.2025.144164. URL: https://www.sciencedirect.com/science/article/pii/S0045653525001067.

De Schutter, Erik (2013). "The importance of stochastic signaling processes in the induction of long-term synaptic plasticity." In: *Neural Networks* 47, pp. 3–10. ISSN: 08936080. DOI: 10.1016/j.neunet.2012.11.015. URL: http://dx.doi.org/10.1016/j.neunet.2012.11.015.

Dempster, A.P., N.M. Laird, and D.B. Rubin (1977). "Maximum likelihood from incomplete data via the EM algorithm." In: *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 1–38.

Detoledo-Morrell, Leyla, Yuri Geinisman, and Frank Morrell (1988). "Age-dependent alterations in hippocampal synaptic plasticity: Relation to memory disorders." In: *Neurobiology of Aging* 9, pp. 581–590. ISSN: 0197-4580. DOI: https://doi.org/10.1016/S0197-4580(88)80117-9. URL: https://www.sciencedirect.com/science/article/pii/S0197458088801179.

Diering, Graham H. and Richard L. Huganir (2018). "The AMPA Receptor Code of Synaptic Plasticity." In: *Neuron* 100.2, pp. 314–329. ISSN: 10974199. DOI: 10.1016/j.neuron.2018.10.018. URL: https://doi.org/10.1016/j.neuron.2018.10.018.

Duchesne, Ronan, Anissa Guillemin, Olivier Gandrillon, and Fabien Crauste (2021). "Practical identifiability in the frame of nonlinear mixed effects models: the example of the in vitro erythropoiesis." In: *BMC Bioinformatics*, pp. 1–21. DOI: 10.1186/s12859-021-04373-4. URL: https://doi.org/10.1186/s12859-021-04373-4.

Edelmann, Elke, Efrain Cepeda-Prado, Martin Franck, Petra Lichtenecker, Tanja Brigadski, and Volkmar Leßmann (2015). "Theta Burst Firing Recruits BDNF Release and Signaling in Postsynaptic CA1 Neurons in Spike-Timing-Dependent LTP." In: *Neuron* 86.4, pp. 1041–1054. ISSN: 10974199. DOI: 10.1016/j.neuron.2015.04.007. URL: http://dx.doi.org/10.1016/j.neuron.2015.04.007.

Edelmann, Elke, Efrain Cepeda-Prado, and Volkmar Leßmann (2017). "Coexistence of multiple types of synaptic plasticity in individual hippocampal CA1 pyramidal neurons." In: *Frontiers in Synaptic Neuroscience* 9.MAR, pp. 1–15. ISSN: 16633563. DOI: 10.3389/fnsyn.2017.00007.

Eriksson, Olivia, Alexandra Jauhiainen, Sara Maad Sasane, Andrei Kramer, Anu G. Nair, Carolina Sartorius, and Jeanette Hellgren Kotaleski (2019). "Uncertainty quantification, propagation and characterization by Bayesian analysis combined with global sensitivity analysis applied to dynamical intracellular pathway models." In: *Bioinformatics* 35 (2), pp. 284–292. ISSN: 14602059. DOI: 10.1093/bioinformatics/bty607.

Faas, Guido C., Sridhar Raghavachari, John E. Lisman, and Istvan Mody (2011). "Calmodulin as a direct detector of Ca2+ signals."

In: *Nature Neuroscience* 14 (3), pp. 301–304. ISSN: 10976256. DOI: 10.1038/nn.2746.

Feinberg, M. (2019). *Foundations of Chemical Reaction Network Theory*. Applied Mathematical Sciences. Springer International Publishing. ISBN: 9783030038588. URL: https://books.google.co.uk/books?id=IBOGDwAAQBAJ.

Fjelde, Tor Erlend et al. (Feb. 2025). "Turing.jl: a general-purpose probabilistic programming language." In: *ACM Transactions on Probabilistic Machine Learning*. DOI: 10.1145/3711897.

Fredkin, Donald R. and John A. Rice (1985). "Aggregated Markov Processes and Channel Gating Kinetics." In: *Journal of Research of the National Bureau of Standards* 90 (6), pp. 517–520.

Frey, Sabine and Julietta U. Frey (2009). "Synaptic plasticity and the analysis of the field-EPSP as well as the population spike using separate recording electrodes in the dentate gyrus in freely moving rats." In: *Journal of Neuroscience Methods* 184.1, pp. 79–87. ISSN: 0165-0270. DOI: https://doi.org/10.1016/j.jneumeth.2009.07.024. URL: https://www.sciencedirect.com/science/article/pii/S0165027009004026.

Gaertner, Tara R., John A. Putkey, and M. Neal Waxham (2004). "RC3/neurogranin and Ca2+/calmodulin-dependent protein kinase II produce opposing effects on the affinity of calmodulin for calcium." In: *Journal of Biological Chemistry* 279 (38), pp. 39374–39382. ISSN: 00219258. DOI: 10.1074/jbc.M405352200.

Gallimore, Andrew R., Taegon Kim, Keiko Tanaka-Yamamoto, and Erik De Schutter (2018). "Switching On Depression and Potentiation in the Cerebellum." In: *Cell Reports* 22 (3), pp. 722–733. ISSN: 22111247. DOI: 10.1016/j.celrep.2017.12.084.

Gillespie, Daniel T. (1977). "Exact stochastic simulation of coupled chemical reactions." In: *Journal of Physical Chemistry* 81.25, pp. 2340–2361. ISSN: 00223654. DOI: 10.1021/j100540a008.

— (2007). "Stochastic simulation of chemical kinetics." In: *Annual Review of Physical Chemistry* 58, pp. 35–55. ISSN: 0066426X. DOI: 10.1146/annurev.physchem.58.032806.104637.

