



**University of
Nottingham**
UK | CHINA | MALAYSIA

Modelling drug binding to biological ion channels

Thesis submitted for the degree of
Doctor of Philosophy

Frankie Patten-Elliott

20495234

Supervised by

**Gary R. Mirams
Simon P. Preston
Richard D. Wilkinson
Chon Lok Lei**

School of Mathematical Sciences

University of Nottingham

April 2026

Abstract

Ion channels are proteins that play a crucial role in many biological functions. In a healthy heart, ion channels in cardiac muscle cells ensure the heart pumps in a regular, coordinated manner. Pharmaceutical drug compounds are prone to binding to cardiac ion channels, disrupting healthy cardiac function and sometimes leading to the onset of cardiac arrhythmias. During drug development, significant time and money are spent on cardiac safety testing to avoid these potentially fatal side-effects. Much consideration is, therefore, given to improving cardiac safety testing methods to reduce uncertainty in risk predictions, while minimising time and cost.

Mathematical models can provide key insights into the underlying mechanisms that define complex biological systems. In the field of cardiac electrophysiology, models of ion channel gating and drug binding mechanisms can be effective tools for predicting drug-induced proarrhythmic risk. In this thesis, we consider methods to improve models of drug binding mechanisms, with a specific focus on binding in the human Ether-à-go-go-Related Gene (hERG) channel.

Recent technological advances have enabled the collection of high-frequency electrophysiology data that can be used to calibrate models of drug-channel binding. However, careful consideration must be given to ensure the col-

lected data are sufficiently information-rich to discriminate between different proposed models of binding mechanisms. We present an approach that produces optimal experimental designs to aid model discrimination, thereby uncovering drug-specific mechanistic insights and assisting in cardiac risk assessment. By developing improved model-fitting methods, we also overcome some of the limitations introduced by experimental artefacts in the collected data, which, if left unaccounted for, can bias model fits and predictions.

We then investigate Machine Learning (ML) methods to learn model equations directly from data. These methods allow greater modelling flexibility by significantly extending the scope of potential model structures. We demonstrate that the effectiveness of the considered ML methods is dependent on sufficiently informative data and ample prior knowledge of the missing dynamics, suggesting that combining ML methods with experimental design techniques could prove more successful.

Acknowledgements

I am grateful to everyone who has supported me in some way over the past three and a half years. Firstly, to my supervisors, Gary, Simon, Richard, and Chon, I greatly appreciate the time, energy, and wealth of knowledge and wisdom imparted. This thesis would obviously not exist without your help and encouragement.

I would also like to thank everyone at the Victor Chang Cardiac Research Institute for making me feel so welcome during my two visits. In particular, Adam, Atif, and Monique, whose help in collecting the data for this thesis was invaluable.

To the rest of the Nottingham-based Cardiac Modelling group, past and present, thanks for providing helpful feedback and support. Much of the work in this thesis has been influenced by the many engaging conversations had with you all.

To my Nottingham friends, thank you for all the fun times and for making me feel at home in this city. And to all my friends from further afield, thanks for checking in and encouraging me from afar.

Lastly, thank you so much to my family, especially Mum, Dad, and Gilbert, for all the love and support from the other side of the world. Love you guys!

Frankie

Contents

Abstract	i
Acknowledgements	iii
List of figures	vii
Abbreviations	xii
Chapter 1 Introduction	1
1.1 Background and motivation	1
1.2 Thesis structure	6
Chapter 2 Literature review	8
2.1 Cardiac electrophysiology	9
2.2 Mathematical modelling	20
2.3 Model calibration	30
2.4 Optimal experimental design	39
2.5 Model selection	40
2.6 Model discovery	44
Chapter 3 Improving robustness to model discrepancy	
in binding model calibration	52
3.1 Introduction	53
3.2 Mathematical models	54
3.3 Fitting binding models to data	57
3.4 Fitting in the presence of model discrepancy	60
3.5 A normal ratio log-likelihood	64
3.6 Physiologically-realistic model discrepancy	66
3.7 Discussion	71

Chapter 4	Developing an OED methodology to assist in binding model selection	74
4.1	Introduction	75
4.2	Mathematical models	77
4.3	Introducing the OED methodology	79
4.4	Applying the OED methodology	93
4.5	Discussion	101
Chapter 5	Determining unknown drug binding dynam- ics using an OED methodology	104
5.1	Introduction	105
5.2	Initial data collection	107
5.3	Model fitting with initial data	116
5.4	Optimising protocols	121
5.5	Optimised protocol data collection	123
5.6	Model fitting with optimised protocol data	125
5.7	Results	126
5.8	Discussion	141
Chapter 6	Learning binding dynamics from data using neural networks and sparse regression	148
6.1	Introduction	149
6.2	Learning missing dynamics in the Lotka–Volterra system . .	151
6.3	Learning missing drug-binding dynamics	164
6.4	Optimisation methods	179
6.5	Discussion	184
Chapter 7	Discussion	188
7.1	Key contributions	189
7.2	Open questions and future directions	191
7.3	Concluding remarks	194

Appendices	195
Appendix A Model equations	196
A.1 hERG models	196
A.2 Drug binding models	198
Appendix B Supplementary material for Chapter 4	206
Appendix C Supplementary material for Chapter 5	216
Appendix D Supplementary material for Chapter 6	222
Bibliography	224

List of Figures

2.1	The cardiac action potential.	10
2.2	Electrophysiological effects of I_{Kr} block.	15
2.3	Schematic of manual and automated whole-cell patch clamp experiments.	17
2.4	Protocol designed to elucidate drug binding dynamics in sodium channels.	19
2.5	Example Markov diagrams of increasing model complexity. .	21
2.6	Drug binding models.	28
2.7	Issues with model selection in the presence of model discrep- ancy.	43
2.8	An example of a neural network structure.	49
3.1	Physiological hERG model with kinetic block component. . .	56
3.2	Synthetic data generated using the conductance block model.	57
3.3	Model and parameter fits when fitting the conductance block model to $x(t)$	59
3.4	Log-likelihood heatmaps for the conductance block model. .	59
3.5	Model and parameter fits when fitting the kinetic model of block to $x(t)$	60
3.6	Synthetic data generated using the conductance block model and a piecewise discrepancy function.	61
3.7	Model and parameter fits when fitting the conductance block model to $x(t)$ in the presence of model discrepancy.	62

3.8	Log-likelihood heatmaps for the conductance block model in the presence of model discrepancy.	62
3.9	Model and parameter fits when fitting the conductance block model to $z(t)$ in the presence of model discrepancy.	63
3.10	Model and parameter fits when using a normal ratio log-likelihood to fit the conductance block model to $z(t)$ in the presence of model discrepancy.	65
3.11	Synthetic data generated using the kinetic model of block. .	67
3.12	Model and parameter fits when fitting the kinetic model of block to $x(t)$ in the presence of model discrepancy.	68
3.13	Log-likelihood heatmaps for the kinetic model of block in the presence of model discrepancy.	68
3.14	Model and parameter fits when fitting the kinetic model of block to $z(t)$ in the presence of model discrepancy.	69
3.15	Model and parameter fits when using a normal ratio log-likelihood to fit the kinetic model of block to $z(t)$ in the presence of model discrepancy.	70
4.1	Markov diagrams.	77
4.2	Synthetic verapamil data.	80
4.3	Model fits to synthetic verapamil data.	84
4.4	Log-likelihoods for model fits to synthetic verapamil data. .	85
4.5	Synthetic verapamil data under an optimised protocol. . . .	89
4.6	Model fits to synthetic verapamil data under an optimised protocol.	90
4.7	Model parameter comparisons.	91
4.8	A comparison of modelled hERG control currents.	92
4.9	Log-likelihoods for model fits to synthetic verapamil data with hERG model discrepancy.	93

4.10	Verapamil log-likelihood heatmaps.	95
4.11	Drug binding model nesting.	96
4.12	Examples of binding dynamics.	98
4.13	Bepridil log-likelihood heatmaps.	99
4.14	Dofetilide log-likelihood heatmaps.	100
5.1	Layout of drugs on SyncroPatch plate.	110
5.2	Protocol procedure.	111
5.3	Example of leak correction.	113
5.4	Example of sweep-to-sweep drift artefact.	115
5.5	Beattie hERG model fits to staircase data.	119
5.6	Design space for protocol optimisation.	122
5.7	Optimised protocols.	123
5.8	Fitting drug binding models to bepridil block data collected under the Milnes protocol and an optimised protocol.	127
5.9	Poor fits to bepridil data collected under the Milnes protocol and an optimised protocol.	128
5.10	Best trapping and non-trapping model fits to bepridil data collected under the Milnes protocol and an optimised protocol.	131
5.11	Fitting drug binding models to quinidine block data col- lected under the Milnes protocol and an optimised protocol.	133
5.12	Poor fits to quinidine data collected under the Milnes pro- tocol and an optimised protocol.	134
5.13	Best trapping and non-trapping model fits to quinidine data collected under the Milnes protocol and an optimised protocol.	136
5.14	Best trapping and non-trapping model fits to verapamil data collected under the Milnes protocol and an optimised protocol.	137
5.15	Best trapping and non-trapping model fits to diltiazem data collected under the Milnes protocol and an optimised protocol.	139

5.16	Best trapping and non-trapping model fits to chlorpromazine data collected under the Milnes protocol and an optimised protocol.	140
5.17	Summary of drug-binding model discrimination results. . . .	142
6.1	Fitting a UDE to a fully observed Lotka–Volterra system. . .	152
6.2	Extrapolating a UDE fitted to a fully observed Lotka–Volterra system.	155
6.3	Using SINDy to learn missing dynamics for the Lotka–Volterra system.	159
6.4	Fitting a UDE to a partially observed Lotka–Volterra system.	162
6.5	Markov diagram of the Beattie hERG model with a drug-binding component representing binding model 2i.	165
6.6	Fitting a UDE to drug block data.	166
6.7	Assessing the quality of a UDE fit with test drug block data.	169
6.8	Comparisons between a fitted UDE and the data-generating dynamics.	170
6.9	Using SINDy to learn missing drug-binding dynamics.	172
6.10	Fitting a UDE to drug block data with no prior knowledge of state-binding preference.	176
6.11	Assessing the quality of a UDE fit to test drug block data with no prior knowledge of state-binding preference.	176
6.12	Using SINDy to learn missing drug-binding dynamics with no prior knowledge of state-binding preference.	178
6.13	Demonstrating functional identifiability issues.	179
B.1	Synthetic verapamil data using the Lu hERG model.	206
B.2	Binding model fits with hERG model discrepancy.	207
B.3	Synthetic verapamil data under an optimised protocol and the Lu hERG model.	208

B.4	Binding model fits to optimised protocol data with hERG model discrepancy.	209
B.5	Model parameter comparisons with hERG model discrepancy.	210
B.6	Chlorpromazine log-likelihood heatmaps.	211
B.7	Verapamil log-likelihood heatmaps for Lu hERG model syn- thetic data.	212
B.8	Bepridil log-likelihood heatmaps for Lu hERG model syn- thetic data.	213
B.9	Chlorpromazine log-likelihood heatmaps for Lu hERG model synthetic data.	214
B.10	Dofetilide log-likelihood heatmaps for Lu hERG model syn- thetic data.	215
C.1	Post-processed currents in wells that pass QC.	217
C.2	Post-processed currents in wells that manually fail QC. . . .	218
C.3	Beattie hERG model fits to staircase data (slow drug plate).	218
C.4	Optimised protocols.	219
C.5	Fitting drug binding models to verapamil block data col- lected under the Milnes protocol and an optimised protocol.	219
C.6	Fitting drug binding models to diltiazem block data collected under the Milnes protocol and an optimised protocol. . . .	220
C.7	Fitting drug binding models to chlorpromazine block data collected under the Milnes protocol and an optimised protocol.	221
D.1	Fitting a UDE to a fully observed Lotka–Volterra system with no noise.	222
D.2	Extrapolating a UDE fitted to a fully observed Lotka–Volterra system with no noise.	223

Abbreviations

AIC Akaike Information Criterion.

CiPA Comprehensive in Vitro Proarrhythmia Assay.

CMA-ES Covariance Matrix Adaptation Evolution Strategy.

DMSO Dimethyl Sulfoxide.

hERG human Ether-à-go-go-Related Gene.

i.i.d. independent and identically distributed.

LASSO Least Absolute Shrinkage and Selection Operator.

ML Machine Learning.

MLE Maximum Likelihood Estimate.

ODE Ordinary Differential Equation.

OED Optimal Experimental Design.

PDF Probability Density Function.

PINTS Probabilistic Inference on Noisy Time-Series.

QC Quality Control.

RMSD Root-Mean-Square Difference.

SINDy Sparse Identification of Non-linear Dynamics.

UA Universal Approximator.

UDE Universal Differential Equation.

UQ Uncertainty Quantification.

Chapter 1

Introduction

Preamble

This chapter describes the background and motivation for the work in this thesis. While I will provide lay explanations of some of the important mathematical and biological concepts here, the subsequent literature review in Chapter 2 will provide more in-depth descriptions. At the end of this chapter, I will also provide an outline of the overall thesis structure.

1.1 Background and motivation

Mathematical models of biological processes allow us to describe complex dynamics in the form of computable equations, ultimately allowing us to make qualitative and quantitative predictions, while improving our understanding of the underlying mechanisms involved. From predicting the spread of epidemics, to understanding cancer growth, mathematical models have been successfully applied over the last century in a diverse range of bi-

ological fields. One such particularly relevant field is that of cardiac electrophysiology, a branch of cardiology focusing on the electrical activities of the heart. Understanding this electrical activity is crucial to ensure that we can predict irregular behaviour and design appropriate treatments. Cardiac arrhythmias, the broad class of disorders covering all irregular heart rhythms, affect millions of people around the world (Khurshid et al., 2018), and in many cases can lead to long-term health complications or death (Huikuri et al., 2001; Kingma et al., 2023). Cardiac ion channels, proteins that form pores in the membranes of heart muscle cells (*cardiomyocytes*), play an important role in controlling the heart’s rhythm. By allowing charged ions to flow in and out of the cell, ion channels induce electrical signals that initiate the heart’s contractions. Cardiac ion channels are prone to unintended blockage by pharmaceutical drug compounds, which can affect the flow of ions and, in some cases, cause arrhythmias. The human Ether-à-go-go-Related Gene (hERG) channel is one such ion channel that is particularly susceptible to blockage. Responsible for carrying the rapid delayed rectifier potassium current, I_{Kr} , the hERG channel plays a key role in controlling heart contractions.

Adverse cardiovascular side-effects represent a major driver in the discontinuation of development, or market withdrawal, of pharmaceutical compounds (Lavery et al., 2011). It is no surprise, then, that much consideration is given to cardiac safety during the drug development process (Valentin, 2010). Regulatory guidelines define specific *in vitro* assessments to determine the propensity of a drug to cause adverse effects on cardiac function. Given the important role of hERG, a particular emphasis of cardiac safety testing is placed on hERG channel inhibition — to what degree does a given compound block the flow of I_{Kr} in hERG? While answers to this question go some way to assessing the proarrhythmic risk of a com-

pound, recent updates to cardiac safety guidelines have suggested that a more nuanced approach involves asking an additional question: what are the biological mechanisms by which these drug compounds bind to hERG? A deeper mechanistic understanding of hERG binding (and hence block of I_{Kr}) can be invaluable in the drug design process, enabling more specific classifications of binding behaviours, and reducing uncertainty in drug risk assessment. The primary biological focus of this thesis is, therefore, on the underlying mechanisms involved in the blockage of the hERG channel, although many of the methods and conclusions detailed could be applied to other ion channels.

The past 65 years have seen significant developments in the detail and accuracy of mathematical models of cardiac electrophysiology. When appropriately calibrated, these models can accurately describe the underlying mechanisms dictating the conformational changes in cardiac ion channels, and how the combined effect of these mechanisms controls the heart's rhythm. Crucially, as for all mathematical modelling, this calibration relies on accurate and informative data. It follows that careful and considered design of experiments is an important, and perhaps relatively overlooked (Fink and Noble, 2009), aspect of the modelling process.

Ion channel models can be extended to include pharmacological drug effects. These drug-effect extensions, or *binding models*, focus on capturing a range of possible mechanisms by which a given compound can bind to a channel. Given the wide range of observed drug-specific binding mechanisms, the problem of accurately modelling (and predicting) drug block effect is two-fold: first, we must determine an appropriate model structure, and only then can we attempt to calibrate this model to data. The problem of determining an appropriate binding model can be approached from a *model selection* perspective. Model selection involves determining the

model, from a set of candidate models, that best explains the data. While various statistical criteria can be used to assess the comparative qualities of models with respect to some data, it is often important to ensure that the data is collected in a specific way to ensure that we can effectively discriminate between candidate models. Optimal experimental design (OED) is a broad class of methods that can be used to generate experimental designs based on some motivating objective. Given an objective of model discrimination, we can apply OED methods to generate bespoke experiments, providing informative data that can assist with both model selection and model calibration.

In vitro experiments measuring ion channel block have traditionally been slow and manual. Recent technological advances, however, have led to automated, high-throughput experiments that enable the collection of much larger quantities of data. The combination of these relatively new experimental capabilities, with carefully designed experiments, represents a promising approach in aiding drug binding model selection and calibration. We note, however, that these automated experiments often have a trade-off in data quality, with several experimental artefacts resulting in discrepancies between the true dynamics and the observed dynamics. While data post-processing can go some way to alleviating these discrepancies, they remain an important consideration when calibrating mathematical models of ion channel block.

Effective model selection relies on the assumption that the set of candidate models contains at least one model capable of representing the ‘true’ dynamics. While previous experimental observations can assist in the development of an extensive set of feasible drug binding models, it is never a guarantee that this set will contain the ‘true’ dynamics, especially given the potential for previously unobserved dynamics in new compounds in the

drug development pipeline. An alternative to model selection, that does not (necessarily) rely on a set of predefined candidate models, is the class of methods described by *model discovery*. Allowing for a greater diversity in dynamics, data-driven model discovery methods harness recent computational advances to learn model equations directly from data. Given sufficient data, machine learning techniques such as *neural networks* can be used to effectively capture the dynamics missing from a mathematical model (e.g. ion channel drug-binding dynamics).

As we have discussed, a key outcome of mathematical modelling is an improved understanding of the dynamics at play. While the complexity of neural networks makes them well-suited to modelling a wide array of dynamics, the flip-side of this complexity is a lack of interpretability; it is often difficult to gain useful mechanistic insights from a trained neural network. This raises some interesting philosophical questions regarding model parsimony. Occam’s razor, the principle of parsimony, suggests that when considering competing theories (i.e. models), the simpler explanation is to be preferred. In the context of ion channel drug binding models, is it reasonable to assume that a simpler model is preferable to a more complex model? While the answer to this question is not obvious, simpler models are certainly more useful, and, should their ability to explain the data be comparable to a complex alternative, a preference for simplicity seems sensible. For an alternative approach to neural networks, taking parsimony into account, we can turn to *sparse regression*. Sparse regression methods explicitly encode an assumption of parsimony to learn simple, interpretable models from data. Regarding model discovery, sparse regression methods can be useful on their own, or in conjunction with more complex approaches (e.g. neural networks) to aid interpretability.

1.2 Thesis structure

In **Chapter 2**, we provide a detailed introduction to the biological and mathematical background for the work that follows in this thesis. We begin by describing the electrical activity responsible for healthy heart function, and the mechanisms by which the introduction of drug compounds can lead to pathological behaviour. We also describe the experimental techniques used to collect relevant data, with reference to experimental design and artefacts. We then consider mathematical modelling and calibration techniques, before introducing the methods we propose to improve and develop models of drug binding mechanisms in cardiac ion channels.

With consideration for model discrepancy, in **Chapter 3** we compare the relative merits of several approaches to fitting models of drug binding dynamics to data. We begin by providing justification for previous fitting methods, before suggesting an extension to these methods that more accurately accounts for error in the data. We then show, via synthetic data examples, how this extension can improve the robustness of the fitting method when there is a discrepancy between the data-generating model and the model being used to fit the data.

In **Chapter 4**, we apply the fitting methods from Chapter 3, in conjunction with OED techniques, to develop a methodology for designing experiments that optimally discriminate between a set of candidate models of drug binding dynamics. We demonstrate the effectiveness of this method via a synthetic data study. Once again, we consider the impact of model discrepancy, noting that a significant discrepancy can reduce the accuracy of the method.

We then apply the OED procedure to real experimental data in **Chapter 5**.

To address the previously flagged model discrepancy limitations, we ensure that the model of ion channel dynamics is well-calibrated before fitting binding models and generating experimental designs. For several example drug compounds, the OED method proves effective in uncovering insights into drug-specific binding mechanisms. While the method shows promise as a tool for drug safety assessment, it is suggested that a further iteration of the experimental design process is necessary to verify our results and further discriminate between remaining feasible models.

As an alternative to the model selection problem, in **Chapter 6** we consider model discovery methods. Focusing on neural networks and sparse regression techniques, we demonstrate how these methods can be used to learn missing model dynamics from data. Although these methods are effective when sufficient data and/or prior knowledge is available, system-specific identifiability issues can limit their usability. In particular, we illustrate how a lack of informative data can prevent us from reliably learning missing drug binding dynamics from data, suggesting these model discovery methods may be more useful in combination with OED methods.

Finally, in **Chapter 7**, we discuss the key results of this thesis, detail the remaining open questions and possible future directions, before making some concluding remarks.

Chapter 2

Literature review

Preamble

In this literature review, we introduce the fundamental biological and mathematical concepts considered in this thesis. Beginning with a focus on cardiac electrophysiology, we provide descriptions of the electrical activity that controls the heart, as well as the subcellular activity governing this electrical activity, specifically, ion channels. We then consider existing mathematical models of ion channels and the mechanisms by which pharmaceutical drug compounds can bind to these channels. This is followed by an introduction to several methods for calibrating models and performing uncertainty quantification. Finally, we consider approaches focused on enhancing these mathematical models, either by improving the data-collection process, or through model selection or model discovery methods.

2.1 Cardiac electrophysiology

The field of cardiac electrophysiology broadly covers the study of the electrical activity of the heart. A heartbeat is controlled by coordinated waves of electrical activity (Pollock and Makaryus, 2022). Various diseases, and some pharmaceutical drugs, can disrupt this electrical activity, leading to serious complications. It is therefore critical that cardiac electrophysiological processes are well understood. At a cellular level, this electrical activity involves the propagation of electrical impulses, or action potentials, through *cardiomyocytes* (heart muscle cells) via the heart's electrical conduction system. These impulses cause the heart to contract and pump blood from the ventricles to the arteries and throughout the body.

2.1.1 The cardiac action potential

Action potentials that control cardiac function involve brief, rapid changes in the electrical potential across the cardiomyocyte membrane. Typically, they consist of a rapid shift to a more positive membrane potential (depolarisation) from some resting potential, followed by a slower return (repolarisation) back to resting potential. While this change in potential is a continuous process, a normal cardiac action potential is often described as being made up of five phases (Grant, 2009):

0. A rapid depolarisation phase responsible for propagating the cardiac impulse and triggering muscle contraction,
1. An early repolarisation phase,
2. A plateau phase,

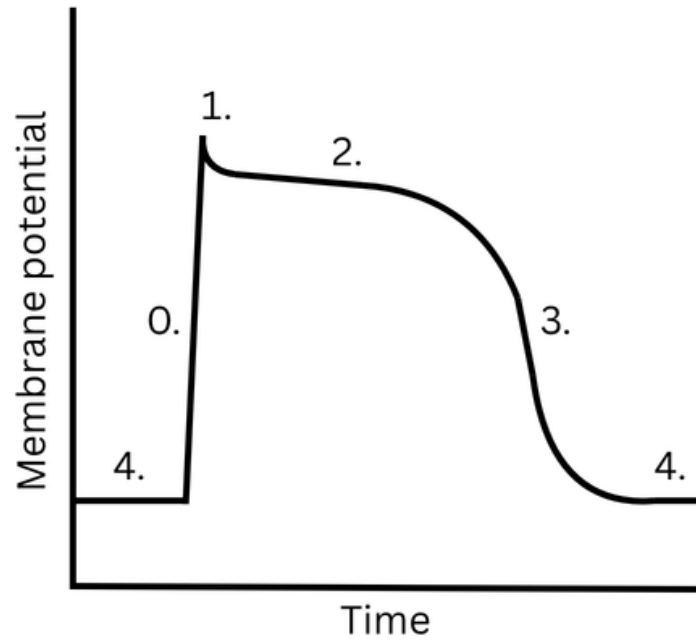


Figure 2.1: Illustration of the phases of a typical cardiac action potential in cardiomyocytes. Starting at resting potential (4.), a rapid depolarisation (0.) is followed by early repolarisation (1.) and a plateau phase (2.), before a rapid repolarisation phase (3.) restores the membrane to resting potential (4.).

- 3. A rapid repolarisation phase that restores the membrane to resting potential,
- 4. A phase at resting potential, typically around -90 mV .

In Figure 2.1 we visualise these phases of the action potential. We illustrate a typical action potential and approximately indicate where each of the phases (numbered 0 to 4) take place. The phases of the action potential are dictated by the flow of ions across the membrane, and this ion flow, or current, is controlled by ion channels.

2.1.2 Ion channels

Ion channels are proteins that form pores in the cell membrane through which ions can flow in and out of the cell (Priest and McDermott, 2015).

A difference in concentration and charge across the membrane results in an electrochemical gradient that drives the flow of ions through these channels. This flow can then be controlled by the diverse gating and permeability properties of ion channels (Grant, 2009). Gating refers to the mechanism by which a channel transitions between various conformational states (e.g. open/closed). Voltage-dependent gating, a common mechanism of ion channel gating, involves changes in channel conductance in response to shifts in membrane potential (Hille, 2001). Other gating mechanisms include ligand-gating or second messengers (intracellular signalling molecules). Permeability, on the other hand, refers to the movement of specific ions through the channel when in an open state. For example, sodium channels are so-called due to their selective permeability for sodium ions (Na^+).

While ion channels represent a passive form of ion transporter, using the electrochemical gradient as a driver of diffusion, it is worth noting that there also exist other forms of transporters, all of which contribute to maintaining homeostasis. Ion pumps are active transporters that use energy from sources such as adenosine triphosphate (ATP) to move ions across the membrane against the electrochemical gradient, while ion exchangers simultaneously move ions against the gradient in exchange for the movement of a second type of ion along this gradient.

In the heart, in the membranes of cardiomyocytes, there are several types of ion channels expressed by different genes. By controlling the flow of various ions, cardiac ion channels play a fundamental role in regulating the cardiac action potential. In fact, the change in membrane potential over time (dV_m/dt) depends directly on the sum of all ion channel currents in

the membrane ($\sum_j I_j$) and can be described by the following relationship

$$\frac{dV_m}{dt} = -C_m \sum_j I_j, \quad (2.1)$$

where C_m is the membrane conductance. Intracellular channels, or gap junctions, ensure that neighbouring cardiomyocytes are electrically coupled, and this coupling then allows the action potential to spread from cell to cell, ultimately resulting in a heartbeat (Zhao et al., 2019).

2.1.3 The hERG channel

The $K_{v11.1}$ channel, encoded by the human Ether-à-go-go-Related Gene (hERG), is responsible for carrying the rapid delayed rectifier potassium current (I_{Kr}) in cardiomyocytes (Sanguinetti et al., 1995). This voltage-gated potassium (K^+) channel plays a key role in controlling the phase 3 repolarisation of the cardiac action potential, allowing the cardiomyocyte to recover from excitation (Grant, 2009; Vandenberg et al., 2012; Priest and McDermott, 2015). A single hERG channel can transition between open, closed or inactivated states depending on changes in the transmembrane voltage (Sanguinetti and Tristani-Firouzi, 2006). While both closed and inactivated states prevent the flow of ions in the channel, an inactivated state is distinct in that the ion block is generally caused by some mechanism other than the channel closing. Typically, in hERG, membrane depolarisation leads to a slow activation (opening) of the channel while the channel rapidly inactivates on repolarisation (Vandenberg et al., 2001).

It is known that the hERG channel is susceptible to block by pharmaceutical small molecules (referred to here as ‘compounds’ throughout). This blockage can result in a reduction in I_{Kr} . Given the relationship expressed

in Equation (2.1), a decrease in I_{Kr} results in a lengthening of the cardiac action potential, which can frequently lead to an increased risk of cardiac arrhythmia (Heist and Ruskin, 2010; Mirams et al., 2012; Hill et al., 2016). For this reason, the hERG channel is of particular relevance when it comes to cardiac safety testing of pharmaceutical compounds.

2.1.4 Cardiac safety testing

Pharmaceutical compounds can enter cardiac ion channels and bind to various binding sites within the channel, disrupting the normal flow of ions. Some drugs are specifically designed to block certain channels to treat conditions such as arrhythmia or high blood pressure (Bakris et al., 1996; Lipton, 2004). However, there also exist drugs, both cardiac and otherwise, for which channel block is an unintended side-effect. In fact, many prescription drugs, including some antihistamines, antibiotics and antipsychotics, have been shown to increase the risk of arrhythmia via hERG block (Roden, 2004). For this reason, it is important to establish any potential drug side-effects during the preclinical pharmaceutical development stages. International drug safety regulations ensure that all new drug compounds are subject to extensive safety testing prior to any human trials. However, current cardiac safety tests are limited in their mechanistic insights and often have a high false positive rate, preventing potentially safe drugs from reaching the market.

The Comprehensive in Vitro Proarrhythmia Assay (CiPA) initiative was proposed in 2013 with the intention of replacing current guidelines for cardiac safety testing during drug development (Sager et al., 2014; Fermini et al., 2016; Gintant et al., 2016). Harnessing advances in computational modelling, the initiative was designed to improve the evaluation of proar-

rhythmic risk via an enhanced mechanistic understanding of drug binding in cardiac ion channels. This work has resulted in a Q&A document by the ICH E14/S7B Implementation Working Group (2022) suggesting cardiac safety testing guideline amendments. Given its key role in controlling the phase 3 repolarisation of the action potential, and its proclivity for being an off-target of many types of pharmaceuticals, the hERG channel has consistently been a focus of cardiac safety testing guidelines and is also a key consideration of the CiPA initiative.

2.1.5 Drug binding in the hERG channel

In the hERG channel, which is particularly promiscuous when it comes to drug binding, drug block prevents the flow of K^+ ions, reducing I_{Kr} (Kalyaanamoorthy and Barakat, 2018). In Figure 2.2, we include a schematic from Mirams et al. (2012) demonstrating how a 50% block in I_{Kr} leads to a lengthening of the action potential, ultimately affecting the whole heart rhythm. The electrocardiogram (ECG) signal in the bottom plot indicates a prolongation of the QT interval (time between the start of the Q wave and end of the T wave), which is associated with an increased risk of developing abnormal heart rhythms and sudden cardiac death (Panoulas et al., 2014).

The primary binding site in the hERG channel is the central cavity (Mitcheson, 2008). Given the molecular structure of the channel, it has been suggested that compounds only bind to the channel when it is not in a closed state (Butler et al., 2020). It has been observed that the mechanism by which compounds bind to the hERG channel can be compound-specific (Redfern et al., 2003; Hancox et al., 2008; Perry et al., 2010). One such example of this is the propensity of a compound to become *trapped* (unable

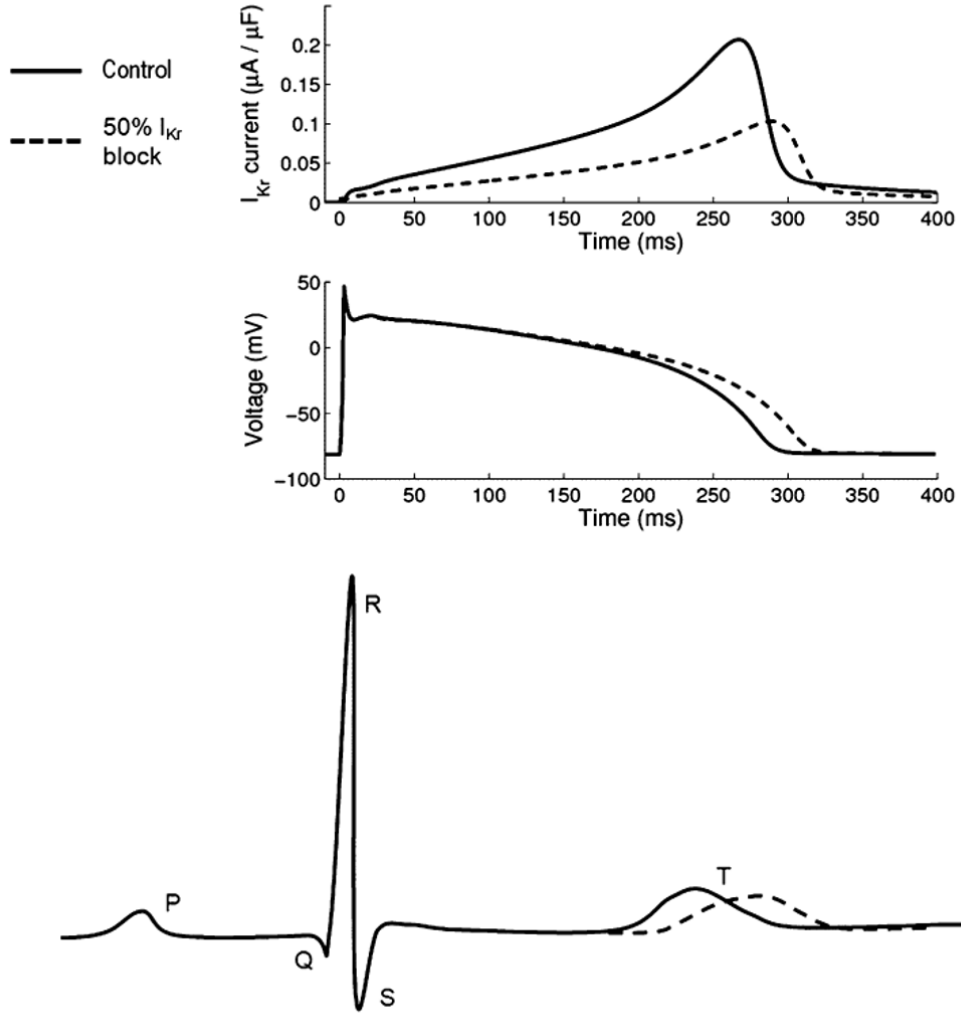


Figure 2.2: Illustration from Mirams et al. (2012) indicating the electrophysiological effects of a 50% block in I_{Kr} . Top: hERG current, Middle: action potential, Bottom: ECG. The top and middle plots are simulations of the model from Grandi et al. (2009), while the bottom is simply a schematic. The dashed lines in each plot indicate the effect of a 50% block in I_{Kr} relative to control (solid lines).

to unbind) when the channel closes. Some compounds, such as bepridil and dofetilide, appear to become trapped inside the central hERG cavity, remaining bound when the channel closes, while others, such as cisapride and verapamil, unbind (Zhang et al., 1999; Kamiya et al., 2006; Stork et al., 2007; Windisch et al., 2011; Windley et al., 2017). It has also been theorised that some drugs bind preferentially to certain channel states over others or, in the extreme case, bind to a particular state only (Starmer and Courtney, 1986; Hondeghem, 1987; Starmer et al., 1990; Carmeliet and Mubagwa, 1998; Thouta et al., 2018). It is worth noting that several other factors, such as temperature, pH, and combinations of drugs, have also been observed to have an influence on drug binding dynamics (Brennan et al., 2009).

2.1.6 Voltage clamp experiments

The transmembrane current of a cell can be measured (*in vitro*) in response to the membrane potential via a voltage clamp experiment, where a function defining the transmembrane voltage, V , dependent on time, t , is applied. We refer to this function, $V(t)$, as a *voltage protocol*. Much of what is known about ion channels in cell membranes comes from voltage clamp experiments (Halliwell et al., 1994). While the specifics of a voltage clamp experiment setup can vary, generally it involves placing the tip of a pipette onto the cell membrane and applying a small amount of negative pressure to create a high-resistance seal, isolating a section of the membrane (Kornreich, 2007). A voltage clamp amplifier can then be used to apply a command voltage, defined by $V(t)$, across the isolated membrane section, and the resulting current that flows out across the membrane can then be measured by the amplifier. Variations of voltage clamp experiments were

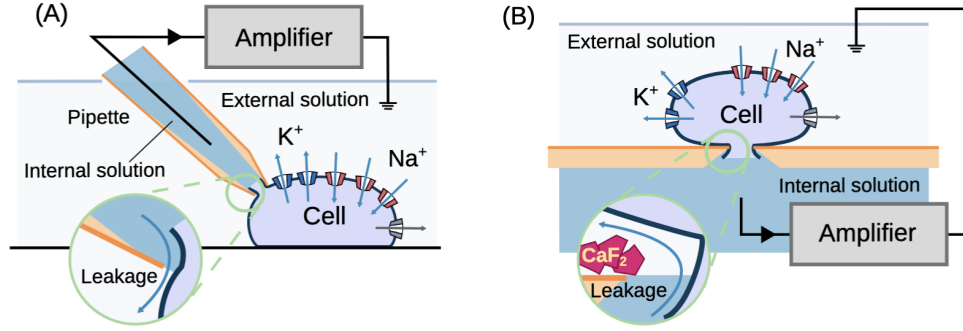


Figure 2.3: Schematic of manual (A) and automated (B) whole-cell patch clamp experiments adapted from Lei et al. (2021). In manual experiments, a glass pipette is placed on the cell, and suction is applied. Automated experiments instead involve the cell being placed on a planar surface with a micro-pore. In both cases, an amplifier, indicated by a grey box, applies a voltage across the cell membrane. The bottom-left insets of each subplot indicate the potential for leak currents to escape the high-resistance seal. To improve the seal in the automated case, a seal-enhancing compound, such as CaF_2 , is often applied.

introduced by Neher and Sakmann (1976), leading to methods such as the ‘whole-cell patch clamp’. In these whole-cell experiments, the isolated section of membrane is ruptured with a second burst of suction, enabling both the extracellular and intracellular potentials to be measured, ultimately allowing the cell’s transmembrane voltage to be controlled. While traditional manual voltage clamp experiments are performed by hand on a single cell, recent technological developments have allowed for high-throughput automated voltage clamp experiments that significantly decrease the time required to collect large amounts of voltage clamp data. Many recent studies have utilised these automated voltage clamp techniques to study cardiac electrophysiology (Li et al., 2017; Lei et al., 2019a,b; Kozek et al., 2020). In Figure 2.3, we include schematics, adapted from Lei et al. (2021), of manual and automated whole-cell patch clamp procedures. In the manual case (A), a pipette is applied to the cell to create the high-resistance seal, while in the automated case (B), the cell is suspended on a micro-pore on a planar surface. In both cases, the seal created is imperfect, resulting in

leakage currents as indicated in the bottom-left insets of both subplots. We will discuss this experimental artefact, among others, in the following section.

2.1.7 Voltage clamp experimental artefacts

While voltage clamp experiments have been the gold standard for studies of cell electrophysiology since their invention (Sakmann and Neher, 1984), the techniques are imperfect, resulting in several experimental artefacts that pollute the measurements of transmembrane current. As illustrated in Figure 2.3, one of these artefacts results from leak currents escaping the high-resistance seal. These leak currents are particularly prevalent in automated patch clamp experiments, where the seal created is often not as tight as in manual experiments. Seal-enhancing compounds (e.g. CaF_2) can go some way to reducing these leak currents in the automated setup, but these seal-enhancers are prone to introducing further side-effects (Lei et al., 2019b). Other artefacts can arise as a result of voltage offsets, cell or pipette capacitances, and series resistance between the internal solution electrode and the cell membrane (Lei et al., 2025). While all of these experimental errors can be partially corrected for in post-processing, the remaining artefacts represent a limitation of these voltage clamp methods. The presence of these artefacts must, therefore, be taken into account when attempting to draw mechanistic insights from data collected via voltage clamp experiments.

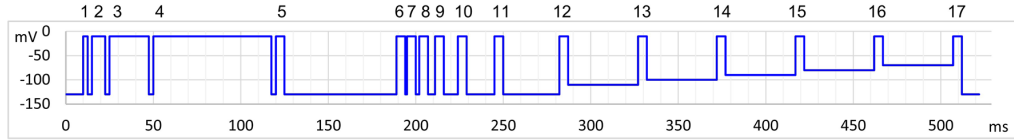


Figure 2.4: Reproduction of a plot from Lukacs et al. (2021) of a protocol designed to elucidate drug binding dynamics in sodium channels. The protocol indicates the changing membrane potential over a 522 ms time period.

2.1.8 Voltage protocol design

An important consideration when performing voltage clamp experiments is the form of the applied voltage protocol: the time-dependent voltage waveform applied as a forcing function to the system. Different combinations of steps and ramps in the applied voltage can emphasise different ion channel gating or drug-binding mechanisms, improving our understanding of these processes. For example, Hodgkin and Huxley (1952) were able to approximate the time constants and steady states of ion channel gating processes by designing simple protocols involving a change in voltage from resting potential to induce differing depolarisations.

Regarding drug-channel interactions, several protocols have been designed to determine specific kinetics. As part of the CiPA initiative, Li et al. (2017) used a modified version of the protocol proposed by Milnes et al. (2010) to assess drug effects on hERG. The protocol was designed to capture both block development and drug trapping, and involved a simple step up from a -80 mV resting potential to 0 mV for 10 s, repeated ten times at 15 s intervals. Gomis-Tena et al. (2020) designed three related protocols to highlight the differences in block potency and preferential state binding. Named P-80, P0, and P40, each protocol involved a 5 s conditioning voltage step (at -80 , 0 or 40 mV respectively) followed by 0.2 s pulses at -60 mV from a -80 mV resting potential. In Figure 2.4, we reproduce a

plot from Lukacs et al. (2021) of a protocol designed to elucidate three aspects of voltage-dependent binding dynamics in sodium channels: how fast the drug binds upon depolarisation, trapping effects, and steady-state channel block. These three aspects were captured respectively via; depolarisation steps of progressively increasing lengths alternating with fixed length hyperpolarizations (steps 1-5 in Figure 2.4), hyperpolarisations of progressively increasing lengths alternating with fixed length depolarisations (steps 6-11), and fixed length hyperpolarisation steps of progressively increasing membrane potential alternating with fixed length and fixed voltage depolarisation steps (steps 12-17). This protocol successfully enabled the identification of several sub-processes involved in drug action in sodium channels.

2.2 Mathematical modelling

In 1952, Hodgkin and Huxley quantified the membrane current in the squid giant axon, paving the way for the widespread use of mechanistic, biophysically-based mathematical models in the field of electrophysiology (Hodgkin and Huxley, 1952). A decade later, Noble (1962) successfully adapted their approach to model the electrophysiology of cardiac cells. Noble’s work, in tandem with new biological discoveries, has laid the foundations for increasingly detailed mechanistic models of cardiac electrophysiology over the past 65 years.