Glasgow, Stephen D., Ryan McPhedrain, Jeanne F. Madranges, Timothy E. Kennedy, and Edward S. Ruthazer (2019). *Approaches and limitations in the investigation of synaptic transmission and plasticity*. DOI: 10.3389/fnsyn.2019.00020.

Gloeckler, Manuel, Shoji Toyota, Kenji Fukumizu, and Jakob H. Macke (2025). *Compositional simulation-based inference for time series*. arXiv: 2411.02728 [cs.LG]. URL: https://arxiv.org/abs/2411.02728.

Goraya, Tasmina A. and Dermot M.F. Cooper (2005). "Ca2+-calmodulin-dependent phosphodiesterase (PDE1): Current perspectives." In: *Cellular Signalling* 17 (7), pp. 789–797. ISSN: 08986568. DOI: 10.1016/j.cellsig.2004.12.017.

Grant, Seth G.N. and Erik Fransén (2020). "The Synapse Diversity Dilemma: Molecular Heterogeneity Confounds Studies of Synapse Function." In: *Frontiers in Synaptic Neuroscience* 12. ISSN: 16633563. DOI: 10.3389/fnsyn.2020.590403.

Grigorian, Gevik, Sandip V. George, and Simon Arridge (2024). *Learning Governing Equations of Unobserved States in Dynamical Systems*. arXiv: 2404.18572 [cs.LG]. URL: https://arxiv.org/abs/2404.18572.

Gutenkunst, Ryan N., Joshua J. Waterfall, Fergal P. Casey, Kevin S. Brown, Christopher R. Myers, and James P. Sethna (2007). "Universally sloppy parameter sensitivities in systems biology models." In: *PLoS Computational Biology* 3.10, pp. 1871–1878. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.0030189.

Hajjem, Ahlem, François Bellavance, and Denis Larocque (2011). "Mixed effects regression trees for clustered data." In: *Statistics & Probability Letters* 81.4, pp. 451–459. ISSN: 0167-7152. DOI: https://doi.org/10.1016/j.spl.2010.12.003. URL: https://www.sciencedirect.com/science/article/pii/S0167715210003433.

— (2014). "Mixed-effects random forest for clustered data." In: *Journal of Statistical Computation and Simulation* 84.6, pp. 1313–1328. DOI: 10.1080/00949655.2012.741599. eprint: https://doi.org/10.1080/00949655.2012.741599. URL: https://doi.org/10.1080/00949655.2012.741599.

Hajjem, Ahlem, Denis Larocque, and François Bellavance (2017). "Generalized mixed effects regression trees." In: *Statistics & Probability Letters* 126, pp. 114–118. ISSN: 0167-7152. DOI: https://doi.org/10.1016/j.spl.2017.02.033. URL: https://www.sciencedirect.com/science/article/pii/S0167715217300895.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*. ISBN: 0387848576.

Haworth, Alexander S. and William J. Brackenbury (2019). "Emerging roles for multifunctional ion channel auxiliary subunits in cancer." In: *Cell Calcium* 80, pp. 125–140. ISSN: 0143-4160. DOI: https://doi.org/10.1016/j.ceca.2019.04.005. URL: https://www.sciencedirect.com/science/article/pii/S014341601930051X.

Hayer, Arnold and Upinder S. Bhalla (2005). "Molecular switches at the synapse emerge from receptor and kinase traffic." In: *PLoS Computational Biology* 1 (2), pp. 0137–0154. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.0010020.

Heid, Esther and William H. Green (2022). "Machine Learning of Reaction Properties via Learned Representations of the Condensed Graph of Reaction." In: *Journal of Chemical Information and Modeling* 62 (9), pp. 2101–2110. ISSN: 1549960X. DOI: 10.1021/acs.jcim.1c00975.

Heil, Katharina F., Emilia M. Wysocka, Oksana Sorokina, Jeanette Hellgren Kotaleski, T. Ian Simpson, J. Douglas Armstrong, and David C. Sterratt (2018). "Analysis of proteins in computational models of synaptic plasticity." In: *bioRxiv*. DOI: 10.1101/254094. eprint: https://www.biorxiv.org/content/early/2018/01/28/254094.full.pdf. URL: https://www.biorxiv.org/content/early/2018/01/28/254094.

Heine, Martin, Laurent Groc, Renato Frischknecht, Jean Claude Béïque, Brahim Lounis, Gavin Rumbaugh, Richard L. Huganir, Laurent Cognet, and Daniel Choquet (2008). "Surface mobility of postsynaptic AMPARs tunes synaptic transmission." In: *Science* 320.5873, pp. 201–205. ISSN: 00368075. DOI: 10.1126/science.1152089.

Hermans, Joeri, Volodimir Begy, and Gilles Louppe (2020). *Likelihood-free MCMC with Amortized Approximate Ratio Estimators*. arXiv: 1903.04057 [stat.ML]. URL: https://arxiv.org/abs/1903.04057.

Hodgkin, A. L. and A. F. Huxley (1952). "A quantitative description of membrane current and its application to conduction and excitation in nerve." In: *The Journal of Physiology* 117.4, pp. 500–544. DOI: https://doi.org/10.1113/jphysiol.1952.sp004764. eprint: https://physoc.onlinelibrary.wiley.com/doi/pdf/10.1113/jphysiol.1952.sp004764. URL: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1952.sp004764.

Hu, Shuwen, You Gan Wang, Christopher Drovandi, and Taoyun Cao (2023). "Predictions of machine learning with mixed-effects in analyzing longitudinal data under model misspecification." In: *Statistical Methods and Applications* 32 (2), pp. 681–711. ISSN: 1613981X. DOI: 10.1007/s10260-022-00658-x.

Hyun, Jae Pi and John E. Lisman (2008). "Coupled phosphatase and kinase switches produce the tristability required for long-term potentiation and long-term depression." In: *Journal of Neuroscience* 28.49, pp. 13132–13138. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.2348-08.2008.