2.2.1 Modelling ion channel kinetics

Models of ion channel kinetics generally focus on capturing the gating mechanisms involved. Instead of modelling every channel individually, compu-

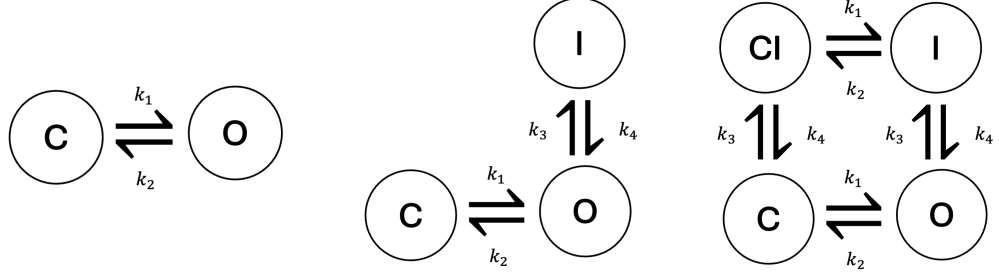


Figure 2.5: (a): A simple single-gate model. States are labelled as closed (C) and open (O). Arrows indicate the possible gating processes between states. Transition rates are defined by parameters k_1 and k_2 . (b): An additional inactivated (I) state is added, introducing gating dependence. Additional rate parameters k_3 and k_4 define transition rates between O and I states. (c) The four state (Beattie et al., 2018) model introduces a closed inactivated (CI) state.

tational expense can be reduced by modelling the proportion of the total number of channels in the membrane that are in each state over time. This approximation is usually reliable for cardiomyocytes, given that each cell contains a large number ($O(10^3)$ to $O(10^4)$) of ion channels of each type. The *Hodgkin-Huxley* formulation assumes independent gating processes, whilst *Markov* models allow for a dependence between gating processes, enabling greater flexibility to capture richer dynamics (Rudy and Silva, 2006). In Figure 2.5a, we plot a *Markov diagram* of a simple gating model with two states: closed (C) and open (O). Given that it contains a single gating process, this model can technically be described as both a Hodgkin-Huxley model and a Markov model. By defining mass-action rates at which transitions between model states occur, a set of ordinary differential equations (ODEs) can be used to mathematically describe how a collection of ion channels change state over time. The transition rates in the example two-state model from Figure 2.5a are defined by the two voltage-dependent

rates k_1 and k_2 . We can write out the ODE for this model explicitly as

$$\frac{dC}{dt} = -k_1C + k_2O, \quad (2.2)$$

$$\frac{dO}{dt} = k_1C - k_2O. \quad (2.3)$$

Given that C and O represent the proportion of channels in closed and open states, respectively, and assuming $C + O = 1$, i.e. all channels are either closed or open at any given time, we can rewrite this system as a single equation

$$\frac{dO}{dt} = k_1(1 - O) - k_2O. \quad (2.4)$$

This equation is often alternatively expressed in terms of a steady state, O_∞ , and time constant describing the rate at which this steady state is reached, τ_O

$$\frac{dO}{dt} = \frac{O_\infty - O}{\tau_O}. \quad (2.5)$$

where $O_\infty = k_1/(k_1 + k_2)$ and $\tau_O = 1/(k_1 + k_2)$. By adding further model states and transition rates, we can model more complex dynamics. In Figure 2.5b we include a Markov diagram with an additional inactivated (I) state. This Markov model can no longer be described via a Hodgkin-Huxley formulation, given the gating dependence introduced by the additional state; the channel can only transition to an inactivated state from an open state — it cannot transition directly from a closed state. We can

describe this Markov model via the following set of ODEs

$$\frac{dC}{dt} = -k_1C + k_2O, \quad (2.6)$$

$$\frac{dO}{dt} = k_1C - (k_2 + k_3)O + k_4I, \quad (2.7)$$

$$\frac{dI}{dt} = k_3O - k_4I. \quad (2.8)$$

Generally, any Markov model, with any number of states, can be defined by the generalised formulation

$$\frac{d}{dt}\mathbf{x} = \mathbf{Q}(V)^T\mathbf{x}, \quad (2.9)$$

where $\mathbf{x}$ is the vector of state variables, V is the transmembrane potential, and $\mathbf{Q}$ is the transition rate matrix such that the element $Q_{i,j}$ is the transition rate between the model's i th and j th states. For example, we can define $\mathbf{Q}$ for the model in Figure 2.5b as

$$\mathbf{Q} = \begin{bmatrix} -k_1 & k_2 & 0 \\ k_1 & -k_2 - k_3 & k_4 \\ 0 & k_3 & -k_4 \end{bmatrix}, \quad (2.10)$$

where $\mathbf{x} = [C, O, I]$. The transition rates are voltage-dependent, ensuring the model does not settle into a steady state if the membrane potential is changing. The resultant ion channel current, I , is generally calculated with the following equation

$$I = g \cdot O \cdot (V - E), \quad (2.11)$$

where g is the maximal conductance, O is the proportion of channels in the open state, and E is the Nernst potential (membrane potential at which there is no net flux because electrical and chemical forces are balanced).

In a Hodgkin-Huxley model, the O variable is modelled as the product of independent gating processes, while in a Markov model it is the sum of conducting states (usually just one, i.e. the open state in Figure 2.5b).

2.2.2 Modelling the hERG channel

Given the importance of the hERG channel in the repolarisation phase of the cardiac action potential, several different hERG models have been developed to assist in understanding hERG gating kinetics. While Hodgkin-Huxley type formulations are often sufficient for modelling action potentials, detailed examinations of hERG kinetics have led to the adoption of Markov models to accurately capture the unique gating characteristics of hERG (Bett et al., 2011). Throughout this thesis, we use the Beattie et al. (2018) model of hERG gating, which can technically be described in either a Hodgkin-Huxley or Markov formulation (Clerx et al., 2019a). In Figure 2.5c, we plot a Markov diagram of this Beattie model. The model has four states: IC (Inactive Closed), C (Closed), I (Inactive), and O (Open), where each variable represents the proportion of channels in that state. Transition rates, k_1 to k_4 , define the rates at which the channel transitions between each of the four states. The dependence of these transition rates on transmembrane voltage, V , is defined by the general equation

$$k_i = \alpha_i \exp(\beta_i V), \quad (2.12)$$

where α_i and β_i are physiological model parameters. The full set of equations describing the Beattie model are included in Appendix A.1.1. Other Markov models of hERG disagree on the number and conformation of states; the six-state model used in the CiPA initiative has additional closed and inactive states (Dutta et al., 2017; Li et al., 2017), while models such

as that used by Fink et al. (2008) have three closed states.

Modelling temperature dependence

It is widely accepted that hERG gating dynamics are temperature dependent (Dhaka et al., 2006; Yang and Zheng, 2014). Higher temperatures typically lead to faster kinetics and differing current magnitudes. Several methods have been proposed for modelling this temperature dependence. One such method is the Eyring formulation, which uses basic thermodynamics and statistical mechanics to derive an equation for the gating transition rates that depends on both voltage and temperature (Eyring, 1935). Another approach is the use of Q_{10} coefficients, which assume an exponential increase in reaction rate with temperature, and involve a simple modification of the fixed temperature binding rate formulation in Equation (2.12) (Běhrádek, 1930). In their 2019 paper on modelling hERG temperature dependence, Lei et al. (2019a) compared the typical Eyring formulation, a generalised version of the Eyring formulation, and Q_{10} coefficients with transition rate parameters fitted to hERG data collected at five distinct temperatures between 25 and 37°C. They found that the typical Eyring formulation and the Q_{10} coefficients were both unable to capture the complexity in the kinetic parameter temperature dependence reflected in the data, while the generalised Eyring formulation, which includes an additional temperature-independent voltage term, could largely capture this complexity.

2.2.3 Modelling drug binding mechanisms

When it comes to modelling mechanisms of drug block in cardiac ion channels, in the majority of cases it is assumed that the drug compound binds to the channel pore and either directly obstructs the flow of ions, or changes the channel conformation, inducing a blockage (Mirams et al., 2012). A fairly simple model of the effects of drug block on channel current is the so-called ‘conductance block’ model. This involves scaling the maximum conductance of an ion channel based on some factor, b , to give the following adjustment to Equation (2.11)

$$I = g \cdot b \cdot O \cdot (V - E). \quad (2.13)$$

It is possible to derive an explicit expression for b by considering the following reaction



where n is the number of drug compounds, D , binding to a single ion channel, C , to produce some complex, CD_n . It follows from the principle of mass-action kinetics (reaction rates are directly proportional to the product of their reactants) that we can write down the following ODE for the concentration of C

$$\frac{d[C]}{dt} = -k_{\text{on}}[C][D]^n + k_{\text{off}}[CD_n]. \quad (2.15)$$

Given that b is the proportion of unblocked channels, it follows that $b = [C]/([C] + [CD_n])$, and so we can divide Equation (2.15) by $([C] + [CD_n])$

to get the following ODE for b

$$\frac{db}{dt} = -k_{\text{on}}b[D]^n + k_{\text{off}}(1 - b). \quad (2.16)$$

When the drug block is at steady state, we will have $db/dt = 0$, which we can substitute into Equation (2.16) to get the following expression for b_{∞} , the steady state level of b

$$b_{\infty} = \frac{\frac{k_{\text{off}}}{k_{\text{on}}}}{[D]^n + \frac{k_{\text{off}}}{k_{\text{on}}}}. \quad (2.17)$$

The concentration at which we get 50% block is known as the ‘50% inhibitory concentration’ or $[IC_{50}]$. At 50% block $b_{\infty} = 0.5$ which implies that $[D]^n = k_{\text{off}}/k_{\text{on}}$, and we can consequently define the following relationship $[IC_{50}]^n = k_{\text{off}}/k_{\text{on}}$. Substituting this into Equation (2.17), we get

$$b_{\infty} = \frac{[IC_{50}]^n}{[D]^n + [IC_{50}]^n} = \frac{1}{1 + \left(\frac{[D]}{[IC_{50}]}\right)^n} \quad (2.18)$$

which are two forms of the Hill equation where n is the *Hill coefficient* (Hill, 1910; Goutelle et al., 2008).

Models of this ‘conductance block’ form do not take binding kinetics into account, and can struggle to capture more complex binding dynamics. We can instead turn to Markov model formulations, introducing additional states and rates to an existing ion channel Markov model to simulate various (compound-specific) binding mechanisms (Mirams, 2023). Lei et al. (2024) proposed a set of fifteen models that characterise different proposed candidate mechanisms for hERG drug-binding. In Figure 2.6, we plot Markov diagrams of these fifteen models. For each model, the I and O states in black are assumed to be part of some underlying hERG physiological model. Taking model 7 as an example of one of these fifteen

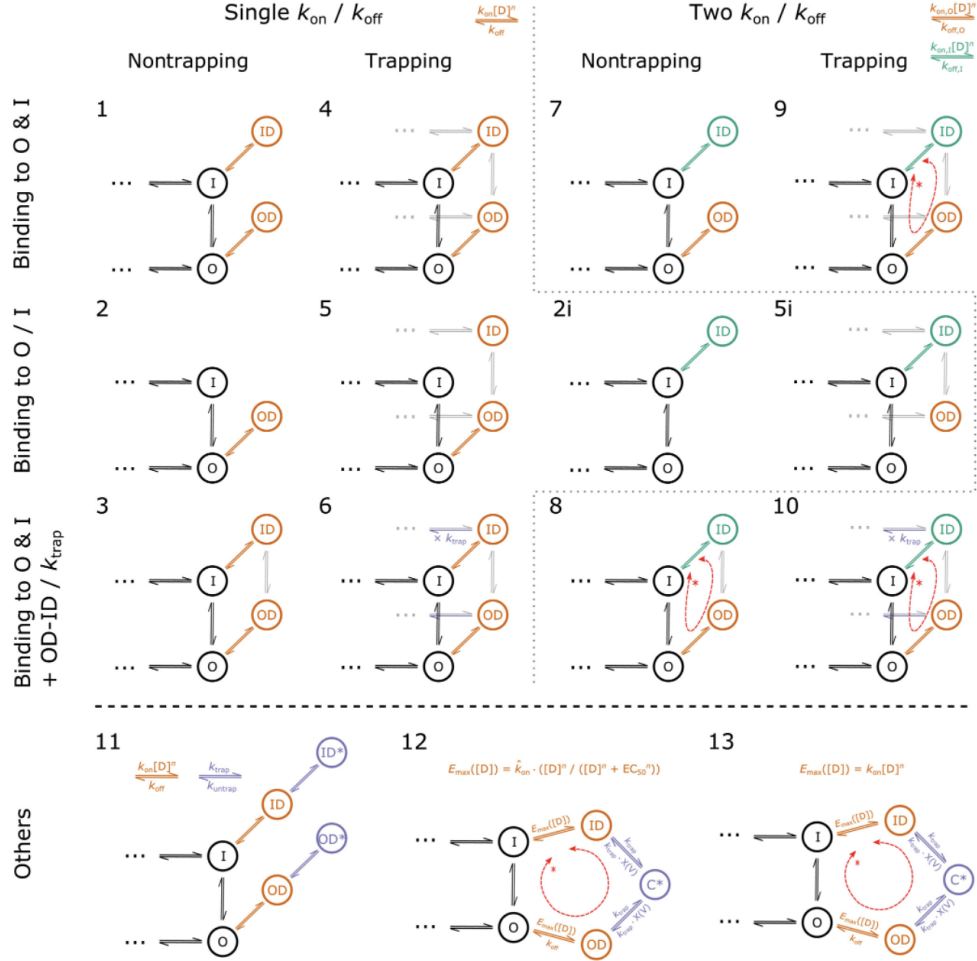


Figure 2.6: Markov diagrams of pharmacological drug binding models reproduced from Lei et al. (2024). In each diagram, the I and O states are assumed to be part of some underlying model of hERG gating. The binding models are then defined by the additional states (e.g. ID and OD) and rate parameters (e.g. k_{on} and k_{off}).

models, we have two additional states: ID (Inactive Drug-bound) and OD (Open Drug-bound). Binding rates are then described by the parameters $k_{\text{on,I}}$, $k_{\text{off,I}}$, $k_{\text{on,O}}$, and $k_{\text{off,O}}$, and the rate of on-binding is dependent on the drug concentration, $[D]$, and the Hill coefficient, n . This example model represents *non-trapping* binding behaviour as the channel cannot enter a closed state without the drug first unbinding (returning from an ID or OD state to an I or O state). In contrast, some of the fifteen binding models represent *trapping* dynamics. The trapping component either involves a ‘mirror image’ of the physiological hERG model, which allows for a channel to close and prevent unbinding from ICD or CD states (corresponding to IC and C in the drug-bound channel), or is represented by additional trapped states. In Figure 2.6, models 4, 5, 5i, 6, 9, and 10 illustrate the ‘mirror image’ trapping, while models 11, 12, and 13 include additional trapped states. These Markov models can also represent state-dependent binding, for example, models 2 and 2i allow binding from only the open or inactive states, respectively. For the full set of equations that define these binding models, see Appendix A. While this set of fifteen models covers a wide range of possible dynamics, the flexible structure of the Markov model form feasibly allows a modeller to extend or reduce these models as they see fit.

2.2.4 Numerical methods

To make use of the models of ion channel and drug binding dynamics detailed in the preceding sections, we must solve the systems of ODEs. Unless the voltage is fixed, there is generally no simple analytical solution, and so we must employ numerical methods. Early numerical methods for solving ODEs include the Euler method, derived by Leonard Euler in 1768 (Brezin-

ski and Wuytack, 2012). The Euler method, and many of the subsequently developed methods (e.g. Runge-Kutta (Butcher, 1996)), are *explicit* methods that calculate the state of the system at some future time, $\mathbf{x}(t+h)$, based on the state of the system at the current time, $\mathbf{x}(t)$. However, the ODEs describing cardiac electrophysiology are stiff (Cooper et al., 2015); the system variables change on very different timescales, leading to stability issues when using explicit numerical methods. We can instead turn to *implicit* methods that, while more computationally expensive, provide better stability. One such implicit method is CVODE, an implicit adaptive multi-step method (Hindmarsh et al., 2005). In this thesis, all simulations were performed using CVODE as implemented in Myokit (Clerx et al., 2016). The accuracy of these simulations is controlled by the specified error tolerance. Adaptive step solvers such as CVODE have absolute and relative tolerance parameters that can both be defined by the user. To ensure minimal errors in our ODE solutions, we performed tolerance checks, tightening the CVODE tolerances until the observed solution converged. As a result of these tolerance checks, and unless otherwise specified, the absolute and relative tolerances are set to 1×10^{-8} and 1×10^{-10} , respectively, throughout this thesis.

2.3 Model calibration

As previously detailed, ion channel and drug binding models generally have several parameters representing the transition rates between various channel states. To ensure these models can be used for predictive purposes, and to help elucidate the underlying mechanisms involved, it is necessary to accurately calibrate these model parameters. This can be done by fitting the models to experimental voltage clamp data. In this section, we detail

several mathematical techniques required for robust model calibration, and their applications with regard to ion channel and drug binding models.

2.3.1 Maximum likelihood estimation

Maximum likelihood estimation, a method for parameter estimation, is used extensively for statistical inference in a variety of fields (Hendry and Nielsen, 2007; Chambers et al., 2012; Ward and Ahlquist, 2018). Let $\mathbf{x} = (x_1, x_2, \dots, x_n)$ be a set of n observed data points, where $\mathbf{x}$ is a realisation of some random variable $\mathbf{X}$ with probability distribution $p_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta})$. Here $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ is an unknown parameter vector, where $\boldsymbol{\Theta}$ is the parameter space. The likelihood of $\boldsymbol{\theta}$ given $\mathbf{x}$ is known as the likelihood function and is simply defined as

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{x}) = p_{\mathbf{X}}(\mathbf{x}|\boldsymbol{\theta}). \quad (2.19)$$

We can evaluate $\mathcal{L}(\boldsymbol{\theta}|\mathbf{x})$ via joint probabilities, taking the product of the probabilities of observing each element of $\mathbf{x}$ such that

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{x}) = \prod_{i=1}^n p_{\mathbf{X}}(x_i|\boldsymbol{\theta}). \quad (2.20)$$

One way to estimate the value of $\boldsymbol{\theta}$ given the data $\mathbf{x}$ is to determine the value of $\boldsymbol{\theta}$ that maximises the likelihood function (Casella and Berger, 2002; Myung, 2003; Rossi, 2018). Given the form of Equation (2.20), optimising $\mathcal{L}(\boldsymbol{\theta}|\mathbf{x})$ is generally a computationally difficult task. The joint probability is calculated by taking the product of several (often small) probabilities, which can lead to underflow errors. Furthermore, differentiation of products (via the product rule) generally results in high computational complexity.

It is therefore common to instead use the log-likelihood where

$$\ell(\boldsymbol{\theta}|\mathbf{x}) = \ln(\mathcal{L}(\boldsymbol{\theta}|\mathbf{x})). \quad (2.21)$$

Because the log function is monotonically increasing, the $\boldsymbol{\theta}$ that maximises the log-likelihood is equal to the $\boldsymbol{\theta}$ that maximises the likelihood function. The value of $\boldsymbol{\theta}$ that maximises the log-likelihood (or equivalently the likelihood) is known as the Maximum Likelihood Estimate (MLE) and is denoted $\hat{\boldsymbol{\theta}}$, where

$$\hat{\boldsymbol{\theta}} = \arg \max_{\boldsymbol{\theta} \in \Theta} \ell(\boldsymbol{\theta}|\mathbf{x}) = \arg \max_{\boldsymbol{\theta} \in \Theta} \mathcal{L}(\boldsymbol{\theta}|\mathbf{x}). \quad (2.22)$$

2.3.2 Fitting ion channel models

Clerx et al. (2019a) detail four common approaches from the literature for fitting ion channel models to data. Of the four methods, they found that the two methods that involve fitting models directly to current traces from voltage clamp experiments performed best in terms of both predictive accuracy and computational efficiency. Specifically, these two methods involved: 1. fitting to several current data traces collected under a range of different voltage protocols, and 2. fitting to a single current data trace collected under one short, rapidly fluctuating protocol.

Moreno et al. (2016) describe how initialisations for model rate parameters can be estimated from experimental electrophysiology data. Alternatively, if available, existing parameterisations from the literature can be used as initialisations. The fitting process itself then generally involves minimising some error function representing the ‘distance’ between the fitted model and the data. This distance can often be described by a likelihood function,

which can then be maximised via some optimisation algorithm. Note here, minimising the *negative* likelihood is equivalent to minimising the distance. With an assumption of Gaussian noise on measured ion channel current data, $x(t)$, a log-likelihood function can be formulated as

$$\ell(\boldsymbol{\theta}, \sigma | x(t)) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \sum_{t \in T} (x(t) - f(\boldsymbol{\theta}))^2, \quad (2.23)$$

where f is the model with parameters $\boldsymbol{\theta}$, T is the set of time-points at which the data is recorded, and σ is the standard deviation of the Gaussian noise. Several previous papers have found that the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) (Hansen, 2006) global optimisation algorithm provides a robust method for obtaining good fits for ion channel models when maximising a likelihood function (Johnstone et al., 2016; Beattie et al., 2018).

2.3.3 Fitting drug binding models

Drug-binding models such as those illustrated in Figure 2.6 can be fitted similarly to the underlying hERG models. In their 2024 paper, Lei et al. (2024) used data from Li et al. (2017) to fit several proposed models of drug-binding dynamics. The data represented the proportion of open channels over time under several sweeps of a modified Milnes voltage protocol as described in Section 2.1.8. These data were derived by dividing I_{Kr} in the presence of a drug, by control I_{Kr} traces with no drug block effect. Model fits (i.e. parameter estimates) were then obtained by minimising a weighted sum-of-squares difference between the model and data open channel proportions. This is effectively equivalent to maximising a Gaussian likelihood function as described in Equation (2.23). We investigate the suitability of this Gaussian assumption in detail in Chapter 3. For

simplicity, the transition rates of the underlying hERG gating model were fitted to control currents first, and then fixed, before separately fitting the drug-binding parameters.

2.3.4 Identifiability analysis

Several difficulties can arise when fitting model parameters to data. In general, these issues can be described by the field of identifiability analysis. If not taken into consideration, identifiability issues can lead to incorrect parameter estimates and erroneous model predictions. Parameter identifiability can be broken down into two categories: structural and practical identifiability.

Structural identifiability

Structural identifiability relates to model structure. For some models, no matter the accuracy and completeness of the observable data, it is not possible to reliably return the data-generating parameters. Given a data set of unlimited size assumed to be generated by some model, f , parameterised by $\boldsymbol{\theta} \in \boldsymbol{\Theta} \subset \mathbb{R}^n$, we can say some parameter $\theta_i \in \boldsymbol{\theta}$ ($i \in \{1, \dots, n\}$) is *structurally identifiable* if it can be uniquely recovered (Bickel and Doksum, 2015; Maclaren and Nicholson, 2019; Anstett-Collin et al., 2020). More precisely, we can define θ_i as structurally *globally* identifiable if for all $\boldsymbol{\theta} \in \boldsymbol{\Theta}$,

$$f(\boldsymbol{\theta}) = f(\hat{\boldsymbol{\theta}}) \implies \theta_i = \hat{\theta}_i. \quad (2.24)$$

We can then define θ_i as structurally *locally* identifiable if for all $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, there exists some neighbourhood $v(\boldsymbol{\theta})$ of $\boldsymbol{\theta}$, such that

$$\hat{\boldsymbol{\theta}} \in v(\boldsymbol{\theta}), \quad f(\boldsymbol{\theta}) = f(\hat{\boldsymbol{\theta}}) \implies \theta_i = \hat{\theta}_i. \quad (2.25)$$

In more practical terms, when considering identifiability, we are interested in whether we can *approximately* return the data-generating $\boldsymbol{\theta}$ (or a specific θ_i of interest) from the data available to us. While it is sometimes possible to overcome structural identifiability issues by designing new experiments to collect qualitatively new data, in many cases, a structurally unidentifiable parameter cannot be estimated, even in an infinite ‘perfect’ data setting. In some cases, we may be able to estimate specific parameter groupings (e.g. the product of two parameters $\theta_1\theta_2$), but the parameters that make up these groupings are algebraically redundant and hence, cannot be estimated.

Practical identifiability

In contrast, practical identifiability pertains to instability issues relating to the amount and/or quality of the data gathered. For likelihood-based inference, in the direction of a practically unidentifiable parameter, the likelihood function becomes very flat, even if it may have a unique maximum (Raue et al., 2009). Consider some noisy data, $\boldsymbol{x}$, generated from the model $f(\boldsymbol{\theta})$ such that

$$\boldsymbol{x} = f(\boldsymbol{\theta}) + \boldsymbol{\epsilon}, \quad (2.26)$$

where $\boldsymbol{\epsilon}$ represents some experimental noise. We can say that the parameter $\theta_i \in \boldsymbol{\theta}$ is *practically identifiable* given $\boldsymbol{x}$, if we can determine a

sufficiently small confidence interval around θ_i (via profile likelihood methods or similar). The size of the confidence interval is context-dependent. Practical non-identifiability generally arises as a result of noisy, inaccurate, or insufficient data. It follows that we can often minimise practical non-identifiability issues by gathering more accurate and complete data.

Functional identifiability

While identifiability is most commonly assessed in the parametric setting, in some cases it can also be useful to consider *functional identifiability*. Instead of using a parameter’s likelihood profile, we can instead get a measure of functional identifiability by producing an ensemble of fitted models (Loman and Baker, 2025). In this way, we can assess the potential variability in model outputs given the model structure and available data. An instance of functional non-identifiability can result from either structural or practical identifiability issues.

2.3.5 Uncertainty quantification

The accuracy and robustness of mathematical models must be thoroughly assessed if we are to use them in safety-critical applications. Uncertainty quantification (UQ) techniques broadly cover the set of methods that allow us to quantitatively describe the uncertainty in model predictions. Higdon et al. (2009) define UQ as follows

“UQ studies all sources of error and uncertainty, including the following: systematic and stochastic measurement error; ignorance; limitations of theoretical models; limitations of numerical representations of those models; limitations of the accu-

racy and reliability of computations, approximations, and algorithms; and human error. A more precise definition is UQ is the end-to-end study of the reliability of scientific inferences.”

With regard to cardiac electrophysiology, sources of uncertainty are introduced at all stages of the modelling process. From experimental errors and artefacts in voltage-clamp experiments, to biases introduced when post-processing data, there can be significant UQ issues to address before we even begin applying our models. Further sources of uncertainty are then introduced by the modelling assumptions and limitations of our ion channel and drug binding models. Tools such as the Fisher information matrix (FIM) (see the following section for details) can assist in quantifying uncertainty on model parameter estimates by providing a measure of how sharply the log-likelihood function varies on average. However, this technique relies on us having confidence in our log-likelihood function, which is often not realistic, especially given the various experimental artefacts commonly seen in electrophysiology data. Given the safety-critical nature of cardiac safety assessment, and as emphasised by the CiPA initiative, it is important that we can provide UQ analysis for any results pertaining to drug risk.

2.3.6 The Fisher information matrix

Evaluating the FIM can be a useful method for assessing the uncertainty in parameter estimates. The FIM provides a measure of the amount of information that a random variable, $\mathbf{X}$, has about a parameter vector, $\boldsymbol{\theta}$ (Fisher, 1922). For $\boldsymbol{\theta}$ of length p , we can define the FIM, under certain

regularity conditions (Lehmann and Casella, 1998), as a $p \times p$ matrix

$$\mathcal{I}(\boldsymbol{\theta})_{ij} = -\mathbb{E} \left[\frac{\partial^2 \ell^2(\boldsymbol{\theta}|\mathbf{X})}{\partial \theta_i \partial \theta_j} \right], \quad i, j = 1, \dots, p. \quad (2.27)$$

We can then define the *observed* Fisher information at some particular parameter set, $\boldsymbol{\theta}^*$, as

$$\mathcal{J}(\boldsymbol{\theta}^*)_{ij} = -\frac{\partial^2 \ell^2(\boldsymbol{\theta}^*|\mathbf{x})}{\partial \theta_i \partial \theta_j}, \quad i, j = 1, \dots, p, \quad (2.28)$$

for a vector of observations $\mathbf{x}$. This is considered a sample-based version of the FIM. If we have some MLE, $\hat{\boldsymbol{\theta}}$, then the observed information matrix can be evaluated at $\hat{\boldsymbol{\theta}}$ to give a measure of the local shape of the log-likelihood surface around $\hat{\boldsymbol{\theta}}$. We note that FIM-based measures assume that the log-likelihood approximates a parabolic or ellipsoid shape, an assumption that does not necessarily hold in the presence of non-identifiable parameters (Villaverde et al., 2022). UQ alternatives to the FIM include Bayesian inference or profile likelihood methods (Kreutz et al., 2013; Mitra and Hlavacek, 2019).

2.3.7 Model discrepancy

Another key consideration when fitting models to data is *model discrepancy*. We define model discrepancy as any parametric or functional difference between the data-generating process and the model being used to approximate that process. While it is well appreciated that these model discrepancies are inevitable (i.e. ‘all models are wrong, but some are useful...’ (Box and Draper, 1987)), their presence often remains unaccounted for. Ignoring model discrepancy can lead to errors in parameter estimation and predictions (Kennedy and O’Hagan, 2001). Depending on the source,

the effects of model discrepancy can be reduced by developing improved models, or by accounting for experimental artefacts during data collection or post-processing. Alternatively, the model can be extended to include the (potentially unknown) source of discrepancy, in conjunction with any available prior knowledge, via methods such as Gaussian processes (Brynjarsdóttir and O’Hagan, 2014).

2.4 Optimal experimental design

Optimal experimental design (OED) involves designing experiments to collect data that contain ‘optimal information’ for model calibration (Lindley, 1956; Kiefer, 1959; Smucker et al., 2018). This ‘optimal information’ can be defined to reduce uncertainty in parameter estimates and/or improve the accuracy of model predictions. OED methods can also lead to a reduction in the number of experimental runs required to estimate parameters to a given level of precision, in turn reducing experimental costs. OED methods have been used in a variety of fields, including systems biology, psychology, and engineering (Kutalik et al., 2004; Papadimitriou et al., 2005; Myung and Pitt, 2009). Given that the goal of OED is to maximise information, the reciprocal relationship between estimator variance and information can be used to define optimality criteria. One such example is the A-optimality criterion, which seeks to minimise the average variance of the estimated model parameters. Alternatively, the C-optimality criterion seeks to minimise the variance of a specified linear combination of model parameters. Lei et al. (2023) demonstrated an application of OED in the context of cardiac electrophysiology by designing ‘optimal’ protocols for voltage-clamp and current-clamp experiments. Employing several OED objectives, including A-optimality, these experimental designs were

shown to perform better than many manually designed experiments from the literature, achieving more accurate parameter estimation and model prediction. Further work, by Mirams et al. (2024), considered a ‘space-filling’ OED objective for protocol design that involved optimising the coverage of a three-dimensional phase space representing the channel states and command voltage.

2.5 Model selection

Given a set of candidate models, it is often useful to determine which, if any, are more suitable given some data. Model selection is defined as the process of selecting a model from a set of candidate models based on a chosen performance criterion (Hastie et al., 2009). This often involves investigating the trade-off between the simplicity of a model and how well the model fits some data.

2.5.1 The likelihood-ratio test

The likelihood-ratio test assesses whether one model is a significantly better fit than the other by using the ratio of the two models’ likelihood functions evaluated at their maxima (King, 1989; Koch, 1999). The likelihood-ratio test requires that the two models being compared are nested. Specifically, this means that we are comparing a more complex model to a simpler model, where the simpler model is the equivalent of the more complex model with constraints imposed on its parameters. In order to conduct a test, we have to specify a null hypothesis, H_0 , that states that the parameter vector $\boldsymbol{\theta}$ is in a specified subset $\boldsymbol{\Theta}_0$ of $\boldsymbol{\Theta}$. A key quantity to calculate to perform the likelihood-ratio test is the likelihood-ratio test statistic, which

is given by

$$\lambda_{LR} = -2 \ln \left[\frac{\sup_{\boldsymbol{\theta} \in \boldsymbol{\Theta}_0} \mathcal{L}(\boldsymbol{\theta})}{\sup_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{\theta})} \right], \quad (2.29)$$

where ‘sup’ here refers to the supremum. We can alternatively work with log-likelihoods and express the above as

$$\lambda_{LR} = -2 \left[\ell(\boldsymbol{\theta}_0) - \ell(\hat{\boldsymbol{\theta}}) \right], \quad (2.30)$$

where $\ell(\boldsymbol{\theta}_0)$ is the maximal value of the log-likelihood in the case where H_0 is true and $\ell(\hat{\boldsymbol{\theta}})$ is the value of the log-likelihood at the MLE when $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. The likelihood-ratio test has previously been applied to problems in cardiac electrophysiology. For example, Csanády (2006) investigated the use of the likelihood ratio test to assist in model identification of ion channel gating in synthetic data.

2.5.2 The Akaike information criterion

Another common criterion for model selection is the Akaike Information Criterion (AIC) (Stoica and Selen, 2004). The AIC is practically similar to the likelihood-ratio test, but is generally more flexible in its use, as the AIC does not rely on candidate models being nested. Let k be the number of parameters for a statistical model of some data. The AIC value of the model is then given by

$$\text{AIC} = 2k - 2\ell(\hat{\boldsymbol{\theta}}). \quad (2.31)$$

For a given set of candidate models, the preferred model is that with the minimum AIC value (Akaike, 1974; Burnham and Anderson, 2007). In practice, we can compare a set of m models by calculating a quantity known as the relative likelihood (RL) for each model $i \in [1, m]$. This is

defined as

$$\text{RL}_i = \exp((\text{AIC}_{\min} - \text{AIC}_i)/2) \quad (2.32)$$

where $\text{AIC}_{\min}$ is the minimum AIC value across all candidate models. AIC, and the closely related Bayesian information criterion (BIC), were both used for ion channel gating model selection by Ball and Sansom (1989).

2.5.3 Model selection with model discrepancy

While traditional model selection methods such as the likelihood-ratio test and AIC are useful tools for selecting an appropriate model, they can become less practically reliable in the presence of model discrepancy (in the observation model), particularly when working with high-frequency data. To illustrate this, for some input x , consider a function

$$f(x) = A \exp(x) + C, \quad (2.33)$$

with parameters A and C , that defines some mechanistic process. Now, suppose we can take measurements, y_i , that ideally follow the relationship defined by $f(x)$; however, due to some experimental artefact, the data-generating process is as follows

$$y_i = g(x) + \epsilon_i, \quad (2.34)$$

where

$$g(x) = A \exp(Bx) + C, \quad (2.35)$$

and $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ represents normally distributed, independent and identically distributed (i.i.d.) noise. The experimental artefact has introduced

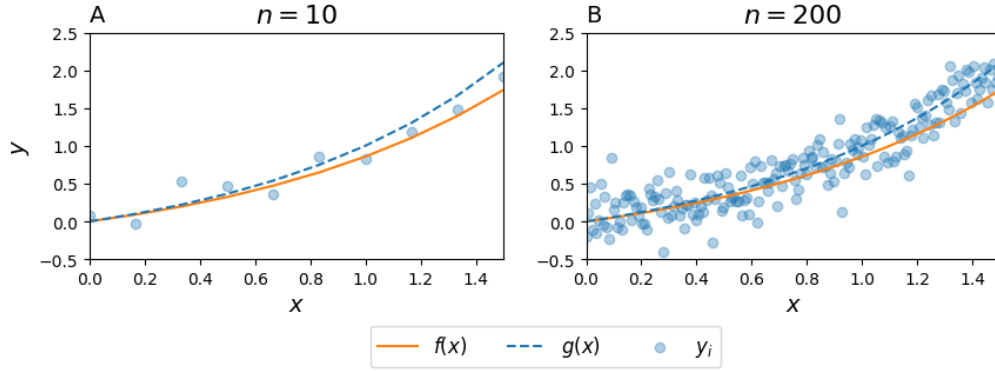


Figure 2.7: Issues with model selection in the presence of model discrepancy. A: When the sampling frequency is low ($n = 10$), the relative likelihoods favour the simpler ‘true’ model, $f(x)$, despite the presence of model discrepancy. B: For higher sampling frequencies ($n = 200$), the model discrepancy causes the relative likelihoods to instead suggest that $g(x)$ is a far more appropriate model than $f(x)$.

a discrepancy where the applied input, x , is (multiplicatively) altered by some parameter, B . In Figure 2.7, we plot, as blue dots, synthetic data, y_i for $x \in [0, 1.5]$, with $A = 0.5$, $B = 1.1$, $C = -0.5$, and $\sigma = 0.25$. The blue dashed lines indicate the trajectory of the data-generating process, $g(x)$, while the orange lines represent the ‘true’, artefact-free, $f(x)$ function. In Figure 2.7A, the synthetic data has a sampling frequency of $n = 10$, while in Figure 2.7B, this is increased to $n = 200$. If we assume that we know the values of A , B , C , and σ , then, using Equation (2.23), we can calculate the log-likelihoods of each model, $f(x)$ and $g(x)$, given the data. For the $n = 10$ data, if we then calculate the relative likelihoods of the two models, $f(x)$ and $g(x)$, via Equation (2.32), we obtain values of 1 and 0.29, respectively; the simpler model, $f(x)$, has a higher relative likelihood and hence is favoured, despite the discrepancy introduced by the Bx artefact in the data-generating process. However, when increasing the sampling frequency to $n = 200$, we find that the result is drastically reversed, with relative likelihoods for $f(x)$ and $g(x)$ of 3.57×10^{-8} and 1, respectively. As a result of the high sampling frequency, the relative likelihoods now suggest that $g(x)$ is a far more appropriate model for the data than $f(x)$. This may seem

like an obvious and not particularly useful result; however, consider the practical implications when attempting to gain mechanistic insights from a model. If we were unaware of the presence of the experimental artefact, then, in the $n = 200$ scenario, we would confidently select the $g(x)$ model as the ‘true’ model of the underlying mechanisms, misinterpreting the Bx component as a part of these mechanisms, as opposed to some undesirable factor introduced when collecting the data. This suggests that model selection methods should be applied with caution, especially when we are aware of experimental artefacts in the data collection process.

2.6 Model discovery

The development of mathematical models is often a time and labour-intensive process relying on expert knowledge and insight. A potential alternative to traditional manual model-building methods is data-driven model discovery. Methods for model discovery are broad, but largely involve learning unknown model equations (or parts of these equations) from data.

2.6.1 The LASSO method

The Least Absolute Shrinkage and Selection Operator (LASSO) method is a regression analysis approach that promotes parsimony and interpretability of fitted statistical models (Tibshirani, 1996). In its most basic form, the LASSO is a simple extension of linear regression that encourages model sparsity. We define sparsity here as a property of a model pertaining to the fraction of fitted model parameters that are non-zero; i.e. a model with a small fraction of non-zero fitted parameters would be considered a sparse

model. The simple linear regression version of LASSO can be defined as follows. Say we have a sample of N data points x_i for $i \in 1, \dots, N$, such that $\mathbf{x} = [x_1, x_2, \dots, x_N]$. For each x_i , we have a corresponding vector of m covariates $\phi_i := [\phi_{i,1}, \phi_{i,2}, \dots, \phi_{i,m}]$. We can then define a design matrix $\Phi = [\phi_1, \phi_2, \dots, \phi_N]^T$. The objective of the LASSO is to solve the optimisation problem

$$\min_{\beta} \left\{ \frac{1}{N} \|\mathbf{x} - \Phi\beta\|_2^2 + \lambda \|\beta\|_1 \right\}, \quad (2.36)$$

where β is the parameter vector $\beta = [\beta_1, \beta_2, \dots, \beta_m]^T$, and λ is a hyperparameter known as the regularisation parameter. In effect, the LASSO allows us to estimate β for a linear model of the form

$$\mathbf{x} = \Phi\beta + \epsilon, \quad (2.37)$$

where ϵ represents some vector of Gaussian, i.i.d. noise. Note that if $\lambda = 0$, the LASSO is equivalent to standard least squares linear regression, while a positive value of λ penalises non-zero parameters in β , in doing so, encouraging simple and interpretable models. In contrast to other regression approaches such as ridge regression, the LASSO method uses an L_1 -norm regularisation term ($\|\beta\|_1$) which specifically forces some parameters to zero, as opposed to an L_2 -norm, which shrinks all coefficients towards zero without eliminating them. If the covariates in Φ are on different scales, it is often practical to standardise the columns of Φ by, for example, dividing each column by the L_2 -norm. LASSO has been previously applied in cardiac electrophysiology applications (Howell et al., 2021; Jia et al., 2022), as well as more broadly in mathematical biology (Gupta et al., 2020; Elmes et al., 2022), to simplify complex models and aid in clinical prediction.

2.6.2 The SINDy method

Sparse Identification of Non-linear Dynamics (SINDy) is a model discovery algorithm used for learning dynamical systems from data via sparsity-encouraging methods such as the LASSO (Brunton et al., 2016). Suppose the vector $\mathbf{x}(t) \in \mathbb{R}^m$ is the state of some dynamical system at time t and we want to learn a set of ODEs

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\theta}), \quad (2.38)$$

that describe the system dynamics. If we observe the process at time points $t_1, t_2, \dots, t_n$, we can form an $n \times m$ matrix, $\mathbf{X}$, of observations across all n time points and m states, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_n) \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_m(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_m(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_n) & x_2(t_n) & \cdots & x_m(t_n) \end{bmatrix}, \quad (2.39)$$

where the T superscript indicates a transposed vector. We can then determine an approximation for the derivatives $d\mathbf{X}/dt$ via numerical methods. This could be achieved, for example, by first approximating $\mathbf{x}(t)$ by fitting basis splines to the observed data, $\mathbf{X}$. These basis splines, consisting of a collection of piecewise polynomials, could then be differentiated to provide estimates of the gradients of $\mathbf{x}(t)$. We will define the $(n \times m)$ matrix of

(approximated) time derivatives as

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{x}}^T(t_1) \\ \dot{\mathbf{x}}^T(t_2) \\ \vdots \\ \dot{\mathbf{x}}^T(t_n) \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \cdots & \dot{x}_m(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \cdots & \dot{x}_m(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{x}_1(t_n) & \dot{x}_2(t_n) & \cdots & \dot{x}_m(t_n) \end{bmatrix}. \quad (2.40)$$

The set of equations, $\mathbf{f}$, defining the dynamical system, can then be estimated by modelling the system as a sparse regression problem of the form

$$\dot{\mathbf{X}} = \Phi(\mathbf{X})\boldsymbol{\beta} + \boldsymbol{\epsilon}, \quad (2.41)$$

similarly to Equation (2.37). The design matrix, $\Phi(\mathbf{X})$, is now a basis of candidate non-linear functions of $\mathbf{X}$. For example, a basis containing constant, linear, and polynomial terms would take the form

$$\Phi(\mathbf{X}) = [\mathbf{1}_{n \times m}, \mathbf{X}, \mathbf{X}^2], \quad (2.42)$$

where $\mathbf{1}_{n \times m}$ is a matrix of ones of dimension $n \times m$. We then define $\boldsymbol{\beta}$ as a matrix of coefficients that dictate which non-linear functions in $\Phi(\mathbf{X})$ are active in the dynamical system. For the example design matrix in Equation (2.42), $\boldsymbol{\beta}$ would have dimension $3m \times m$. With this sparse regression setup, we can turn to SINDy to determine a sparse system that can accurately represent the data. We estimate the regression coefficients in $\boldsymbol{\beta}$ by minimising the regularised sum of square errors:

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{N} \|\dot{\mathbf{X}} - \Phi(\mathbf{X})\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \quad (2.43)$$

which is equivalent to the LASSO objective in Equation (2.36). SINDy has previously been used to learn the missing dynamics in cardiac action

potential models from synthetic data (Welch and Cherry, 2025), and in several other biological applications (Mangan et al., 2017; Hirsh et al., 2022).

2.6.3 Artificial neural networks

Taking inspiration from biological neural networks, artificial neural networks (which we will refer to simply as neural networks henceforth) are computational models that can be trained to capture complex non-linear dynamics. Consisting of a network of connected nodes (or neurons), in their most fundamental form, the structure of a neural network involves an input layer, one or more hidden layers, and an output layer. By defining a series of weights and biases, and selecting some activation function, one or more inputs are propagated through the network from input layer, to hidden layer(s), to output layer, ultimately producing some output. By selecting an appropriate activation function, even a relatively simple network structure can produce complex non-linear output. In Figure 2.8, we include an illustration of an example of a simple neural network structure. This example has two nodes in the input layer, one hidden layer with three nodes, and an output layer with a single node. For each node in the hidden layer, we take a weighted sum of the inputs, add a bias parameter, before applying an activation function. This process is repeated for the output layer, taking a weighted sum of the hidden layer nodes and adding a bias. Unless we require specific constraints on outputs, generally, no activation function is necessary when propagating from the hidden layer to the output layer. We can think of this example network as a function that takes two inputs (i_1 and i_2) and produces a single output (o_1). In fact, if we consider the weight parameters in the illustration (w) and select some activation

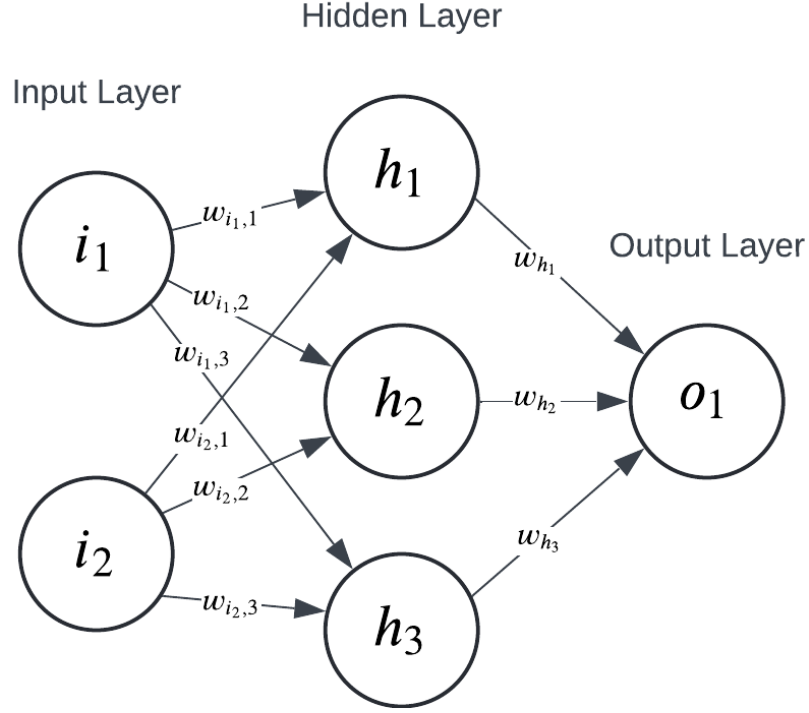


Figure 2.8: An example neural network structure. This network has two inputs nodes (i_1 and i_2), one hidden layer with three nodes (h_1 , h_2 , and h_3), and one output node (o_1).

function, for example $\tanh$, then we can explicitly define the function that this example network represents. The output, o_1 , can be defined as

$$o_1 = \sum_{i=1}^3 w_{h_i} h_i + b_{out}, \quad (2.44)$$

where b_{out} is some bias parameter, and h_i for $i \in [1, 3]$ are defined as

$$h_1 = \tanh(w_{i_1,1}i_1 + w_{i_2,1}i_2 + b_1), \quad (2.45)$$

$$h_2 = \tanh(w_{i_1,2}i_1 + w_{i_2,2}i_2 + b_2), \quad (2.46)$$

$$h_3 = \tanh(w_{i_1,3}i_1 + w_{i_2,3}i_2 + b_3), \quad (2.47)$$

where b_1 , b_2 , and b_3 are also bias parameters. We can then define a loss function that compares the neural network output to some data. By optimising the weight and bias parameters, the neural network can then be trained to accurately predict a given process characterised by the data.