Incontro, Salvatore et al. (2018). "The CaMKII/NMDA receptor complex controls hippocampal synaptic transmission by kinase-dependent and independent mechanisms." In: *Nature Communications* 9 (1). ISSN: 20411723. DOI: 10.1038/s41467-018-04439-7.

Jayant, K., J. J. Hirtz, I. J. Plante, D. M. Tsai, A. De Boer W. D. A. M. Semonche, D. S. Peterka, J. S. Owen, O. Sahin, K. L. Shepard, and R. Yuste (2017). "Targeted intracellular voltage recordings from dendritic spines using quantum-dot-coated nanopipettes." In: *Neuron* 12, pp. 335–342.

Jedrzejewska-Szmek, Joanna, Vincent Luczak, Ted Abel, and Kim T. Blackwell (2017). "β-adrenergic signaling broadly contributes to LTP induction." In: *PLoS Computational Biology* 13.7, pp. 1–32. DOI: 10.1371/journal.pcbi.1005657.

Ji, Weiqi and Sili Deng (2021). "Autonomous Discovery of Unknown Reaction Pathways from Data by Chemical Reaction Neural Network." In: *Journal of Physical Chemistry A* 125 (4), pp. 1082–1092. ISSN: 15205215. DOI: 10.1021/acs.jpca.0c09316.

Jumper, John et al. (2021). "Highly accurate protein structure prediction with AlphaFold." In: *Nature* 596 (7873), pp. 583–589. ISSN: 14764687. DOI: 10.1038/s41586-021-03819-2.

Karlsson, Markus, David L.I. Janzén, Lucia Durrieu, Alejandro Colman-Lerner, Maria C. Kjellsson, and Gunnar Cedersund (2015). "Nonlinear mixed-effects modelling for single cell estimation: When, why, and how to use it." In: *BMC Systems Biology* 9 (1). ISSN: 17520509. DOI: 10.1186/s12918-015-0203-x.

Kefauver, J. M., A. B. Ward, and A. Patapoutian (2020). "Discoveries in structure and physiology of mechanically activated ion channels." In: *Nature* 587 (7835), pp. 567–576. ISSN: 14764687. DOI: 10.1038/s41586-020-2933-1.

Kim, Myungsook, Ted Huang, Ted Abel, and Kim T. Blackwell (2010). "Temporal sensitivity of protein kinase A activation in late-phase long term potentiation." In: *PLoS Computational Biology* 6 (2). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1000691.

Kim, Myungsook, Alan Jung Park, Robbert Havekes, Andrew Chay, Leonardo Antonio Guercio, Rodrigo Freire Oliveira, Ted Abel, and Kim T. Blackwell (2011). "Colocalization of protein kinase a with adenylyl cyclase enhances protein kinase a activity during induction of Long-Lasting Long-Term-Potentiation." In: *PLoS Computational Biology* 7.6. ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1002084.

Kobyzev, Ivan, Simon J. D. Prince, and Marcus A. Brubaker (2019). "Normalizing Flows: An Introduction and Review of Current Methods." In: DOI: 10.1109/TPAMI.2020.2992934. URL: http://arxiv.org/abs/1908.09257http://dx.doi.org/10.1109/TPAMI.2020.2992934.

Krishnapriyan, Aditi S., Amir Gholami, Shandian Zhe, Robert M. Kirby, and Michael W. Mahoney (2021). *Characterizing possible failure modes in physics-informed neural networks*. arXiv: 2109.01050 [cs.LG]. URL: https://arxiv.org/abs/2109.01050.

Kristensen, Anders S, Meagan a Jenkins, Tue G Banke, Arne Schousboe, Richard C Johnson, Richard Huganir, and Stephen F Traynelis (2011). "Mechanism of CaMKII regulation of AMPA receptor gating." In: *Nat Neurosci* 14.6, pp. 727–735. DOI: 10.1038/nn.2804.Mechanism.

Kuhn, Thomas Samuel (1962). *The Structure of Scientific Revolutions*. Ed. by Otto Neurath. Chicago: University of Chicago Press.

Laban, Guy, Sophie Chiang, and Hatice Gunes (2025). *What People Share With a Robot When Feeling Lonely and Stressed and How It*

Helps Over Time. arXiv: 2504.02991 [cs.HC]. URL: https://arxiv.org/abs/2504.02991.

Lai, Massimo, Denis Brun, Stuart J. Edelstein, and Nicolas Le Novère (2015). "Modulation of Calmodulin Lobes by Different Targets: An Allosteric Model with Hemiconcerted Conformational Transitions." In: *PLoS Computational Biology* 11 (1). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1004063.

Lampert, Angelika and Alon Korngreen (2014). "Chapter One - Markov Modeling of Ion Channels: Implications for Understanding Disease." In: *Computational Neuroscience*. Ed. by Kim T. Blackwell. Vol. 123. Progress in Molecular Biology and Translational Science. Academic Press, pp. 1–21. DOI: https://doi.org/10.1016/B978-0-12-397897-4.00009-7. URL: https://www.sciencedirect.com/science/article/pii/B9780123978974000097.

Larsen, Rylan S. and P. Jesper Sjöström (2015). "Synapse-type-specific plasticity in local circuits." In: *Current Opinion in Neurobiology* 35, pp. 127–135. ISSN: 18736882. DOI: 10.1016/j.conb.2015.08.001.

Lee, Seok Jin R., Yasmin Escobedo-Lozoya, Erzsebet M. Szatmari, and Ryohei Yasuda (2009). "Activation of CaMKII in single dendritic spines during long-term potentiation." In: *Nature* 458 (7236), pp. 299–304. ISSN: 00280836. DOI: 10.1038/nature07842. URL: http://dx.doi.org/10.1038/nature07842.