The example network in Figure 2.8 has a total of 13 parameters, making (gradient-free) optimisation fairly straightforward; however, as the size of the network structure increases, the number of parameters grows rapidly. For more complex network structures, gradient-based optimisation methods, employing back-propagation to compute gradients, can be used to efficiently train the network parameters (Goodfellow et al., 2016).

2.6.4 Universal differential equations

A Universal Differential Equation (UDE) is a differential equation defined either partially or in full by some Universal Approximator (UA) such as a neural network (Chen et al., 2018; Kidger et al., 2020). In their paper introducing UDEs, Rackauckas et al. (2020) describe a UA as “a parameterized object capable of representing any possible function in some parameter size limit”. If we let U represent some UA, then we can define a dynamical system as

$$\frac{d}{dt}\mathbf{x}(t) = U(\mathbf{x}(t), \boldsymbol{\phi}), \quad (2.48)$$

where $\boldsymbol{\phi}$ is the vector of parameters of the UA, e.g. the weights and biases of a neural network. If we have some prior knowledge of the particular dynamics of the system, then we can include this as the function $\mathbf{g}$. Including this in our system of equations, we get

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{g}(\mathbf{x}(t), \boldsymbol{\theta}_{\mathbf{g}}) + U(\mathbf{x}(t), \boldsymbol{\phi}), \quad (2.49)$$

where $\boldsymbol{\theta}_{\mathbf{g}}$ are the parameters of the known dynamics described by $\mathbf{g}$. If U is a neural network, then for a particular set of weights and biases, $\boldsymbol{\phi}$, and parameters, $\boldsymbol{\theta}_{\mathbf{g}}$, we can solve the set of ODEs and obtain a solution which we will denote $\hat{X}_{\boldsymbol{\phi}}$. It follows that an appropriate loss function, using the

(sum of squares) Frobenius norm, can be defined as

$$\min_{\theta_g, \phi} ||X - \hat{X}_\phi||_F, \quad (2.50)$$

where X is the matrix of observations as defined in Equation (2.39). UDEs have previously been applied in varied mathematical biology applications such as epidemiology and systems biology (Kuwahara and Bauch, 2024; Philipps et al., 2025), as well as in the context of ion channel modelling (Lei and Mirams, 2021; Su et al., 2021).

Chapter 3

Improving robustness to model discrepancy in binding model calibration

Preamble

When fitting mathematical models to data, it is important to consider model *discrepancy* (alternatively referred to as model *misspecification*). Model discrepancy broadly refers to differences between the dynamics defined by a model and the ‘true’ dynamics we are attempting to model. This discrepancy can be introduced in several ways, including, but not limited to: simplifying modelling assumptions, gaps in our knowledge of the dynamics being modelled, and experimental artefacts introduced during data collection. When dealing with complex biological systems in particular, assumptions are inevitably made to reduce model complexity and computational costs. As discussed in Section 2.1.7, voltage clamp experiments can often introduce experimental artefacts, which can result in biased and

over-confident parameter estimates if these artefacts are left unaccounted for (Brynjarsdóttir and O’Hagan, 2014). In this chapter, we consider methods for improving the robustness of fitting drug binding models with regard to model discrepancy, enabling more accurate parameter estimates. We consider two models of binding dynamics: a simple conductance block model and a more complex kinetic model of block. By introducing a normal ratio error model, we demonstrate how the calibration of these two binding models can be improved in comparison to model fitting using a more simple Gaussian error model. We note that while these methods are not a panacea for model discrepancy, they represent a useful tool for improving the process of fitting drug-binding models to data.

3.1 Introduction

To measure ion channel drug block, it is often useful to consider the proportion of channels blocked. For example, estimations of $[IC_{50}]$, the drug concentration at which we see a 50% channel block, can be used as a fairly basic measure of drug action. It follows that a typical voltage clamp experiment to measure ion channel drug block involves collecting two sets of time-series data: a control current before the drug compound is introduced, $y(t)$, and the current in the presence of the drug compound, $x(t)$, where $y(t)$ and $x(t)$ are time-dependent state variables. These data are often post-processed to obtain the proportion of channels blocked, $x(t)/y(t)$ (Li et al., 2017). As demonstrated by Lei et al. (2024), it is then possible to fit mechanistic drug-binding models to these proportion data. While the drug mechanism should be fully described by $x(t)$, there is an intuition that model discrepancy in the underlying ion channel model may partially cancel when working with the $x(t)/y(t)$ quantity; any multiplicative discrepancy

effect that is independent of the drug mechanism should be ‘cancelled out’ by working with the ratio of drug and control currents. In this chapter, we seek to investigate, via a synthetic data study, whether this intuition is accurate, and also whether model fitting can be improved by considering a normal ratio error model as opposed to a more simple Gaussian error model.

3.2 Mathematical models

3.2.1 Ion channel model

We will begin by introducing the mathematical modelling framework. To model baseline ion channel currents with no drug effect, we will use the gating model described by Beattie et al. (2018), and illustrated in the Markov diagram included in the previous chapter in Figure 2.5c. The voltage-dependent transition rates, k_i , are defined (as per Equation (2.12)) via α_i and β_i parameters. Full Beattie model equations are included in Appendix A.1.1. For this chapter, unless otherwise specified, we will use the mean Beattie model parameters from Lei et al. (2019a) fitted to hERG current data collected at physiological temperature (37°C). The models of drug block considered in this chapter involve extending the Beattie model to include drug effects. When fitting these drug block models, we will fix the parameters of the ‘base’ Beattie model, and fit only the parameters of the drug block model extensions. We describe these model extensions in the following sections, starting with a conductance block model, and then a more complex kinetic model of block.

3.2.2 Conductance block model

In the previous chapter (Section 2.2.3), we introduced the ‘conductance block’ model for modelling ion channel drug block. This model ignores specific binding kinetics and simply models drug block as a scaling factor on channel current. We reproduce Equation (2.16) here, which defines the rate of change of the proportion of unblocked channels, b ,

$$\frac{db}{dt} = -k_{\text{on}}b[D]^n + k_{\text{off}}(1 - b). \quad (3.1)$$

Here k_{on} and k_{off} are, respectively, the binding and unbinding rate parameters, $[D]$ is the drug concentration, and n is the Hill coefficient. We can then define the drug-blocked current, I_b , similarly to Equation (2.13) as

$$I_b = g \cdot b \cdot O \cdot (V - E), \quad (3.2)$$

where O is the open proportion defined by our physiological ion channel model. For $b = 1$, Equation (3.2) simply reduces to the standard equation for drug-free ion channel current described by Equation (2.11), while $b = 0$ implies full block, i.e. zero current.

3.2.3 Kinetic model of block

We also consider a more complex kinetic model of drug block. This model can be described by the Markov model illustrated in Figure 3.1. We have extended the Beattie model from Figure 2.5c to include two new binding states, inactivate drug-bound (ID) and open drug-bound (OD). These additional states comprise the drug-binding model component, which is equivalent to drug-binding model 1 in Figure 2.6. Specifically, this binding model

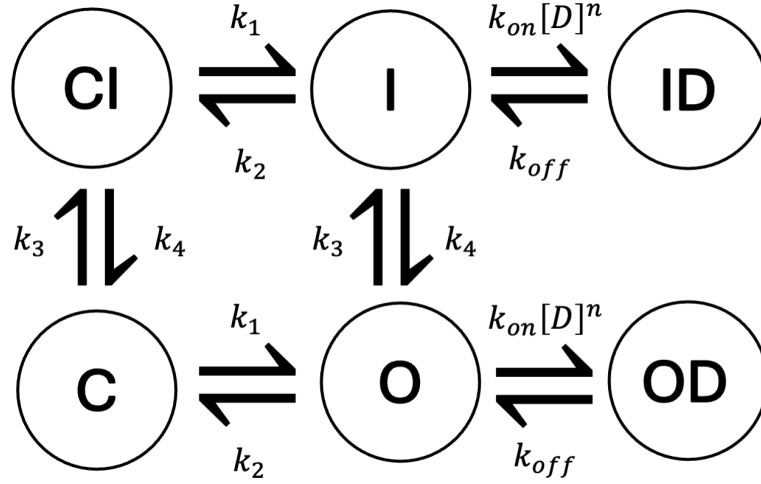


Figure 3.1: Markov diagram of a physiological hERG model (Beattie model from Figure 2.5c) with a kinetic block component representing drug-binding model 1 in Figure 2.6.

describes non-trapping dynamics with identical open and inactive binding rates. Equivalently to the conductance block model, drug on-binding occurs at a rate of $k_{\text{on}}[D]^n$, while the unbinding rate is k_{off} . For the purposes of this chapter, for both models we will assume that n is known, and the set of parameters we wish to estimate is $\theta = [k_{\text{on}}, k_{\text{off}}]$. In contrast to the conductance block model, we can simply define I_b as

$$I_b = g \cdot O \cdot (V - E), \quad (3.3)$$

where instead of the b scaling factor, a reduction in current results from a reduction in O , which is induced by the additional OD and ID states in the Markov model. The full equations for this kinetic model of block are included in Appendix A.2.1.

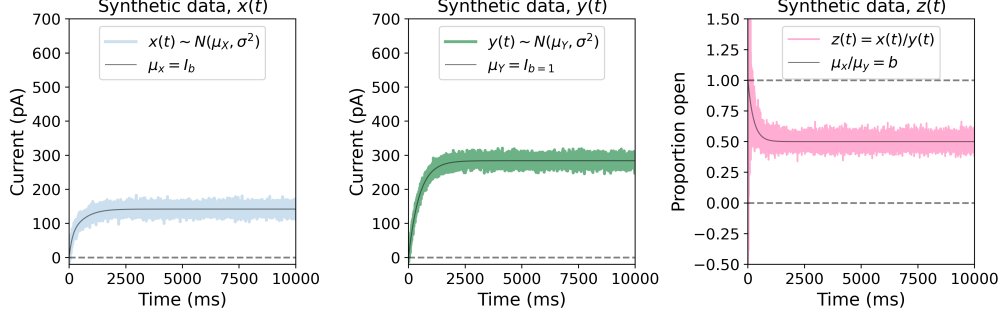


Figure 3.2: Synthetic data generated using the conductance block model. Left: Current, $x(t)$, in the presence of a 100 nM virtual drug with binding parameters $k_{\text{on}} = 2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}$, $k_{\text{off}} = 2 \times 10^{-3} \text{ ms}^{-1}$. Gaussian noise is added with standard deviation $\sigma = 10 \text{ pA}$. Middle: Current, $y(t)$, with no drug. Gaussian noise is again added with standard deviation $\sigma = 10 \text{ pA}$. Right: Proportion of channels open, $x(t)/y(t)$.

3.3 Fitting binding models to data

To assess different model fitting methods, we will begin by generating some synthetic drug block data. For both models, we can define this data-generating process as follows

$$y(t) \sim \mathcal{N}(I_{b=1}(t, V_t), \sigma^2), \quad (3.4)$$

$$x(t) \sim \mathcal{N}(I_b(t, V_t, \boldsymbol{\theta}), \sigma^2), \quad (3.5)$$

where σ is the standard deviation of the measurement error on both the drug and control currents, and V_t is the voltage at time t . To avoid confusion with the inactivated state, I, in the Markov models, we will use the notation $I_{b=1}$ to refer to the drug-free ‘control’ currents. With this framework in place, we now wish to determine whether there is a benefit to inferring our drug-binding model parameters, $\boldsymbol{\theta}$, by fitting our drug-binding models to the quantity $z(t) = x(t)/y(t)$, or if using just $x(t)$ is preferable. In Figure 3.2, we generate and plot synthetic data $x(t)$, $y(t)$, and $z(t)$ under the conductance block model with $\sigma = 10 \text{ pA}$. We choose a drug concentration of 100 nM and arbitrarily select $\boldsymbol{\theta} = [2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2 \times 10^{-3} \text{ ms}^{-1}]$

representing realistic binding dynamics for some virtual drug. The voltage protocol used to generate the data is a simple modified Milnes protocol (Milnes et al., 2010) which begins at -80 mV, steps up to 0 mV for 10 s, before returning to -80 mV. For the purposes of this example, we plot only the 10 s where $V = 0$ mV (where observed currents are non-zero), and set $t = 0$ ms to be the time at which the 0 mV step begins. We can fit our conductance block model to the $x(t)$ synthetic data via maximum likelihood estimation, where the log-likelihood function for Gaussian noise (which is the data-generating noise model for x) is

$$\ell(\boldsymbol{\theta}, \sigma | x(t)) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} \sum_{t \in T} [x(t) - I_b]^2, \quad (3.6)$$

and n is the number of data-points. To maximise the log-likelihood, we run five repeats of the CMA-ES optimisation algorithm with random initialisations for both binding parameters drawn from wide bounds of $[1 \times 10^{-9}, 1]$. We also fit the noise parameter, σ , and for this we use initialisation bounds of $[1 \times 10^{-2}, 1 \times 10^2]$. We use a log-transformed parameter space when sampling these initialisations and when running the optimisations. We use the default CMA-ES hyperparameters as implemented in Probabilistic Inference on Noisy Time-Series (PINTS) (Clerx et al., 2019b), and run for a maximum of 100 unchanged iterations (with a threshold of 1×10^{-1}). Of the five repeats, we select the MLE solution, $[\hat{\boldsymbol{\theta}}, \hat{\sigma}]$, corresponding to the maximum likelihood. In Figure 3.3, we plot: the fit to $x(t)$, a comparison between $b(t, \boldsymbol{\theta})$ and $b(t, \hat{\boldsymbol{\theta}})$, and a comparison between data-generating parameters and the newly fitted parameters. The model fit is excellent, with $b(t, \hat{\boldsymbol{\theta}})$ appearing to perfectly match the data-generating $b(t, \boldsymbol{\theta})$ dynamics. This is corroborated by the returned MLE, $\hat{\boldsymbol{\theta}} = [2.02 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2.02 \times 10^{-3} \text{ ms}^{-1}]$ and $\hat{\sigma} = 9.99 \text{ pA}$. In the left subplot of Figure 3.4, we plot a heatmap of the log-likelihood surface

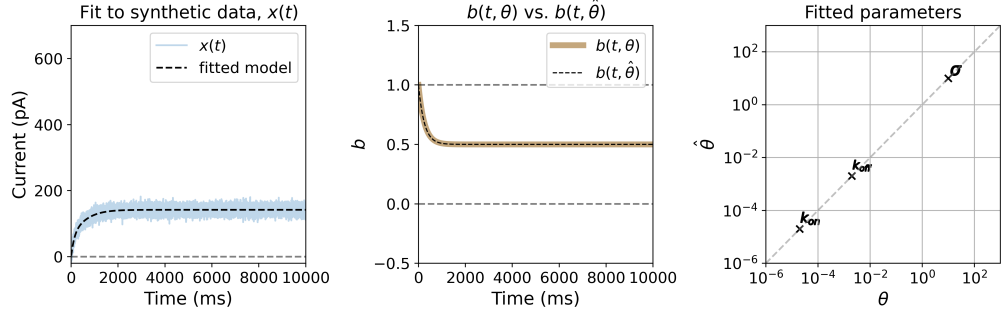


Figure 3.3: Model and parameter fits when fitting the conductance block model to $x(t)$. Left: Model fit (black dashed line) to $x(t)$ data (noisy blue trace). Middle: Model fit (black dashed line) to the data-generating block model (brown solid line). Right: Fitted parameters, $\hat{\theta}$, vs. data-generating parameters, θ . The grey dashed line along the diagonal indicates where well-estimated parameters should lie.

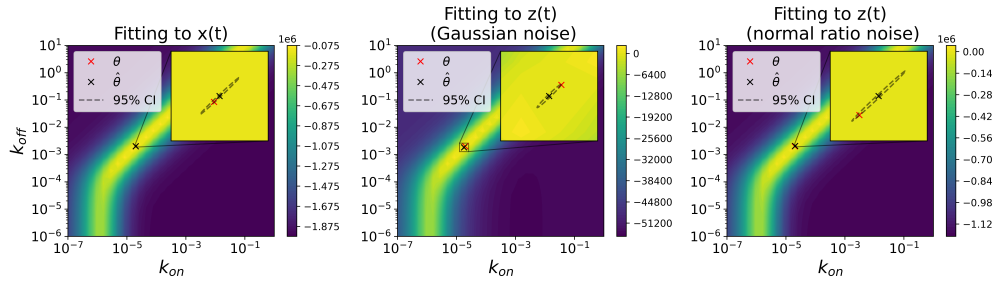


Figure 3.4: Log-likelihood heatmaps for the conductance block model with no model discrepancy. 95% confidence ellipses are calculated via finite difference FIM approximation and plotted as grey dashed lines. Left: Fitting to $x(t)$. Middle: Fitting to $z(t)$ with a Gaussian noise model. Right: Fitting to $z(t)$ with a normal ratio noise model.

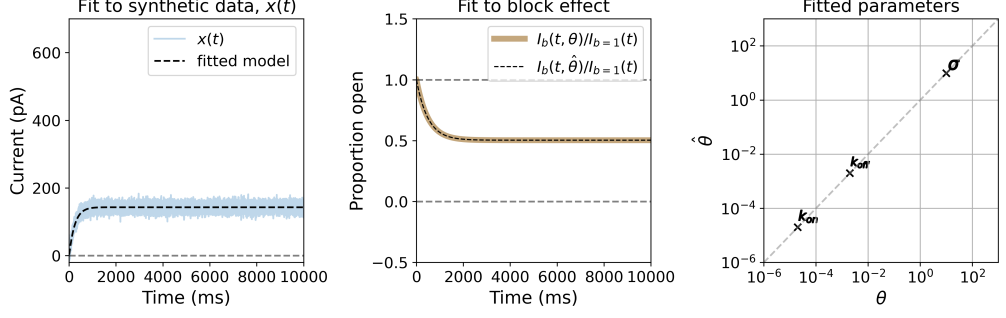


Figure 3.5: Model and parameter fits when fitting the kinetic model of block to $x(t)$. Subplots are similarly defined as in Figure 3.7.

$\ell(\boldsymbol{\theta}, \sigma | x(t))$ with fixed $\sigma = \hat{\sigma}$. We indicate $\boldsymbol{\theta}$ and $\hat{\boldsymbol{\theta}}$, and also include a 95% confidence ellipse around the MLE. We estimate this confidence interval by calculating a finite difference local approximation of the FIM in log-transformed parameter space. The data-generating parameters, $\boldsymbol{\theta}$, lie within this very tight confidence ellipse. So, it appears that in this simple case, with no model discrepancy, fitting to $x(t)$ appears to be sufficient. In Figure 3.5, we plot equivalent model fits for synthetic $x(t)$ data generated from the kinetic model of block, giving similar results.

3.4 Fitting in the presence of model discrepancy

We now investigate how the presence of model discrepancy can affect the model fitting procedure. Considering the conductance block model first, we introduce a multiplicative discrepancy term, $\delta(t)$, into our synthetic data generation. We redefine our data-generating process as

$$y(t) \sim \mathcal{N}(\delta(t)I_{b=1}(t, V_t), \sigma^2), \quad (3.7)$$

$$x(t) \sim \mathcal{N}(\delta(t)I_b(t, V_t, \boldsymbol{\theta}), \sigma^2), \quad (3.8)$$

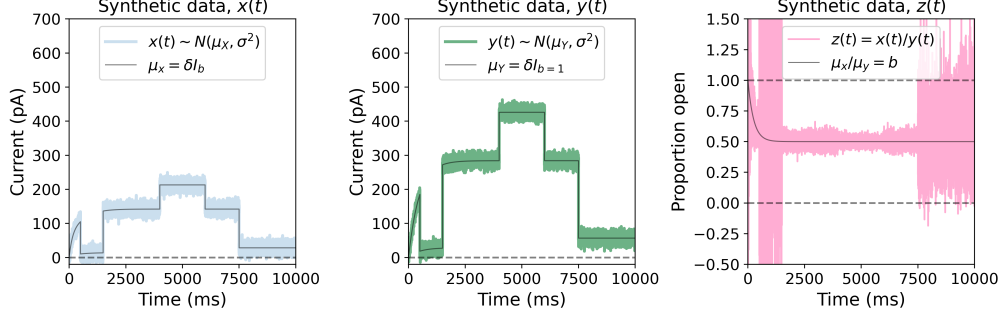


Figure 3.6: Synthetic data generated using the conductance block model and a piecewise discrepancy function defined by $\delta(t)$ in Equation (3.9). Subplots are similarly defined as in Figure 3.2.

where, for $\delta(t)$, let us define a rather arbitrary piecewise discrepancy function

$$\delta(t) = \begin{cases} 0.1 & 500 \leq t < 1500 \\ 0.2 & 7500 \leq t < 10000 \\ 1 & t < 500, 1500 \leq t < 4000, 6000 \leq t < 7500 \\ 1.5 & 4000 \leq t < 6000. \end{cases} \quad (3.9)$$

This function is a deliberately extreme choice of model discrepancy, with which we can investigate the benefit of fitting to the ratio of $x(t)/y(t)$ in a setting where we expect this to work well. In Section 3.6, we then consider a more physiologically realistic example of model discrepancy. With $\delta(t)$ defined, we can then generate new synthetic data under the same procedure as in Figure 3.2, and we plot this in Figure 3.6. Once again, we can attempt to fit our conductance block model to $x(t)$, assuming no knowledge of the presence of $\delta(t)$ in the data-generating process. We use the same log-likelihood as defined in Equation (3.6), and the same optimisation procedure as described in the previous section. In Figure 3.7 we plot the model fit and fitted parameter values. We see that the discrepancy appears to be biasing the model fit, and we are unable to re-

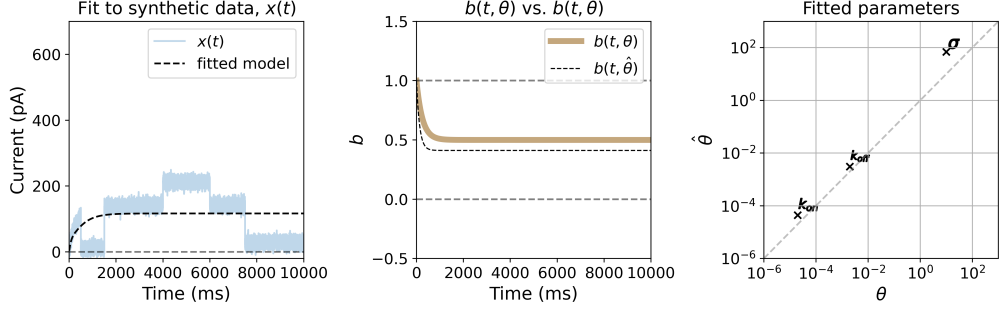


Figure 3.7: Model and parameter fits when fitting the conductance block model to $x(t)$ in the presence of multiplicative model discrepancy defined by the function $\delta(t)$ in Equation (3.9). Subplots are similarly defined as in Figure 3.3.