Lima, Fernando Arrais R.D., Carine M. Rebello, Erbet A. Costa, Vinícius V. Santana, Marcellus G.F. de Moares, Amaro G. Barreto, Argimiro R. Secchi, Maurício B. de Souza, and Idelfonso B.R. Nogueira (2023). "Improved modeling of crystallization processes by Universal Differential Equations." In: *Chemical Engineering Research and Design* 200, pp. 538–549. ISSN: 02638762. DOI: 10.1016/j.cherd.2023.11.032.

Lindstrom, Mary J and Douglas M Bates (1990). "Nonlinear Mixed Effects Models for Repeated Measures Data." In: *Biometrics* 46 (3), pp. 673–687. URL: https://www.jstor.org/stable/2532087?seq=1&cid=pdf-.

Linkevicius, D. (2019). "Mixed electrical-biochemical modeling of synaptic plasticity in hippocampal CA1 pyramidal cells using KappaNEURON." MScR thesis at University of Edinburgh.

Linse, Sara, Anna Helmersson, and Sture Forsen (1991). "The Journal of Biological Chemistry." In: 266 (13), pp. 8050–8054.

Liu, Yan, Yiming Mo, and Youwei Cheng (2025). "Machine Learning Accelerated Analysis of Chemical Reaction Networks for Gas-Phase Reaction Systems." In: *Industrial and Engineering Chemistry Research*. ISSN: 15205045. DOI: 10.1021/acs.iecr.4c03938.

Lueckmann, Jan-Matthis, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke (2017). "Flexible Statistical Inference for Mechanistic Models of Neural Dynamics." In: *Advances in Neural Information Processing Systems*. Ed. by

I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc.

Lunn, David J. and Leon J. Aarons (2002). "Markov Chain Monte Carlo Techniques for Studying Interoccasion and Intersubject Variability: Application to Pharmacokinetic Data." In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 46.1, pp. 73–91. ISSN: 0035-9254. DOI: 10.1111/1467-9876.00048. eprint: https://academic.oup.com/jrsssc/article-pdf/46/1/73/48750310/jrsssc_46_1_73.pdf. URL: https://doi.org/10.1111/1467-9876.00048.

Maeder, Marcel, Yorck-Michael Neuhold, and Graeme Puxty (2004). "Application of a genetic algorithm: near optimal estimation of the rate and equilibrium constants of complex reaction mechanisms." In: *Chemometrics and Intelligent Laboratory Systems* 70.2, pp. 193–203. ISSN: 0169-7439. DOI: https://doi.org/10.1016/j.chemolab.2003.11.006. URL: https://www.sciencedirect.com/science/article/pii/S0169743903002053.

Magee, Jeffrey C. and Christine Grienberger (2020). "Synaptic Plasticity Forms and Functions." In: *Annual Review of Neuroscience* 43, pp. 95–117. ISSN: 15454126. DOI: 10.1146/annurev-neuro-090919-022842.

Malmendal, Anders, Johan Evenäs, Sture Forsén, and Mikael Akke (1999). "Structural dynamics in the C-terminal domain of calmodulin at low calcium levels11Edited by P. E. Wright." In: *Journal of Molecular Biology* 293.4, pp. 883–899. ISSN: 0022-2836. DOI: https://doi.org/10.1006/jmbi.1999.3188. URL: https://www.sciencedirect.com/science/article/pii/S0022283699931881.

Mao, Li Min, Dao Zhong Jin, Bing Xue, Xiang Ping Chu, and John Q. Wang (2014). "Phosphorylation and regulation of glutamate receptors by CaMKII." In: *Sheng li xue bao : [Acta physiologica Sinica]* 66.3, pp. 365–372. ISSN: 03710874. DOI: 10.13294/j.aps.2014.0044.

Martensen, Carl Julius, Niklas Korsbo, Vijay Ivaturi, and Sebastian Sager (2024). "Data-Driven Discovery of Feedback Mechanisms in Acute Myeloid Leukaemia: Alternatives to classical models using Deep Nonlinear Mixed Effect modeling and Symbolic Regression." In: *bioRxiv*. DOI: 10.1101/2024.06.17.599366. eprint: https://www.biorxiv.org/content/early/2024/06/19/2024.06.17.599366.full.pdf. URL: https://www.biorxiv.org/content/early/2024/06/19/2024.06.17.599366.

Meyer, Tobias, Phyllis I. Hanson, Lubert Stryer, and Howard Schulman (1992). "Calmodulin Trapping by Calcium-Calmodulin-Dependent Protein Kinase." In: *Science* 256.5060, pp. 1199–1202. DOI: 10.1126/science.256.5060.1199. eprint: https://www.science.org/doi/pdf/10.1126/science.256.5060.1199. URL: https://www.science.org/doi/abs/10.1126/science.256.5060.1199.

Michalski, P. J. and L. M. Loew (2012). "CaMKII activation and dynamics are independent of the holoenzyme structure: An infinite subunit holoenzyme approximation." In: *Physical Biology* 9 (3). ISSN: 14783967. DOI: 10.1088/1478-3975/9/3/036010.

Miller, Paul, Anatol M. Zhabotinsky, John E. Lisman, and Xiao Jing Wang (2005). "The stability of a stochastic CaMKII switch: Dependence on the number of enzyme molecules and protein turnover." In: *PLoS Biology* 3 (4), pp. 0705–0717. ISSN: 15449173. DOI: 10.1371/journal.pbio.0030107.

Nair, Anu G., Upinder S. Bhalla, and Jeanette Hellgren Kotaleski (2016). "Role of DARPP-32 and ARPP-21 in the Emergence of Temporal Constraints on Striatal Calcium and Dopamine Integration." In: *PLoS Computational Biology* 12 (9). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1005080.