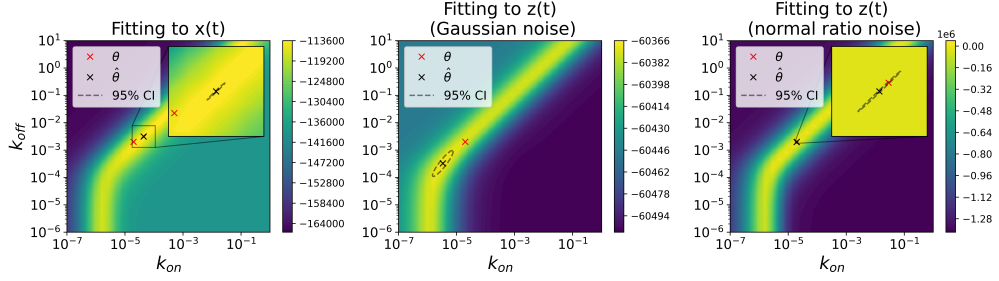


Figure 3.8: Log-likelihood heatmaps for the conductance block model in the presence of multiplicative model discrepancy defined by the function $\delta(t)$ in Equation (3.9). 95% confidence ellipses are calculated via finite difference FIM approximation and plotted as grey dashed lines. Subplots are similarly defined as in Figure 3.4.

turn accurate estimates of the data-generating parameters. The MLE is $\hat{\theta} = [4.45 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 3.11 \times 10^{-3} \text{ ms}^{-1}]$ and the noise parameter is overestimated with $\hat{\sigma} = 70.4 \text{ pA}$. The left subplot of Figure 3.8 shows the approximate FIM-based 95% confidence ellipse around $\hat{\theta}$, which does not contain θ . Clearly, the presence of model discrepancy is causing issues when fitting our conductance block model to $x(t)$. This motivates instead fitting to $z(t)$ based on the intuition that the model discrepancy may cancel (to leading order). As $z(t)$ is the ratio of two variables drawn from normal distributions, we should use a different log-likelihood, but for now, we will assume $z(t)$ itself is drawn from a normal distribution and use the following

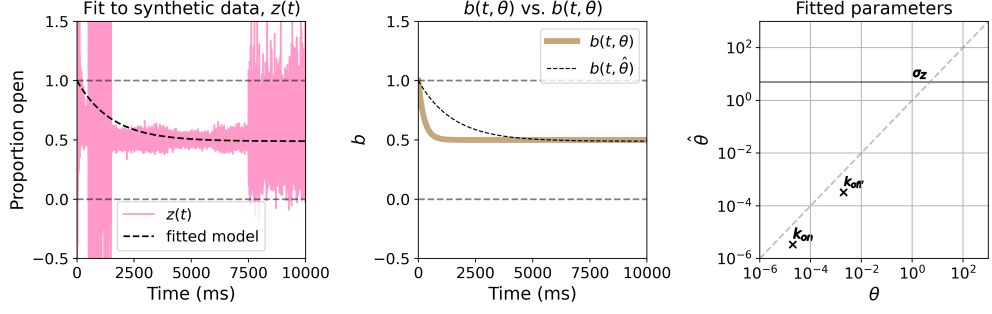


Figure 3.9: Model and parameter fits when fitting the conductance block model to $z(t)$ in the presence of multiplicative model discrepancy defined by the function $\delta(t)$ in Equation (3.9). It is (incorrectly) assumed that $z(t)$ has Gaussian noise. Left: Model fit (dashed line) to $z(t)$ data (noisy pink trace). Middle: Model fit (black dashed line) to the data-generating block model (brown solid line). Right: Fitted parameters, $\hat{\theta}$, vs. data-generating parameters, θ . The interpretation of the ‘data-generating’ noise parameter, σ_Z , is not obvious given that the data $z(t)$ are not generated with Gaussian noise, so we simply plot a vertical line to indicate the fitted value $\hat{\sigma}_Z$.

(inaccurate) log-likelihood

$$\ell(\theta, \sigma_Z | z(t)) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma_Z^2) - \frac{1}{2\sigma_Z^2} \sum_{t \in T} [z(t) - b]^2. \quad (3.10)$$

Here σ_Z is the standard deviation of the (wrongly) assumed Gaussian noise on the synthetic data $z(t)$. The likelihood defined in Equation (3.10) is equivalent to using a sum of squares error function, which was the method used by Lei et al. (2024) to fit binding models to data in the form of $z(t)$. In Figure 3.9, we plot the model fit to $z(t)$ and the fitted parameters. Again, we observe significant differences between $b(t, \hat{\theta})$ and $b(t, \theta)$, and the fitted parameters are even further from their true values ($\hat{\theta} = [3.38 \times 10^{-6} \text{ ms}^{-1} \text{ nM}^{-n}, 3.23 \times 10^{-4} \text{ ms}^{-1}]$). In the middle subplot in Figure 3.8, we plot θ and $\hat{\theta}$ on a heatmap of the log-likelihood function. The 95% confidence ellipse plotted around the MLE, once again, does not contain the true data-generating parameters. It seems that the incorrect assumption of Gaussian noise in $z(t)$ is causing issues, resulting in less accurate parameter estimates than when fitting to $x(t)$.

3.5 A normal ratio log-likelihood

We next consider $z(t)$ to be the ratio of two normal variables, i.e. the true data-generating process. For this case, we will need to determine a new log-likelihood for this normal ratio. Consider two independent, normally distributed random variables $X \sim (\mu_x, \sigma_x^2)$ and $Y \sim (\mu_y, \sigma_y^2)$. If we define the following variables $\beta = \mu_x/\mu_y$, $\rho = \sigma_x/\sigma_y$ and $\delta_y = \sigma_y/\mu_y$, then the random variable $Z = X/Y$ has the following Probability Density Function (PDF) (Díaz-Francés and Rubio, 2013)

$$f_Z(z; \beta, \rho, \delta_y) = \frac{\rho}{\pi(1 + \rho^2 z^2)} \left\{ \exp \left[-\frac{\rho^2 \beta^2 + 1}{2\delta_y^2} \right] + q \sqrt{\frac{\pi}{2}} \operatorname{erf} \left(\frac{q}{\sqrt{2}} \right) \exp \left[-\frac{\rho^2 (z - \beta)^2}{2\delta_y^2(1 + \rho^2 z^2)} \right] \right\}, \quad (3.11)$$

where

$$q = \frac{1 + \beta \rho^2 z}{\delta_y \sqrt{1 + \rho^2 z^2}}. \quad (3.12)$$

This PDF is based on the expression derived by Kamerud (1978), and has had previous use in physiological applications (Kuethe et al., 2000). In the case of our synthetic $z(t)$ data, we have $\sigma_x = \sigma_y \implies \rho = 1$. We also note that $\beta = b(t, \boldsymbol{\theta})$. It follows that the PDF for $z(t)$ can be written as

$$f_Z(z; b, \delta_y) = \frac{1}{\pi(1 + z^2)} \exp \left[-\frac{b^2 + 1}{2\delta_y^2} \right] + \frac{1 + bz}{\pi \delta_y (1 + z^2)^{\frac{3}{2}}} \sqrt{\frac{\pi}{2}} \operatorname{erf} \left(\frac{1 + bz}{\delta_y \sqrt{2(1 + z^2)}} \right) \exp \left[-\frac{(z - b)^2}{2\delta_y^2(1 + z^2)} \right]. \quad (3.13)$$

We should note here that $\delta_y = \sigma/\delta I_{b=1}$, which depends on the unknown discrepancy function, $\delta(t)$. To overcome this, we can work with an approx-

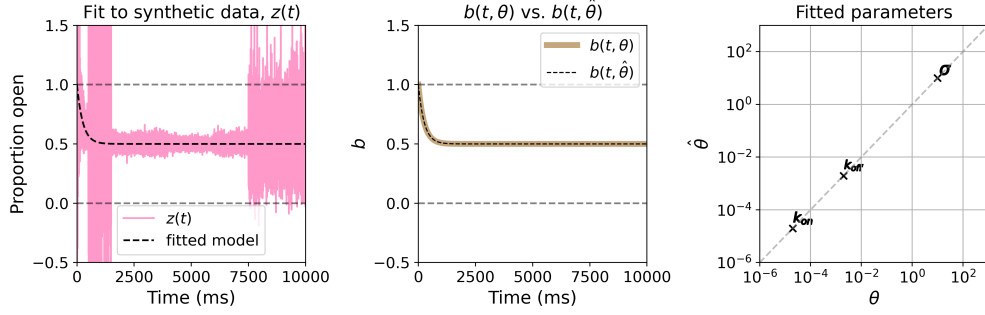


Figure 3.10: Model and parameter fits when using the normal ratio log-likelihood (Equation (3.14)) to fit the conductance block model to $z(t)$ in the presence of multiplicative model discrepancy defined by the function $\delta(t)$ in Equation (3.9). Subplots are similarly defined as in Figure 3.9.

imation for $\delta(t)I_{b=1}(t, V_t)$ by fitting splines to our control data, $y(t)$. Given the obvious discontinuities in $y(t)$, we fit these splines by first dividing $y(t)$ into the six distinct sections corresponding to the distinct discrepancy sections defined in Equation (3.9) (i.e. $t_1 = [0, 500]$, $t_2 = [500, 1500]$ etc.). We then use the `scikit-fda` Python package to fit basis splines to each of these sections. We use 4th-order polynomials and three evenly spaced knots for each section. Our log-likelihood can then be defined as

$$\ell(\boldsymbol{\theta}, \sigma | z(t)) = \sum_{t \in T} \ln \left[f_Z(z(t); b(t), \tilde{\delta}_y(t)) \right], \quad (3.14)$$

where $\tilde{\delta}_y(t)$ is the spline-fitted approximation of $\delta_y(t)$. With an appropriate log-likelihood defined in Equation (3.14), we can then optimise as previously and obtain $\hat{\boldsymbol{\theta}}$ and $\hat{\sigma}$. In Figure 3.10 we plot the fitted model and parameters. In comparison to both previous cases (fitting to $x(t)$ and fitting to $z(t)$ with a Gaussian noise assumption) we now return an MLE very close to the data-generating parameters despite the presence of model discrepancy. The MLE is $\hat{\boldsymbol{\theta}} = [1.96 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 1.96 \times 10^{-3} \text{ ms}^{-1}]$ (cf. $\boldsymbol{\theta} = [2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2 \times 10^{-3} \text{ ms}^{-1}]$) with fitted $\hat{\sigma} = 10.04 \text{ pA}$ (cf. $\sigma = 10 \text{ pA}$). We can again consider the likelihood surface via the heatmap plotted in the right subplot of Figure 3.8. We note that the true data-

generating parameters, k_{on} and k_{off} , are now contained within a tight 95% confidence ellipse. It appears that, for the conductance block model, with some unknown (multiplicative) discrepancy in the underlying ion channel model, fitting to the ratio, $z(t)$, with a normal ratio noise assumption (via an improved approximation of the true likelihood), provides more accurate predictions than when fitting to $x(t)$, or $z(t)$ with a Gaussian noise assumption. We next consider a more physiologically realistic example of model discrepancy using the kinetic model of block.

3.6 Physiologically-realistic model discrepancy

In the preceding conductance block example, the model discrepancy, $\delta(t)$, acted as a simple multiplier on the underlying ion channel current. In more complex, kinetic models of drug block, the interactions between model discrepancy and binding dynamics are no longer so straightforward. It therefore becomes less obvious that fitting to the $x(t)/y(t)$ quantity will lead to a cancellation in model discrepancy. To investigate this, we now redefine our synthetic data to be drawn from the following distributions

$$y(t) \sim N(I_{\delta,b=1}(t, V_t), \sigma^2), \quad (3.15)$$

$$x(t) \sim N(I_{\delta,b}(t, V_t, \boldsymbol{\theta}), \sigma^2). \quad (3.16)$$

Here $I_{\delta,b=1}$ is the ‘true’ data-generating current, which may differ in some way (e.g. a different parameterisation or model structure) from our best guess model of underlying ion channel current, $I_{b=1}$. We can think of $I_{\delta,b=1}$ as an analogue to $\delta I_{b=1}$ in the conductance block model example. $I_{\delta,b}$ is the data-generating current in the presence of some drug compound and

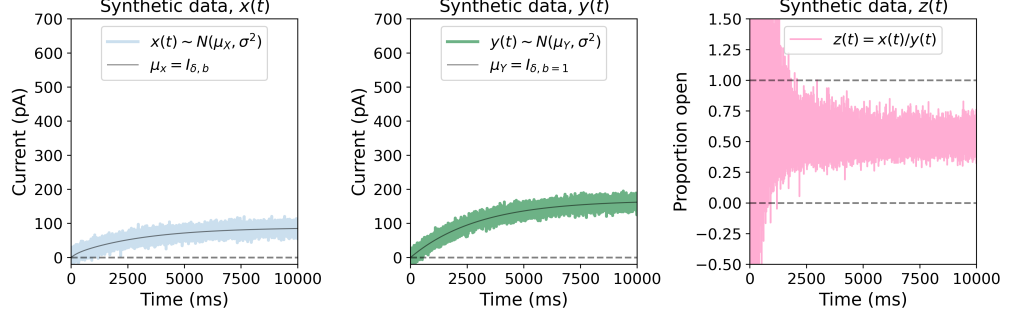


Figure 3.11: Synthetic data generated using the kinetic model of block with 33°C hERG parameters from Lei et al. (2019a). Subplots are similarly defined as in Figure 3.2.

is an analogue to δI_b in the conductance block case. To generate synthetic data, we use the same hERG model as illustrated in Figure 3.1; however, we now use the 33°C mean parameters from Lei et al. (2019a). We then use the methods described in Section 3.3 to generate this new synthetic data, which we plot in Figure 3.11. We now assume that, when fitting a binding model to this data, we know the ‘true’ hERG model structure, but we have an inaccurate parameterisation of this hERG model. Specifically, we assume that our best-guess hERG model parameters are the 37°C parameters used previously, as opposed to the data-generating 33°C parameters. With this model discrepancy introduced, we can then try fitting our binding model parameters to the $x(t)$ data by maximising the log-likelihood defined in Equation (3.6). The resultant fitted model and parameters are plotted in Figure 3.12. The fit is clearly unsatisfactory; the fitted model parameters are particularly far from the data-generating parameters ($\hat{\theta} = [0.032 \text{ ms}^{-1} \text{ nM}^{-n}, 0.99 \text{ ms}^{-1}]$ vs. $\theta = [2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2 \times 10^{-3} \text{ ms}^{-1}]$) and the fitted blocking dynamics plotted in the middle subplot of Figure 3.12 are significantly different from the data-generating dynamics. In the left subplot of Figure 3.13, we plot a heatmap of the log-likelihood surface. We note that the plotted 95% confidence ellipse does not contain θ . We next try fitting to $z(t)$, starting with the incorrect assumption of

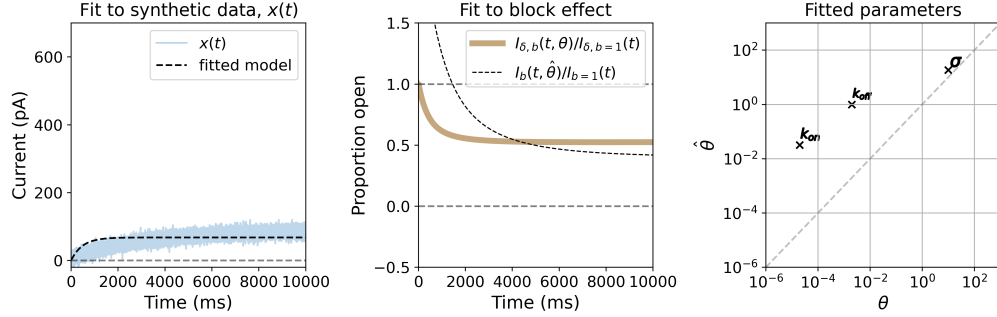


Figure 3.12: Model and parameter fits when fitting the kinetic model of block to $x(t)$ in the presence of model discrepancy. The data-generating hERG model parameters are the 33°C mean parameters from Lei et al. (2019a), while we use the 37°C parameterisation when fitting $\hat{\theta}$. We use the Gaussian log-likelihood defined in Equation (3.6) to fit $\hat{\theta}$. Left: Model fit (dashed line) to $x(t)$ data (noisy blue trace). Middle: Comparison of blocking dynamics between model fit (black dashed line) and data-generating dynamics (brown solid line). Right: Fitted parameters, $\hat{\theta}$, vs. data-generating parameters, θ .

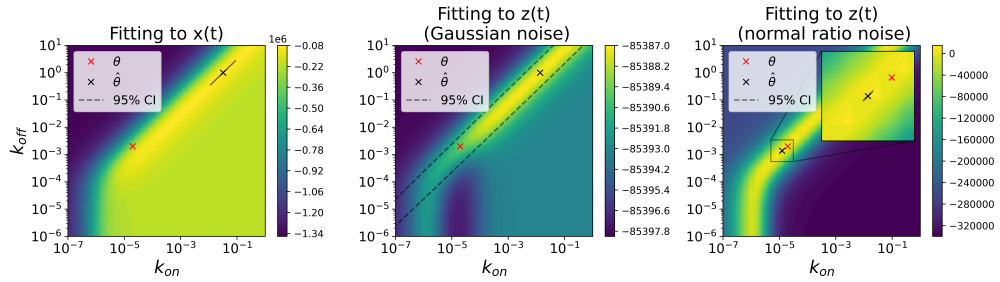


Figure 3.13: Log-likelihood heatmaps for the kinetic model of block in the presence of model discrepancy. The data-generating hERG model parameters were the 33°C mean parameters from Lei et al. (2019a), while we use the 37°C parameterisation to calculate the log-likelihood and fit $\hat{\theta}$. 95% confidence ellipses are calculated via finite difference FIM approximation and plotted as grey dashed lines. Subplots are similarly defined as in Figure 3.4.

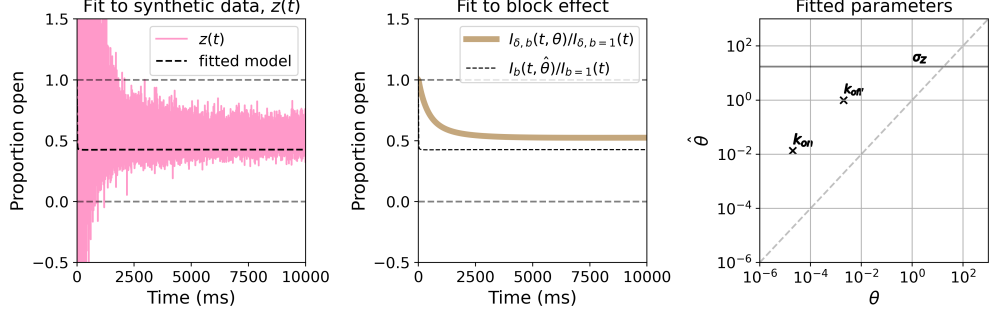


Figure 3.14: Model and parameter fits when fitting the kinetic model of block to $z(t)$ in the presence of model discrepancy. The data-generating hERG model parameters are the 33°C mean parameters from Lei et al. (2019a), while we use the 37°C parameterisation when fitting $\hat{\theta}$. We use the Gaussian log-likelihood defined in Equation (3.17) to fit $\hat{\theta}$. Left: Model fit (dashed line) to $z(t)$ data (noisy pink trace). Middle: Comparison of blocking dynamics between model fit (black dashed line) and data-generating dynamics (brown solid line). Right: Fitted parameters, $\hat{\theta}$, vs. data-generating parameters, θ . The interpretation of the ‘data-generating’ noise parameter, σ_Z , is not obvious given that the data $z(t)$ are not generated with Gaussian noise, so we simply plot a vertical line to indicate the fitted value $\hat{\sigma}_Z$.

Gaussian noise. The log-likelihood for this case requires a slight adjustment from Equation (3.10) and can be defined as

$$\ell(\theta, \sigma_Z | z(t)) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma_Z^2) - \frac{1}{2\sigma_Z^2} \sum_{t \in T} \left(z(t) - \frac{I_b}{I_{b=1}} \right)^2. \quad (3.17)$$

In Figure 3.14, we plot the model and parameter fits obtained when maximising this log-likelihood. Once again, the erroneous Gaussian assumption appears to be leading to issues in the fit. We get fitted MLE parameters, $\hat{\theta} = [0.014 \text{ ms}^{-1} \text{ nM}^{-n}, 0.99 \text{ ms}^{-1}]$, that differ significantly from the data-generating θ , and the fitted blocking dynamics do not reflect the true dynamics. The likelihood surface in the middle subplot of Figure 3.13 is fairly flat, and so θ is contained within the approximated FIM 95% confidence ellipse. However, this broad confidence ellipse is unhelpful with regard to accurate parameter estimation. Therefore, as in the conductance block example, we turn to the normal ratio error model. We use the same PDF as defined in Equation (3.13), but we now have $\beta = I_b(t, \theta)/I_{b=1}(t)$ and

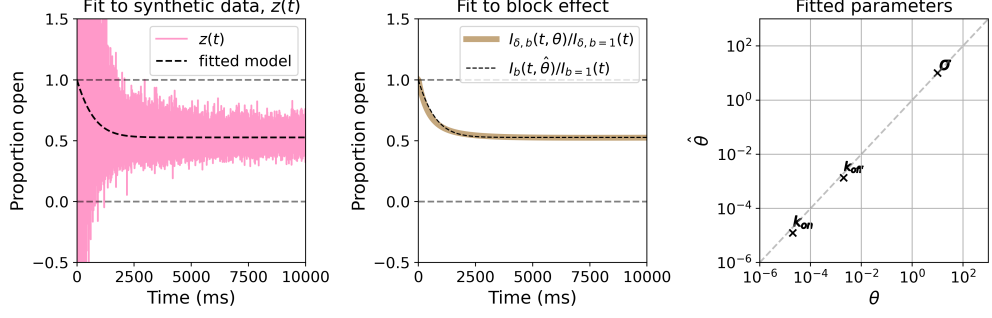


Figure 3.15: Model and parameter fits when using the normal ratio log-likelihood (3.18) to fit the kinetic model of block to $z(t)$ in the presence of model discrepancy. The data-generating hERG model parameters are the 33°C mean parameters from Lei et al. (2019a), while we use the 37°C parameterisation when fitting $\hat{\theta}$. Subplots are similarly defined as in Figure 3.14.

$\delta_y = \sigma/I_{\delta,b=1}(t)$. Our log-likelihood, therefore, becomes

$$\ell(\boldsymbol{\theta}, \sigma | z(t)) = \sum_{t \in T} \ln \left[f_Z(z(t); \frac{I_b(t)}{I_{b=1}(t)}, \tilde{\delta}_y(t)) \right]. \quad (3.18)$$

As previously noted, $\tilde{\delta}_y(t)$ requires an approximation of $I_{\delta,b=1}(t)$ acquired via spline fitting or an equivalent method. We again use the `scikit-fda` package to fit a basis spline with 4th-order polynomials. Given the continuity of $I_{\delta,b=1}(t)$, in contrast to the conductance block example, we simply fit a single basis spline with three evenly spaced knots. In Figure 3.15, we plot the model and parameter fits obtained using the normal ratio log-likelihood defined in Equation (3.18). The fit appears to be good, with the data-generating block dynamics closely matched. Our fitted MLE parameters are fairly close to $\boldsymbol{\theta}$, with $\hat{\boldsymbol{\theta}} = [1.25 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 1.37 \times 10^{-3} \text{ ms}^{-1}]$ and the estimated noise parameter $\hat{\sigma} = 10.03 \text{ pA}$ is also close to the truth ($\sigma = 10 \text{ pA}$). In the right subplot of Figure 3.13, we note that there is a slight difference between $\boldsymbol{\theta}$ and $\hat{\boldsymbol{\theta}}$, and the approximated 95% confidence interval does not contain $\boldsymbol{\theta}$. However, given the presence of model discrepancy and the potential errors introduced by the spline-fitting step, the model fit is particularly good, and a clear improvement on fitting to either

$x(t)$, or $z(t)$ with a Gaussian noise assumption.

3.7 Discussion

In this chapter, we have investigated the intuition that fitting models of drug binding mechanism to the ratio of drug and control currents ($z(t) = x(t)/y(t)$) is preferable to fitting to just the drug currents ($x(t)$). We have employed two example models of differing complexity: a conductance block model and a kinetic model of block; and have considered how discrepancies in the underlying ion channel model and the noise model can influence parameter estimation. When no model discrepancy is present, we have observed no substantial differences between model fitting methods. For each of the example models, the parameter estimates obtained when fitting to $x(t)$ and $z(t)$ were both very close to the true data-generating parameter values. Even when the incorrect Gaussian noise model was used when fitting to $z(t)$, reasonable parameter estimates were returned, and the true parameters were contained within the approximated 95% confidence ellipses. However, in a real-world scenario, we can always expect some degree of model discrepancy. It, therefore, becomes difficult to draw any practical conclusions from a non-discrepant case.

To incorporate discrepancy in our conductance block example, we introduced a piecewise function, $\delta(t)$, that acted multiplicatively on the channel current. Despite the introduction of $\delta(t)$, the parameter estimates obtained when fitting to $x(t)$ were still not far from the true data-generating parameters ($[4.45 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 3.11 \times 10^{-3} \text{ ms}^{-1}]$ cf. $[2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2 \times 10^{-3} \text{ ms}^{-1}]$). However, as illustrated in Figure 3.8, the true parameters were not contained within the 95% confidence ellipse around our MLE, and com-

parisons between $b(t, \boldsymbol{\theta})$ and $b(t, \hat{\boldsymbol{\theta}})$ in Figure 3.7 showed notable differences. When fitting to $z(t)$, we found that assuming a normal ratio error model resulted in excellent fits, despite the presence of $\delta(t)$ and the necessity for a spline-fitting approximation for δ_y in the normal ratio log-likelihood. In contrast, using the Gaussian error model when fitting to $z(t)$ resulted in poor fits. This is an interesting result given that fits of drug-binding models have previously been performed using a sum-of-squares fitting method that assumes Gaussian noise on the $z(t)$ quantity. This is the case in Lei et al. (2024), although, in that paper, particularly noisy regions of the $z(t)$ trace were cropped out to avoid the noise biasing the fits. While the level and type of model discrepancy defined by $\delta(t)$ potentially causes more issues than would be expected in a real-world example, it does serve as an illustration of how an incorrect Gaussian assumption can lead to large errors in parameter estimates and model fits.

For the kinetic model of drug block, we incorporated model discrepancy via two different parameterisations of the underlying ion channel model. It is worth noting that this does not introduce any discrepancy in model structure. However, it should at least provide an idea of the influence that some discrepancy in these channel dynamics can have, given that the dynamics of the drug-binding model are directly influenced by these very channel dynamics. In this discrepant case, we obtained particularly poor parameter estimates when fitting to $x(t)$, and when fitting to $z(t)$ under the assumption of Gaussian noise. On the other hand, fitting to $z(t)$ with the normal ratio noise model gave reasonably good parameter estimates ($[1.25 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 1.37 \times 10^{-3} \text{ ms}^{-1}]$ cf. $[2 \times 10^{-5} \text{ ms}^{-1} \text{ nM}^{-n}, 2 \times 10^{-3} \text{ ms}^{-1}]$). In contrast to the conductance block model, however, we found that the model discrepancy (and perhaps the necessary approximation for δ_y) resulted in $\boldsymbol{\theta}$ no longer being contained within the 95% confi-

dence ellipse. Nevertheless, we can conclude that the method of fitting to $z(t)$ with a normal ratio noise model is consistently the best of the three considered methods for dealing with ion channel model discrepancy. We note that these results do not consider the inevitable discrepancies in the drug-binding models themselves.

Throughout this chapter, we have made several other simplifications and assumptions for illustrative purposes. Firstly, we have assumed that the noise in the control and drug data is equivalent (i.e. $\sigma_X = \sigma_Y = \sigma$). This is a reasonable assumption, but an obvious extension would be to consider how differences in σ_X and σ_Y affect the results of the above experiments. The toy examples considered have also been relatively simple, and other possible extensions could include working with synthetic data more reflective of what we would collect in a real-world scenario. This could involve using multiple drug concentrations and multiple sweeps of the applied voltage protocol. Finally, we could also consider other drug-binding models with more parameters, as well as looking at different discrepancy effects.

Chapter 4

Developing an OED methodology to assist in binding model selection

Preamble

In this chapter, we introduce a methodology for optimising experimental voltage protocol designs to produce data that enable us to distinguish between different proposed models of drug-specific hERG binding mechanisms. We demonstrate the performance of this methodology via a synthetic data study. If the underlying model of hERG current is known exactly, then the optimised protocols show noticeable improvements in our ability to select the true model when compared to a simple protocol used in previous studies. However, if the model is not known exactly, and we assume a discrepancy between the data-generating hERG model and the hERG model used in fitting the models, then the optimised protocols become less effective in determining the ‘true’ binding dynamics. While the

introduced methodology shows promise, we must be careful to ensure that, if applied to a real data study, we have a well-calibrated model of hERG current gating. The majority of the content in this chapter was published in Royal Society Philosophical Transactions A (Patten-Elliott et al., 2025).

4.1 Introduction

As described in Section 2.1.3 in Chapter 2, the hERG channel is susceptible to block by pharmaceutical drug compounds which can lengthen the cardiac action potential, and, in some cases, increase the risk of cardiac arrhythmia (Heist and Ruskin, 2010; Mirams et al., 2012; Hill et al., 2016). To model the interactions of drug compounds with an ion channel, additional states and rates can be introduced to an existing ion channel model to simulate various binding mechanisms (Mirams, 2023). However, the binding mechanisms in the hERG channel can be compound-specific (Redfern et al., 2003; Hancox et al., 2008; Perry et al., 2010). This motivates determining the specific mechanisms at play to assist in modelling how a given compound binds to hERG.

In a recent study, Lei et al. (2024) fitted the set of fifteen pharmacological models shown in Figure 2.6 to previously collected voltage-clamp data (Li et al., 2017) under a modified version of the voltage protocol proposed by Milnes et al. (2010). We will refer to this protocol as the *Milnes* protocol from now on. The binding models in Figure 2.6 represent possible drug-binding mechanisms (e.g. trapping/non-trapping, state binding preference, etc.). After fitting the set of models, it was suggested that more information-rich experiments may be needed to distinguish between model outputs and to assist in determining compound-specific binding mecha-

nisms.

As detailed in Section 2.1.8, previous studies on drug-binding dynamics have considered ‘manual’ experimental design techniques to increase the information extracted from voltage-clamp experiments (Yao et al., 2005; Milnes et al., 2010; Di Veroli et al., 2013; Gomis-Tena et al., 2020; Escobar et al., 2022). This often involves some degree of expert knowledge to design protocols that are expected to emphasise particular compound-specific behaviours. Here, we instead consider ‘automated’ optimal experimental design (OED) techniques. These OED methods have been used recently in the field of cardiac modelling with some success (Lei et al., 2023; Mirams et al., 2024). In this chapter, we consider OED methods to design voltage protocols that can be used in voltage-clamp experiments to better distinguish between different models of drug-binding mechanisms. We detail a synthetic data methodology for generating an optimised protocol and fitting models to data collected under this protocol. Our results demonstrate how the optimised designs can assist in establishing the true binding dynamics at play across a range of simulated drug compounds exhibiting differing dynamics. However, we find that introducing a *discrepancy* between the hidden ‘true’ data-generating model and the proposed model we work with to fit the data, reduces the effectiveness of this method, and suggests that further work may be needed to account for these inevitable model discrepancies when working with real data.

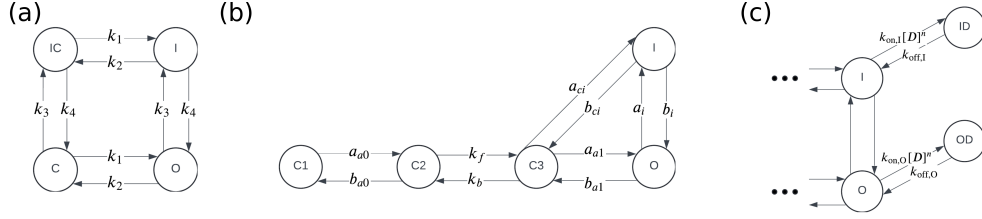


Figure 4.1: Markov diagrams of physiological hERG models (a, b) and an example pharmacological binding model (c). The model in (a) is a four-state symmetric hERG model equivalent to that described in Figure 2.5c. The model in (b) is a five-state model equivalent to that derived by Lu et al. (2001). This model has three closed states (C1, C2, and C3) as well as O (Open) and I (Inactive) states. The pharmacological drug-binding in (c) is model 7 as described by Lei et al. (2024) and included in Figure 2.6.

4.2 Mathematical models

4.2.1 hERG physiological models

We consider two physiological models of hERG: one simple four-state model used throughout, and a slightly more complex five-state model exhibiting differing behaviour that is used when we introduce hERG model discrepancy. These models describe the voltage-dependent gating behaviour of the hERG channel at physiological temperature when no drug compound is present. Figure 4.1 shows Markov diagrams of the two hERG models we will be considering. Figure 4.1a shows the four-state model, with transition rates k_1 to k_4 , which is equivalent to the model described in the Markov diagram in Figure 2.5c. As per the previous chapter, we will henceforth refer to this model as the Beattie model. Figure 4.1b shows the five-state model from Lu et al. (2001), with transition rates a_{a0} , b_{a0} , k_f , k_b , a_{a1} , b_{a1} , a_{ci} , b_{ci} , a_i , and b_i . Referred to from now on as the Lu model, this model has three closed states (C1, C2, C3), and no states that are both inactive and closed. All transition rates in both models (apart from k_f and k_b) are dependent on transmembrane voltage, V ; they are defined by the general

equation $\alpha_i \exp(\beta_i V)$ as per Equation (2.12) in Chapter 2. In both models, the rate of change of each state over time, t , is defined by a differential equation. For example, for the Beattie model, the rate of change of open proportion is defined by

$$\frac{dO}{dt} = k_1 C + k_4 I - (k_2 + k_3) O. \quad (4.1)$$

The measured hERG current, I_{Kr} , is then calculated for both models via the following equation

$$I_{Kr} = g_{Kr} \cdot O \cdot (V - E_K), \quad (4.2)$$

where g_{Kr} is hERG channel conductance, O is the open state proportion, and E_K is the Nernst potential. Full model equations for the Beattie and Lu models are included in Appendix A.

4.2.2 Pharmacological binding models for hERG

We can extend the Markov models of hERG gating to model the mechanisms by which drug compounds bind to hERG. As previously described in Section 2.2.3 and illustrated in Figure 2.6, we will consider a set of fifteen models that characterise different candidate drug-binding dynamics. Proposed by Lei et al. (2024), these models include a variety of trapping and state-specific binding mechanisms intended to capture a large range of potential dynamics. In Figure 4.1c, we include model 7 as an example of one of the fifteen binding models from Figure 2.6. In Appendix A, we include model equations for all fifteen binding models considered.

4.3 Introducing the OED methodology

4.3.1 Initial synthetic data

To measure the effect of drug block on the hERG channel, we can collect two sets of voltage-clamp time-series data; the control current, $y(t)$, before the drug compound is introduced, and the current in the presence of the drug compound, $x_c(t)$ (often at several different concentrations $c \in C$). We begin by generating synthetic ion channel drug-binding data in a form resembling what we would expect to collect in a voltage-clamp experiment under the Milnes protocol. With this synthetic data, we will fit the set of fifteen drug-binding models from Figure 2.6, and illustrate a need for a more complex protocol to differentiate between these models. The process by which this synthetic data is generated is described as follows. All electrophysiology simulations are performed in Myokit (Clerx et al., 2016).

In Figure 4.2 we plot the Milnes protocol, i.e. controlled voltage time-series (left), synthetic current data (top middle), and synthetic open proportion data (bottom middle). In the synthetic current plot, the control current in black, $y(t)$, is generated under the Beattie model with parameters for physiological temperatures taken from Lei et al. (2019a). We run ten sweeps (repeats in series) of the Milnes protocol and plot the 10 s of each sweep corresponding to the 0 mV pulse in the protocol. Ten sweeps are generally sufficient to observe longer time-scale blocking effects. We generate equivalent currents in the presence of four concentrations of verapamil ($x_c(t), c \in C = \{30, 100, 300, 1000\}$) and these are plotted in blue (30 nM), orange (100 nM), green (300 nM), and red (1000 nM). To generate noisy data similar to real experimental patch clamp data, we add Gaussian random noise to all sweeps with a standard deviation of 10 pA. Similarly to

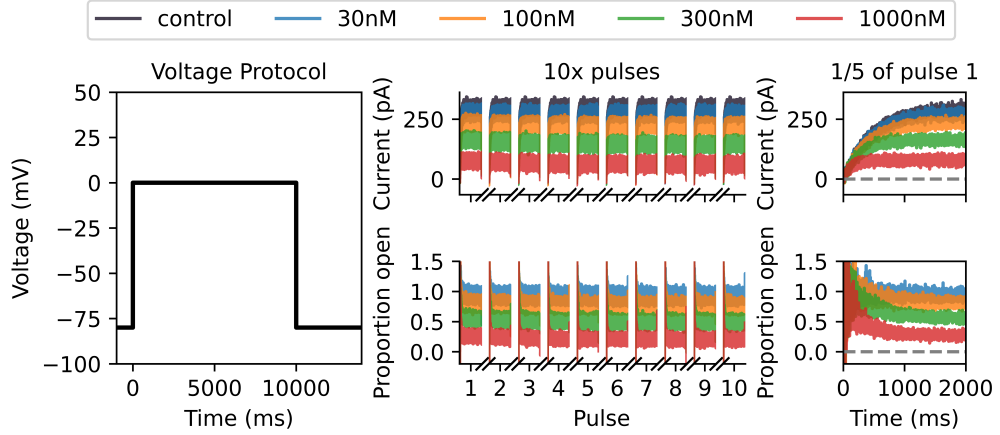


Figure 4.2: Synthetic verapamil data under drug-binding model 7 generated under a Milnes protocol with the Beattie 37°C hERG model. On the left, we plot a single sweep of the Milnes protocol. In the top middle, we plot the control (in black) and drug currents (blue, orange, green, and red corresponding to four different drug concentrations) for 10 sweeps of the Milnes protocol. We only plot the currents that occur during the 10 s pulse at 0 mV for each sweep. In the bottom middle, we plot the corresponding proportion open for each drug concentration, which is calculated by dividing each drug sweep by the control current. On the right, we include a zoomed-in look at the first fifth (2 s) of the first pulse for both currents and proportion open. This illustrates the currents starting at approximately 0 pA, and the open proportion starting at approximately 1. The noise in the initial low currents contributes to the increased noise at the beginning of the open proportion sweeps. This motivates using a noise model for the ratio of two normally distributed variables when fitting drug-binding models to this data.

automated patch clamp sampling frequencies, we use a step size of 0.5 ms between data points, defining T as the set of all times for the plotted traces. The plotted proportion of channels open, $z_c(t)$, in the bottom middle of Figure 4.2, is calculated by dividing the drug current sweeps, $x_c(t)$, by the control sweep, $y(t)$, and this is the normalised quantity we will use to fit the drug-binding models, as in Li et al. (2017, 2019), and motivated by the results of Chapter 3.

As an example for many of the figures in this chapter, for our ‘true’ data-generating model of drug-binding dynamics, we have used the drug-binding model 7 from Figure 4.1c with parameters estimated from verapamil voltage-clamp data collected at physiological temperatures by Li et al. (2017, 2019). Lei et al. (2024) found that this model gave ‘plausible’ fits to the Li et al. data and agrees with the literature that verapamil does not tend to become trapped (Windley et al., 2017). This non-trapping behaviour is represented by the lack of a bound closed state in the binding model; the drug can only be bound and block the channel when the channel is in an open (OD) or inactive (ID) state. In Section 4.4, we will go on to examine findings for different data-generating binding models and different drugs.

4.3.2 Initial model fitting

Next, we fit each of the fifteen drug-binding models to the synthetic ‘proportion open’ data, $z_c(t)$. As a first pass, we assume that the underlying Beattie hERG model is known exactly (i.e. there is no hERG model discrepancy and the model parameters are known) and we wish to fit only the parameters of the drug-binding models. The data, $z_c(t)$, are generated by dividing two current traces which both have normally distributed, inde-

pendent and identically distributed (i.i.d.) noise, i.e. the data are a ratio of two normally distributed random variables. As detailed in the previous chapter, we can use a normal ratio probability density function to derive a suitable log-likelihood when our data is in this form. If $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$ are two independent normal random variables, then the ratio $Z = X/Y$ has the PDF, $f_Z(z; \beta, \rho, \delta)$, as introduced in Equation (3.11). Recall that $\beta = \mu_X/\mu_Y$, $\rho = \sigma_X/\sigma_Y$, and $\delta = \sigma_Y/\mu_Y$. For our synthetic $z_c(t)$ data, $\mu_X(t, c, \boldsymbol{\theta})$ is the modelled current at time t in the presence of a drug compound of concentration c , under some drug-binding parameterisation $\boldsymbol{\theta}$, while $\mu_Y(t)$ is the control current at time t and does not depend on $\boldsymbol{\theta}$ or c . We will simplify things by assuming that σ_X and σ_Y , the standard deviations of the measurement error on the drug and control currents, respectively, are equal, $\sigma_X = \sigma_Y = \sigma$, resulting in $\rho = 1$. We also approximate the μ_Y term in δ by fitting basis splines, as per sections 3.5 and 3.6 in the previous chapter. We can then use the PDF in Equation (3.11) to derive a log-likelihood function for some drug-binding model parameterisation $\boldsymbol{\theta}$ and standard deviation σ given a data set z

$$\ell(\boldsymbol{\theta}, \sigma | z) = \sum_{c \in C} \sum_{t \in T} \log \left(f_Z(z_c(t); \beta(t, c, \boldsymbol{\theta}), 1, \delta(t, \sigma)) \right). \quad (4.3)$$

Each of our fifteen drug-binding models can then be fitted to $z_c(t)$ by maximising this log-likelihood function with respect to $\boldsymbol{\theta}$ and σ . As per the results of Chapter 3, we expect that parameter estimation will be more robust to model discrepancy when using this log-likelihood as opposed to using a Gaussian log-likelihood, or fitting to $x_c(t)$.

We use the CMA-ES optimisation algorithm (Hansen, 2006) via the Probabilistic Inference on Noisy Time-Series (PINTS) framework (Clerx et al., 2019b) to maximise Equation (4.3). As noted in Section 2.3.2, CMA-ES

has previously been successfully used to fit ion channel models (Johnstone et al., 2016; Beattie et al., 2018), and was also used by Lei et al. (2024) to fit drug-binding models. We repeat the CMA-ES optimisation ten times, with each repeat starting from a different parameter initialisation sampled from wide boundaries as described in Lei et al. (2024), and take the largest obtained log-likelihood. The choice of ten repeats is motivated by a desire to balance computation time with accuracy; on average, eight of the ten repeats give very similar maximised likelihoods and corresponding parameter estimates. In Figure 4.3, we plot the fits obtained via this method for each of the fifteen drug-binding models. It is difficult to visually distinguish between the quality of these fits, and, at a glance, it appears that all models fit the data relatively well.