Nair, Anu G., Omar Gutierrez-Arenas, Olivia Eriksson, Alexandra Jauhiainen, Kim T. Blackwell, and Jeanette H. Kotaleski (2014). *Modeling Intracellular Signaling Underlying Striatal Function in Health and Disease*. 1st ed. Vol. 123. Elsevier Inc., pp. 277–304. ISBN: 9780123978974. DOI: 10.1016/b978-0-12-397897-4.00013-9. URL: http://dx.doi.org/10.1016/B978-0-12-397897-4.00013-9.

Neal, Radford M. (1996). "Monte Carlo Implementation." In: *Bayesian Learning for Neural Networks*. New York, NY: Springer New York, pp. 55–98. ISBN: 978-1-4612-0745-0. DOI: 10.1007/978-1-4612-0745-0_3.

Nieves, Emily, Raj Dandekar, and Chris Rackauckas (2024). "Uncertainty quantified discovery of chemical reaction systems via Bayesian scientific machine learning." In: *Frontiers in Systems Biology* 4. ISSN: 26740702. DOI: 10.3389/fsysb.2024.1338518.

O'Reilly-Shah, Vikas, Benjamin Chagot, and Walter Chazin (2015). "Calcium-Dependent Regulation of Ion Channels." In: *Calcium Binding Proteins* 1.

Oddi, Facundo J., Fernando E. Miguez, Luciana Ghermandi, Lucas O. Bianchi, and Lucas A. Garibaldi (2019). "A nonlinear mixed-effects modeling approach for ecological data: Using temporal dynamics of vegetation moisture as an example." In: *Ecology and Evolution* 9.18, pp. 10225–10240. DOI: https://doi.org/10.1002/ece3.5543. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/ece3.5543. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ece3.5543.

Oikonomou, Efthymios, Yannick Juli, Rajkumar Reddy Kolan, Linda Kern, Thomas Gruber, Christian Alzheimer, Patrick Krauss, Andreas Maier, and Tobias Huth (2024). "A deep learning approach to real-time Markov modeling of ion channel gating." In: *Communications chemistry* 7 (1), p. 280. ISSN: 2399-3669. DOI: 10.1038/s42004-024-01369-y. URL: http://www.ncbi.nlm.nih.gov/pubmed/39616256.

Opazo, Patricio, Simon Labrecque, Cezar M. Tigaret, Arnaud Frouin, Paul W. Wiseman, Paul De Koninck, and Daniel Choquet (2010). "CaMKII triggers the diffusional trapping of surface AMPARs through phosphorylation of stargazin." In: *Neuron* 67.2, pp. 239–252. ISSN: 08966273. DOI: 10.1016/j.neuron.2010.06.007. URL: http://dx.doi.org/10.1016/j.neuron.2010.06.007.

Ordyan, Mariam, Tom Bartol, Mary Kennedy, Padmini Rangamani, and Terrence Sejnowski (2020). "Interactions between calmodulin and neurogranin govern the dynamics of CaMKII as a leaky integrator." In: *PLoS Computational Biology* 16 (7). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1008015.

Pastoll, Hugh, Derek L Garden, Ioannis Papastathopoulos, Gülşen Sürmeli, and Matthew F Nolan (2020). "Inter- and intra-animal variation in the integrative properties of stellate cells in the medial entorhinal cortex." In: *eLife* 9. Ed. by Laura L Colgin, Lisa Giocomo, and Michael E Hasselmo, e52258. ISSN: 2051-084X. DOI: 10.7554/eLife.52258. URL: https://doi.org/12.7554/eLife.52258.

Payne, Helen L. (2008). "The role of transmembrane AMPA receptor regulatory proteins (TARPs) in neurotransmission and receptor trafficking (Review)." In: *Molecular Membrane Biology* 25.4, pp. 353–362. ISSN: 09687688. DOI: 10.1080/09687680801986480.

Pepke, Shirley, Tamara Kinzer-Ursem, Stefan Mihalas, and Mary B. Kennedy (2010). "A dynamic model of interactions of Ca2+, calmodulin, and catalytic subunits of Ca2+/calmodulin-dependent protein kinase II." In: *PLoS Computational Biology* 6 (2). ISSN: 1553734X. DOI: 10.1371/journal.pcbi.1000675.

Pharris, Matthew C., Neal M. Patel, and Tamara L. Kinzer-Ursem (2018). "Competitive Tuning Among Ca2+/Calmodulin-Dependent Proteins: Analysis of In Silico Model Robustness and Parameter Variability." In: *Cellular and Molecular Bioengineering* 11 (5), pp. 353–365. ISSN: 18655033. DOI: 10.1007/s12195-018-0549-4.

Piazza, Michael, Valentina Taiakina, Thorsten Dieckmann, and J. Guy Guillemette (2017). "Structural Consequences of Calmodulin EF Hand Mutations." In: *Biochemistry* 56 (7), pp. 944–956. ISSN: 15204995. DOI: 10.1021/acs.biochem.6b01296.

Pillai, Goonaseelan, France Mentré, and Jean Louis Steimer (2005). "Non-linear mixed effects modeling - From methodology and software development to driving implementation in drug development science." In: *Journal of Pharmacokinetics and Pharmacodynamics* 32 (2), pp. 161–183. ISSN: 1567567X. DOI: 10.1007/s10928-005-0062-y.

Polifke, Wolfgang, Weiqun Geng, and Klaus Döbbeling (1998). "Optimization of Rate Coefficients for Simplified Reaction Mechanisms with Genetic Algorithms." In: *Combustion and Flame* 113.1, pp. 119–134. ISSN: 0010-2180. DOI: https://doi.org/10.1016/

S0010-2180(97)00212-5. URL: https://www.sciencedirect.com/science/article/pii/S0010218097002125.

Porumb, T. (1994). "Determination of Calcium-Binding Constants by Flow Dialysis." In: *Analytical Biochemistry* 220.2, pp. 227–237. ISSN: 0003-2697. DOI: https://doi.org/10.1006/abio.1994.1332. URL: https://www.sciencedirect.com/science/article/pii/S0003269784713327.

Postlethwaite, M., M. H. Hennig, J. R. Steinert, B. P. Graham, and I. D. Forsythe (2007). "Acceleration of AMPA receptor kinetics underlies temperature-dependent changes in synaptic strength at the rat calyx of Held." In: *Journal of Physiology* 579.1, pp. 69–84. ISSN: 00223751. DOI: 10.1113/jphysiol.2006.123612.

Prokop, Bartosz and Lendert Gelens (2024). "From biological data to oscillator models using SINDy." In: *iScience* 27.4, p. 109316. ISSN: 2589-0042. DOI: https://doi.org/10.1016/j.isci.2024.109316. URL: https://www.sciencedirect.com/science/article/pii/S2589004224005376.

Putkey, John A., M. Neal Waxham, Tara R. Gaertner, Kari J. Brewer, Michael Goldsmith, Yoshihisa Kubota, and Quinn K. Kleerekoper (2008). "Acidic/IQ motif regulator of calmodulin." In: *Journal of Biological Chemistry* 283 (3), pp. 1401–1410. ISSN: 00219258. DOI: 10.1074/jbc.M703831200.

Quintana, Andrea R., Dan Wang, Joanna E. Forbes, and M. Neal Waxham (2005). "Kinetics of calmodulin binding to calcineurin." In: *Biochemical and Biophysical Research Communications* 334 (2), pp. 674–680. ISSN: 0006291X. DOI: 10.1016/j.bbrc.2005.06.152.

Racca, Claudia, F Anne Stephenson, Peter Streit, J David, B Roberts, and Peter Somogyi (2000). "NMDA Receptor Content of Synapses in Stratum Radiatum of the Hippocampal CA1 Area." In: *The Journal of Neuroscience* 20 (7), pp. 2512–2522.

Rackauckas, Chris et al. (2022). "Accelerated Predictive Healthcare Analytics with Pumas, A High Performance Pharmaceutical Modeling and Simulation Platform." In: *bioRxiv*. DOI: 10.1101/2020.11.28.402297. eprint: https://www.biorxiv.org/content/early/2022/03/20/2020.11.28.402297.full.pdf. URL: https://www.biorxiv.org/content/early/2022/03/20/2020.11.28.402297.

Rackauckas, Christopher, Yingbo Ma, Julius Martensen, Collin Warner, Kirill Zubov, Rohit Supekar, Dominic Skinner, Ali Ramadhan, and Alan Edelman (2020). "Universal Differential Equations for Scientific Machine Learning." In: URL: http://arxiv.org/abs/2001.04385.

Rackauskas, Christopher and Vijay Ivaturi (2019). "Neural-Embedded Nonlinear Mixed Effects Models (NENLME) in Pumas.jl." In: *American Conference on Pharmacometrics*.

Raissi, M., P. Perdikaris, and G.E. Karniadakis (2019). "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." In: *Journal of Computational Physics* 378, pp. 686 –707. ISSN: 0021-9991. DOI: https://doi.org/10.1016/j.jcp.2018.10.045. URL: http://www.sciencedirect.com/science/article/pii/S0021999118307125.

Raissi, Maziar and George Em Karniadakis (2018). "Hidden physics models: Machine learning of nonlinear partial differential equations." In: *Journal of Computational Physics* 357, pp. 125–141. ISSN: 10902716. DOI: 10.1016/j.jcp.2017.11.039. arXiv: 1708.00588. URL: https://doi.org/10.1016/j.jcp.2017.11.039.

Ranjan, Rajnish, Emmanuelle Logette, Michela Marani, Mirjia Herzog, Valérie Tâche, Enrico Scantamburlo, Valérie Buchillier, and Henry Markram (2019). "A Kinetic Map of the Homomeric Voltage-Gated Potassium Channel (Kv) Family." In: *Frontiers in Cellular Neuroscience* 13. ISSN: 16625102. DOI: 10.3389/fncel.2019.00358.

Regehr, W. G. and D. W. Tank (1992). "Calcium concentration dynamics produced by synaptic activation of CA1 hippocampal pyramidal cells." In: *Journal of Neuroscience* 12.11, pp. 4202–4223. ISSN: 02706474. DOI: 10.1523/jneurosci.12-11-04202.1992.

Ribeiro, Antônio H., Koen Tiels, Jack Umenberger, Thomas B. Schön, and Luis A. Aguirre (2020). "On the smoothness of nonlinear system identification." In: *Automatica* 121. ISSN: 00051098. DOI: 10.1016/j.automatica.2020.109158.

Roach, Cameron, Rob Hyndman, and Souhaib Ben Taieb (2021). "Nonlinear mixed-effects models for time series forecasting of smart meter demand." In: *Journal of Forecasting* 40.6, pp. 1118–1130. DOI: https://doi.org/10.1002/for.2750. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/for.2750. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/for.2750.

Roy, Marcia, Oksana Sorokina, Colin McLean, Silvia Tapia-González, Javier DeFelipe, J. Douglas Armstrong, and Seth G.N. Grant (2018). "Regional diversity in the postsynaptic proteome of the mouse brain." In: *Proteomes* 6 (3), pp. 1–18. ISSN: 22277382. DOI: 10.3390/proteomes6030031.

Sabatini, Bernardo L., Thomas G. Oertner, and Karel Svoboda (2002). "The life cycle of Ca2+ ions in dendritic spines." In: *Neuron* 33 (3), pp. 439–452. ISSN: 08966273. DOI: 10.1016/S0896-6273(02)00573-1.

Sansom, M.S., F.G. Ball, C.J. Kerry, R. McGee, R.L. Ramsey, and P.N. Usherwood (1989). "Markov, fractal, diffusion, and related models of ion channel gating. A comparison with experimental data from two ion channels." In: *Biophysical Journal* 56.6, pp. 1229–1243. ISSN: 0006-3495. DOI: https://doi.org/10.1016/S0006-

3495(89)82770-5. URL: https://www.sciencedirect.com/science/article/pii/S0006349589827705.

Sartori, N and T A Severini (2004). "Conditional likelihood inference in generalized linear mixed models." In: *Statistica Sinica* 14, pp. 349–360.

Scheuss, Volker, Ryohei Yasuda, Aleksander Sobczyk, and Karel Svoboda (2006). "Nonlinear [Ca2+] signaling in dendrites and spines caused by activity-dependent depression of Ca2+ extrusion." In: *Journal of Neuroscience* 26 (31), pp. 8183–8194. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.1962-06.2006.

Shen, Chaopeng et al. (2023). "Differentiable modelling to unify machine learning and physical models for geosciences." In: *Nature Reviews Earth and Environment* 4 (8), pp. 552–567. ISSN: 2662138X. DOI: 10.1038/s43017-023-00450-9.

Shifman, Julia M., Mee H. Choi, Stefan Mihalas, Stephen L. Mayo, and Mary B. Kennedy (2006). "Ca2+/calmodulin-dependent protein kinase II (CaMKII) is activated by calmodulin with two bound calciums." In: *Proceedings of the National Academy of Sciences* 103.38, pp. 13968–13973. DOI: 10.1073/pnas.0606433103. eprint: https://www.pnas.org/doi/pdf/10.1073/pnas.0606433103. URL: https://www.pnas.org/doi/abs/10.1073/pnas.0606433103.

Shouval, Harel Z., Mark F. Bear, and Leon N. Cooper (2002). "A unified model of NMDA receptor-dependent bidirectional synaptic plasticity." In: *Proceedings of the National Academy of Sciences of the United States of America* 99.16, pp. 10831–10836. ISSN: 00278424. DOI: 10.1073/pnas.152343099.

Snyder, Gretchen L., Stacey Galdi, Allen A. Fienberg, Patrick Allen, Angus C. Nairn, and Paul Greengard (2003). "Regulation of AMPA receptor dephosphorylation by glutamate receptor agonists." In: *Neuropharmacology* 45.6, pp. 703–713. ISSN: 00283908. DOI: 10.1016/S0028-3908(03)00319-8.

Somacal, Agustín, Yamila Barrera, Leonardo Boechi, Matthieu Jonckheere, Vincent Lefieux, Dominique Picard, and Ezequiel Smucler (2022). "Uncovering differential equations from data with hidden variables." In: *Phys. Rev. E* 105 (5), p. 054209. DOI: 10.1103/PhysRevE.105.054209. URL: https://link.aps.org/doi/10.1103/PhysRevE.105.054209.

Sorokina, Oksana, Colin Mclean, Mike D.R. Croning, Katharina F. Heil, Emilia Wysocka, Xin He, David Sterratt, Seth G.N. Grant, Thomas I. Simpson, and J. Douglas Armstrong (2021). "A unified resource and configurable model of the synapse proteome and its role in disease." In: *Scientific Reports* 11.1, pp. 1–9. ISSN: 20452322. DOI: 10.1038/s41598-021-88945-7. URL: https://doi.org/10.1038/s41598-021-88945-7.

Sorokina, Oksana, Anatoly Sorokin, and J. Douglas Armstrong (2011). "Towards a quantitative model of the post-synaptic proteome."

In: *Molecular BioSystems* 7.10, pp. 2813–2823. ISSN: 1742206X. DOI: 10.1039/c1mb05152k.

Stefan, Melanie I., Thomas M. Bartol, Terrence J. Sejnowski, and Mary B. Kennedy (2014). "Multi-state Modeling of Biomolecules." In: *PLoS Computational Biology* 10 (9). ISSN: 15537358. DOI: 10.1371/journal.pcbi.1003844.

Sumioka, Akio, Dan Yan, and Susumu Tomita (2010). "TARP Phosphorylation Regulates Synaptic AMPA Receptors through Lipid Bilayers." In: *Neuron* 66.5, pp. 755–767. ISSN: 08966273. DOI: 10.1016/j.neuron.2010.04.035. URL: http://dx.doi.org/10.1016/j.neuron.2010.04.035.

Thöni, Anna C. M., William E. Robinson, Yoram Bachrach, Wilhelm T. S. Huck, and Tal Kachman (2025). *Modelling Chemical Reaction Networks using Neural Ordinary Differential Equations*. arXiv: 2502.19397 [q-bio.MN]. URL: https://arxiv.org/abs/2502.19397.

Tozzi, Arturo (Oct. 2015). *Information processing in the CNS: a supramolecular chemistry?* DOI: 10.1007/s11571-015-9337-1.

Turanyi, Tam ' (1990). "Sensitivity analysis of complex kinetic systems. Tools and applications." In: *Journal of Mathematical Chemistry* 5, pp. 203–248.

Unsleber, Jan P and Markus Reiher (2025). "The Annual Review of Physical Chemistry is online at Downloaded from www." In: *Annual Review of Physical Chemistry Annu. Rev. Phys. Chem. 2020* 46, p. 54. DOI: 10.1146/annurev-physchem-071119. URL: https://doi.org/10.1146/annurev-physchem-071119-.

Urakubo, Hidetoshi, Minoru Honda, Robert C. Froemke, and Shinya Kuroda (2008). "Requirement of an allosteric kinetics of NMDA receptors for spike timing-dependent plasticity." In: *Journal of Neuroscience* 28 (13), pp. 3310–3323. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.0303-08.2008.

Urakubo, Hidetoshi, Miharu Sato, Shin Ishii, and Shinya Kuroda (2014). "In vitro reconstitution of a CaMKII memory switch by an NMDA receptor-derived peptide." In: *Biophysical Journal* 106.6, pp. 1414–1420. ISSN: 15420086. DOI: 10.1016/j.bpj.2014.01.026. URL: http://dx.doi.org/10.1016/j.bpj.2014.01.026.

Vetter, Julius, Jakob H. Macke, and Richard Gao (Sept. 2024). "Generating realistic neurophysiological time series with denoising diffusion probabilistic models." In: *Patterns* 5 (9). ISSN: 26663899. DOI: 10.1016/j.patter.2024.101047.

Virtanen, Pauli et al. (2020). "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python." In: *Nature Methods* 17, pp. 261–272. DOI: 10.1038/s41592-019-0686-2.

Walch, Olivia J. and Marisa C. Eisenberg (2016). "Parameter identifiability and identifiable combinations in generalized Hodgkin–Huxley models." In: *Neurocomputing* 199, pp. 137–143. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2016.03.027.

URL: https://www.sciencedirect.com/science/article/pii/S0925231216300479.

Wang, Lu Yang and George J. Augustine (Jan. 2015). "Presynaptic nanodomains: A tale of two synapses." In: *Frontiers in Cellular Neuroscience* 8 (JAN), pp. 1–10. ISSN: 16625102. DOI: 10.3389/fncel.2014.00455.

Wang, Yaning (2007). "Derivation of various NONMEM estimation methods." In: *Journal of Pharmacokinetics and Pharmacodynamics* 34 (5), pp. 575–593. ISSN: 1567567X. DOI: 10.1007/s10928-007-9060-6.

Wieland, Franz Georg, Adrian L. Hauber, Marcus Rosenblatt, Christian Tönsing, and Jens Timmer (2021). "On structural and practical identifiability." In: *Current Opinion in Systems Biology* 25, pp. 60–69. ISSN: 24523100. DOI: 10.1016/j.coisb.2021.03.005.

Wong, Jian Cheng, Abhishek Gupta, Chin Chun Ooi, Pao-Hsiung Chiu, Jiao Liu, and Yew-Soon Ong (2025). *Evolutionary Optimization of Physics-Informed Neural Networks: Survey and Prospects*. arXiv: 2501.06572 [cs.NE]. URL: https://arxiv.org/abs/2501.06572.

Wörtwein, Torsten, Nicholas B. Allen, Lisa B. Sheeber, Randy P. Auerbach, Jeffrey F. Cohn, and Louis-Philippe Morency (2023). "Neural Mixed Effects for Nonlinear Personalized Predictions." In: *Proceedings of the 25th International Conference on Multimodal Interaction*. ICMI '23. Paris, France: Association for Computing Machinery, 445–454. ISBN: 9798400700552. DOI: 10.1145/3577190.3614115. URL: https://doi.org/10.1145/3577190.3614115.

Xenarios, Ioannis, Lukasz Salwínski, Xiaoqun Joyce Duan, Patrick Higney, Sul-Min Kim, and David Eisenberg (2002). "DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions." In: *Nucleic Acids Research* 30.1, pp. 303–305. ISSN: 0305-1048. DOI: 10.1093/nar/30.1.303. eprint: https://academic.oup.com/nar/article-pdf/30/1/303/9901064/300303.pdf. URL: https://doi.org/10.1093/nar/30.1.303.

Xia, Zhengui and Daniel R. Storm (2005). "The role of calmodulin as a signal integrator for synaptic plasticity." In: *Nature Reviews Neuroscience* 6 (4), pp. 267–276. ISSN: 1471003X. DOI: 10.1038/nrn1647.

Xiong, Yunyang, Hyunwoo J. Kim, and Vikas Singh (2019). "Mixed Effects Neural Networks (MeNets) With Applications to Gaze Estimation." In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7735–7744. DOI: 10.1109/CVPR.2019.00793.

Xu, Hao, Yujun Sun, Xinjie Wang, Yao Fu, Yunfei Dong, and Ying Li (2014). "Nonlinear Mixed-Effects (NLME) Diameter Growth Models for Individual China-Fir (Cunninghamia lanceolata) Trees in Southeast China." In: *PLOS ONE* 9.8, pp. 1–10. DOI: 10.1371/

journal.pone.0104012. URL: https://doi.org/10.1371/journal.pone.0104012.

Yaduvanshi, Shivani, Rya Ero, and Veerendra Kumar (2021). "The mechanism of complex formation between calmodulin and voltage gated calcium channels revealed by molecular dynamics." In: *PLoS ONE* 16 (10 October). ISSN: 19326203. DOI: 10.1371/journal.pone.0258112.

Zhabotinsky, Anatol M., R. Nicholas Camp, Irving R. Epstein, and John E. Lisman (2006). "Role of the neurogranin concentrated in spines in the induction of long-term potentiation." In: *Journal of Neuroscience* 26.28, pp. 7337–7347. ISSN: 02706474. DOI: 10.1523/JNEUROSCI.0729-06.2006.

Zhao, Qiyuan and Brett M. Savoie (2021). "Simultaneously improving reaction coverage and computational cost in automated reaction prediction tasks." In: *Nature Computational Science* 1 (7), pp. 479–490. ISSN: 26628457. DOI: 10.1038/s43588-021-00101-3.

Zhong, Ling, Tiffani Cherry, Christine E. Bies, Matthew A. Florence, and Nashaat Z. Gerges (2009). "Neurogranin enhances synaptic strength through its interaction with calmodulin." In: *EMBO Journal* 28 (19), pp. 3027–3039. ISSN: 02614189. DOI: 10.1038/emboj.2009.236.

Zhong, Ling and Nashaat Z. Gerges (2012). "Neurogranin targets calmodulin and lowers the threshold for the induction of long-term potentiation." In: *PLoS ONE* 7 (7). ISSN: 19326203. DOI: 10.1371/journal.pone.0041275.

Zhu, Ciyou, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal (1997). "Algorithm 778: L-BFGS-B." In: *ACM Transactions on Mathematical Software* 23.4, pp. 550–560. ISSN: 0098-3500. DOI: 10.1145/279232.279236.