In the left plot of Figure 4.4, we plot the maximised log-likelihood values obtained when fitting each model under the Milnes protocol. The true data-generating model (model 7) and one with a very similar structure (model 11) have the largest maximised log-likelihoods, while all other models fall within a range of 10^5 below this value. We note that we are fitting each model to very high-resolution data (8×10^4 data points), which gives rise to large log-likelihood values. As illustrated in Section 2.5.3, a traditional model selection method such as the likelihood ratio test or the Akaike information criterion (AIC) may suggest, based on the differences in log-likelihoods, that the data-generating model is chosen over the other models. However, in a real data scenario, we generally have less confidence in the veracity of the model of observed hERG current due to experimental artefacts (Marty and Neher, 1995) that are not captured in the modelled data-generating process the likelihood represents. We, therefore, avoid using likelihood ratio testing or AIC in this context. Ultimately, we arrive at the same conclusion drawn by Lei et al. (2024); this protocol is

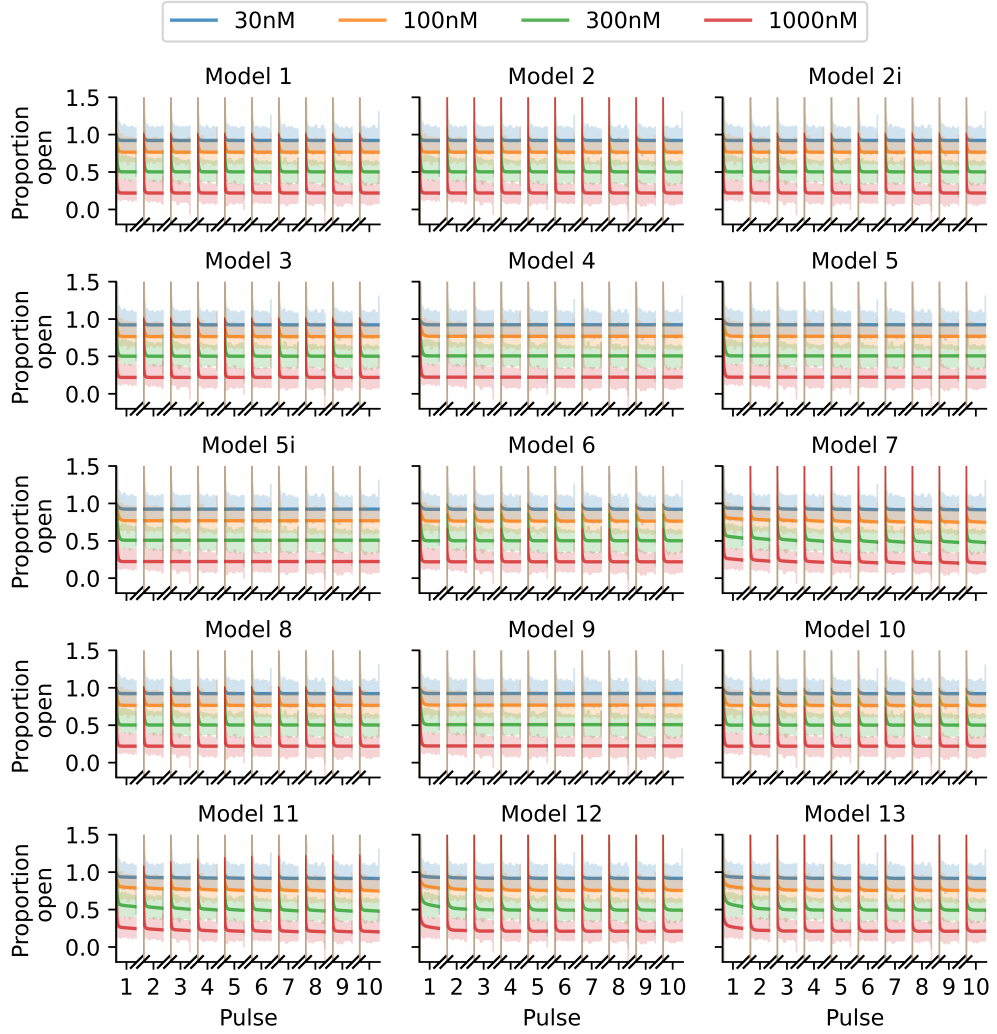


Figure 4.3: Fits of the fifteen binding models to the Milnes protocol proportion-open synthetic data shown in Figure 4.2. Note that it is difficult to visually distinguish between the quality of the model fits for this protocol. The fitted models are shown with solid lines, and we plot the data shaded behind these fits.

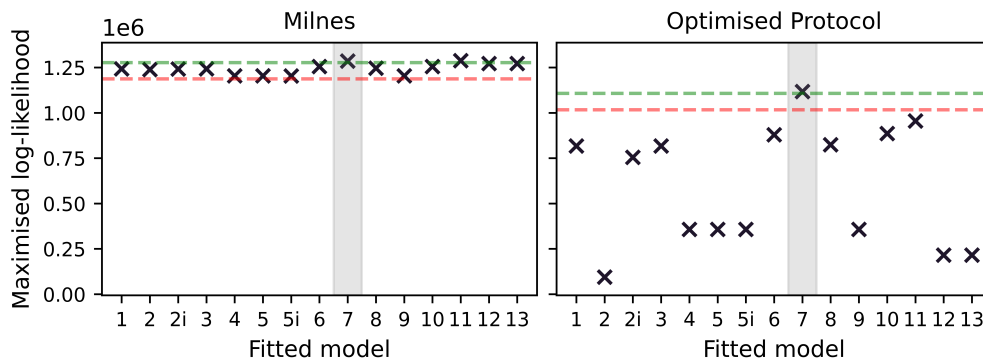


Figure 4.4: Maximised log-likelihoods for model fits to synthetic data generated under the modified Milnes protocol (left) and an optimised protocol (right). In each plot, we include a green dashed line at 10^4 below the largest maximised log-likelihood and a red dashed line at 10^5 below the largest maximised log-likelihood. We note that these lines act simply as visual aids to emphasise the increased spread in the quality of model fits under the optimised protocol. We also shade, in grey, around the true data-generating binding model.

not information-rich enough to distinguish between drug-binding models, which motivates our use of OED methods.

4.3.3 Optimising the experimental design

Our next step is to develop an improved experimental design that emphasises differences between drug-binding models. We will do this by employing OED techniques. As discussed in Section 2.4, OED methods allow us to collect data with ‘optimal information’ based on some user-specified objective function. Let us assume that we have some experimental design, $\mathcal{D}$, that is a function of some vector of parameters ϕ . We then need to establish some optimality criterion, g , that is a function of the design, and determine a ϕ that maximises g . We begin by considering g ; what do we want to optimise? Practically speaking, we want a protocol that accentuates the differences between the models for a specific drug compound. This motivates the following optimality criterion.

Defining an optimality criterion

For the model fits from Figure 4.3, let us define the fitted current as $\mu_{X,m}(t, c, \hat{\boldsymbol{\theta}}_m, \boldsymbol{\phi})$ for $m \in M$ where M is the set of fifteen drug-binding models and $\hat{\boldsymbol{\theta}}_m$ is the MLE parameter set for model m . Note we have included $\boldsymbol{\phi}$ as an input to $\mu_{X,m}$ here, as $\mu_{X,m}$ is dependent on the experimental design. We can then calculate the pairwise squared difference between each of the m model outputs, which we will label d_{ij} , where

$$d_{ij}(\boldsymbol{\phi}) = \sum_{c \in C} \sum_{t \in T} \left(\mu_{X,i}(t, c, \hat{\boldsymbol{\theta}}_i, \boldsymbol{\phi}) - \mu_{X,j}(t, c, \hat{\boldsymbol{\theta}}_j, \boldsymbol{\phi}) \right)^2. \quad (4.4)$$

The squared difference is chosen here as a measure of the separation between model outputs, and has been previously considered in OED for model discrimination (Michalik et al., 2010; Liu et al., 2025). We propose that our optimality criterion seeks to maximise the median value of $d_{ij}(\boldsymbol{\phi})$ across all $i, j \in M, i \neq j$, which we will call $d_{med}(\boldsymbol{\phi})$

$$\hat{\boldsymbol{\phi}} = \max_{\boldsymbol{\phi}} d_{med}(\boldsymbol{\phi}). \quad (4.5)$$

We will refer to this optimality criterion as the *max-med* criterion. There are many choices of design criteria to optimise. A conventional option is the T-optimality criterion introduced by Atkinson and Fedorov (1975), which involves maximising the *minimum* pairwise difference. However, in our case, with some pairs of very similar (or, as we will see in subsequent sections, potentially identical) models, initial experimentation with a T-optimality approach often resulted in the two most similar models being separated slightly, while the optimiser would ignore the pairwise differences between the other thirteen models. Here, we favour the *median* to ensure that the objective will seek to increase the pairwise differences between

at least half of the fifteen models. In a real experimental setting, the aim would then be to reduce the set of feasible models by at least half (via model selection methods or otherwise) and perform another OED and data collection iteration based on the remaining models. In theory, and given sufficient experimental budget, a maximum of four OED iterations would then be necessary to determine the ‘true’ data-generating model (assuming it was contained within the original set of fifteen candidate models).

Optimising the protocol

We now turn our attention to $\mathcal{D}(\phi)$. With consideration for experimental practicality, we will establish a design space that we want to optimise the max-med criterion over. While, in theory, the scope of this design space could be infinitely large, we restrict ourselves to a few degrees of freedom to ensure practical computation times during optimisation. We begin by splitting the ten-second 0 mV pulse used in the Milnes protocol into three separate steps (of lengths 3340 ms, 3330 ms and 3330 ms), each of which can be set to a voltage in the range of -50 to $+40$ mV. This voltage range is selected as a (comfortably) biologically realistic membrane potential range at which we expect to observe non-zero current (and hence drug-binding activity). To increase the design space, we include two 10 s pulses per sweep and allow both pulses to have three different voltage steps each. Further to this, we allow the times between pulses at the holding potential of -80 mV to vary; the time following the first pulse and second pulse can be set to be anywhere between 1050 ms and 21000 ms. Allowing this *inter-pulse* variation should, in theory, reveal any time-dependent unbinding occurring between pulses. In total, we then have eight degrees of freedom for optimising the max-med criterion; six voltage step values and two inter-pulse durations. Let us define $\phi = [V_{11}, V_{12}, V_{13}, V_{21}, V_{22}, V_{23}, t_1, t_2]$ where V_{ij} is

the j th voltage step (in mV) in the i th pulse in the protocol, and t_i is the time (in ms) following the i th pulse in the protocol. We can then optimise the max-med criterion with respect to ϕ via CMA-ES to determine an optimal protocol $D(\hat{\phi})$. To initialise the CMA-ES optimisation, we take 100 random samples from within the voltage and time step bounds defined above ($[-50, +40]$ and $[1050, 21000]$ respectively), and then use the ϕ that gives the largest value of the max-med criterion in Equation (4.5) as our initialisation for the optimiser. We note that finding the global optimum is not necessarily crucial; our goal here is simply to find an improved design that will allow us to discard some models, while striking a balance with computational and experimental practicalities.

4.3.4 Fitting models to synthetic data generated with an optimised protocol

Having generated an optimised protocol, we can then generate a new set of synthetic data using the methodology described in Section 4.3.1. Our new protocol is comprised of two multi-step 10 s pulses of interest, whereas the Milnes protocol has just one single-step 10 s pulse. We, therefore, generate data for only five sweeps (cf. ten) of our new protocol to ensure that the synthetic data under the optimised protocol has the same number of data points as the Milnes synthetic data. In Figure 4.5, we plot this new synthetic current and proportion-open data (right, top and bottom, respectively) alongside the optimised protocol (left). We can then fit the fifteen drug-binding models to this new synthetic data using the same method described in Section 4.3.2; the fits are shown in Figure 4.6. We see that, when compared to Figure 4.3, there are now several models that do not appear to fit the data very well. This is backed up by the right plot in Figure 4.4,

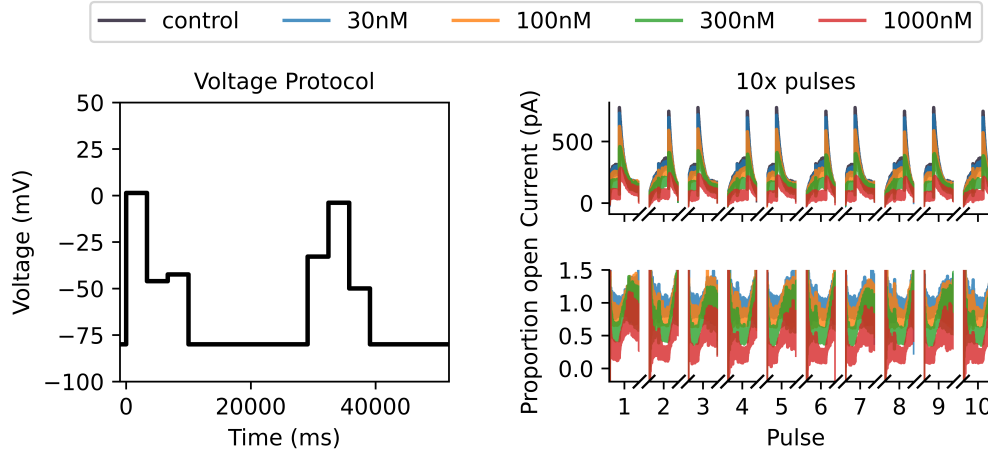


Figure 4.5: Synthetic data for drug-binding model 7 generated under an optimised protocol with the Beattie 37°C hERG model. On the left, we plot our optimised protocol with two 10s pulses each with three optimised voltage steps. In the top right, we plot the control and drug currents for five sweeps of the optimised protocol. We only plot the currents that occur during the two 10s pulses per protocol sweep. The bottom right plot shows the corresponding proportion open for each drug concentration, which is calculated by dividing each drug sweep by the control current.

where we see a greater spread in maximised normal-ratio log-likelihood values (calculated via Equation (4.3)) in the optimised protocol case compared to the Milnes case, making it easier to distinguish between models. Model 7, the ‘true’ data-generating model, has the largest maximised log-likelihood; while no other models have maximised log-likelihoods within 10^5 of model 7. In Figure 4.7 we plot a comparison between the model parameters fitted to the Milnes data, and the model parameters fitted to the optimised protocol data. While the fitted parameters of some models (e.g. 1, 2i, 3, etc.) show little change when fitting to the different datasets, others, including the data-generating model 7, do notably change, highlighting the new information introduced by the optimal protocol data. Figure 4.7 suggests that perhaps performing simultaneous fits to both datasets, forcing parametric compromises, would be a useful additional step to aid model discrimination.

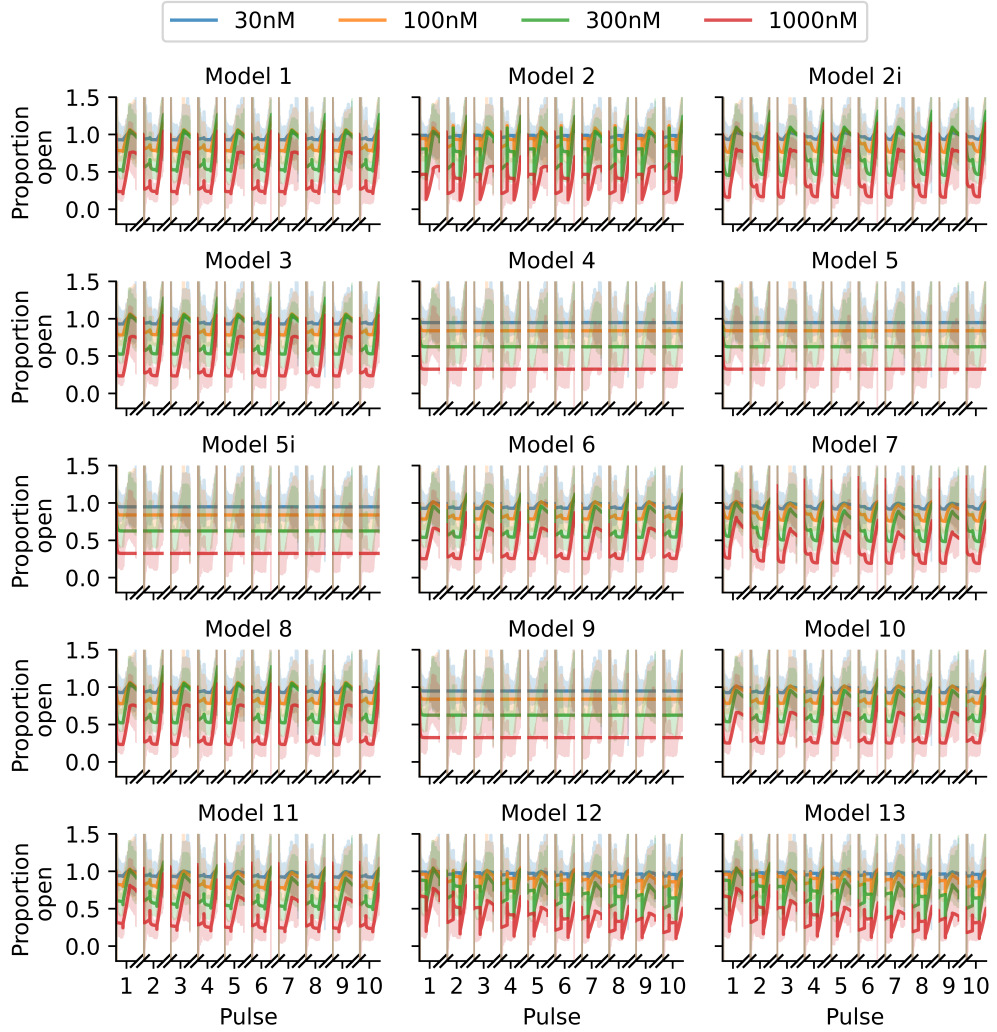


Figure 4.6: Fits of the binding models to the optimised protocol proportion open synthetic data shown in Figure 4.5, in the same style as Figure 4.3. Note how many fits are now visually distinguishable; we can immediately see that many models provide a worse fit to these data than others.

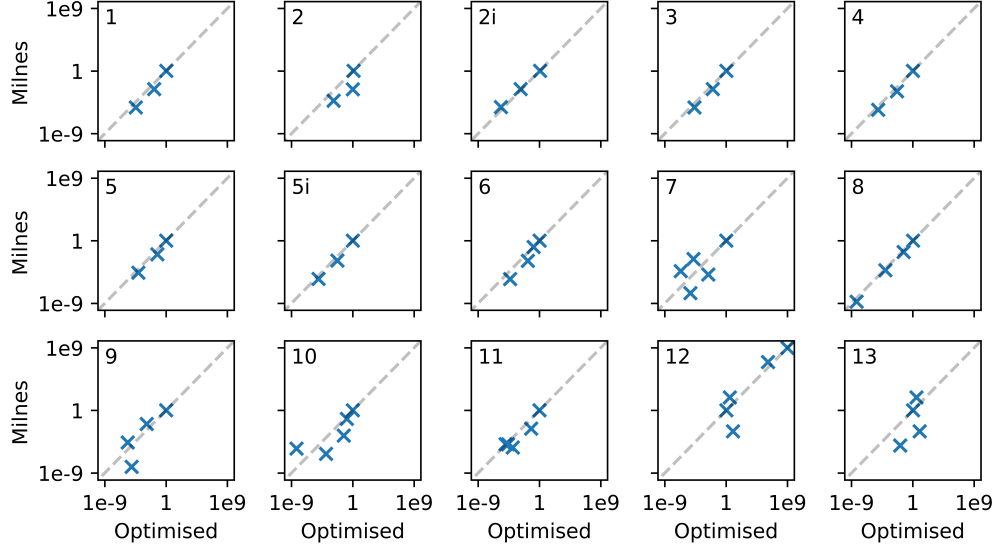


Figure 4.7: Comparison plots of model parameters fitted to Milnes data and optimised protocol data. This is for the verapamil model 7 case, with no model discrepancy. The blue crosses indicate the values of the fitted model parameters for each binding model.

4.3.5 A discrepant hERG model

So far, we have been working under the assumption that the structure and parameterisation of the underlying data-generating hERG model are known exactly. Bernardo and Smith (2009) describe this as an *M-closed* model space, where the true data-generating model is included in the set of models considered for model selection. We can never guarantee this practically, so we now introduce an example where the assumed hERG model used during the model fitting and protocol optimisation steps is different from the hERG model used in the data-generating process. We are now operating in an *M-open* model space where the true data-generating model is not within our set of candidate models.

To generate synthetic data, we now use the Lu model, as illustrated in Figure 4.1b. In Figure 4.8, we plot a comparison between control currents under the Lu model and the 37°C Beattie model previously used to generate synthetic data. We set the conductances (g_{Kr}) for both models to be equal.

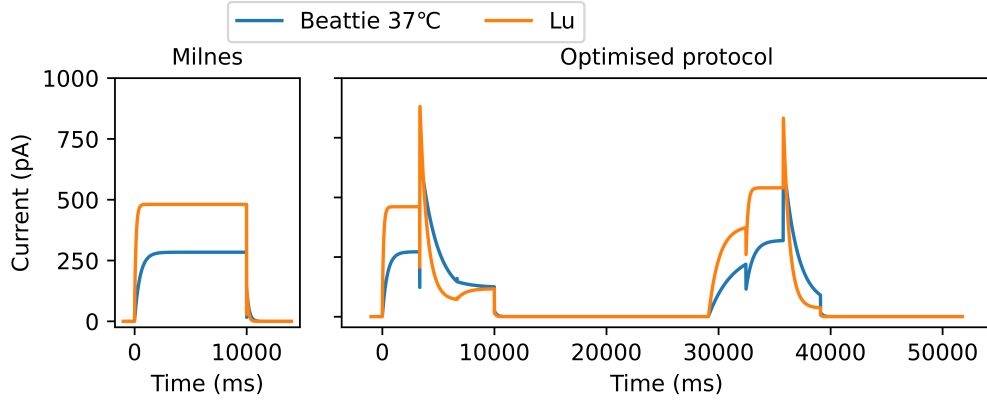


Figure 4.8: A comparison of control currents between the Beattie model (Figure 4.1a) parameterised for 37°C and the Lu model (Figure 4.1b). To the left, we plot the currents in response to one sweep of the Milnes protocol, and to the right, in response to one sweep of the optimised protocol from Figure 4.5. Conductances have been set to 33.3 nS for both models.

We note that the difference in dynamics between these two models appears to be quite significant; for example, there is an approximately 350 ms difference in the 0 mV time constants of activation, τ_a (observed during the Milnes protocol). Sanguinetti and Jurkiewicz (1990) approximated hERG 0 mV τ_a as 50 ms based on experimental data collected under similar experimental conditions (at 35°C). Comparing this experimental estimate to the τ_a approximated for the Beattie and Lu models, we get differences of 400 ms and 50 ms, respectively. The difference between our model dynamics (at least regarding activation times) falls within this range of differences between models and real data. While this model discrepancy is on the larger end of what we would expect from a well-calibrated model of hERG, we consider this a stress test of how our OED methodology performs when there is a relatively large model discrepancy.

We can then repeat the procedure described in sections 4.3.1-4.3.4, but this time using the Lu hERG model when generating any synthetic data (binding model 7 is still used as the data-generating drug-binding model). In Figure 4.9 we plot the log-likelihoods obtained using the Lu model as the

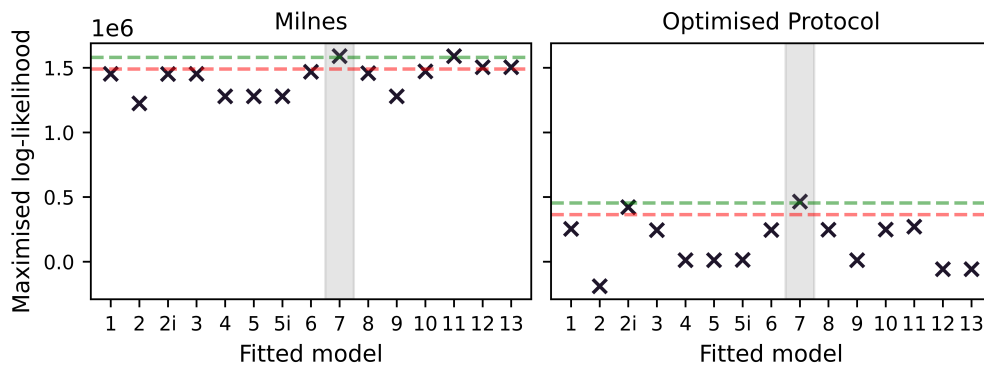


Figure 4.9: Maximised log-likelihoods for model fits to synthetic data generated under the modified Milnes protocol (left) and an optimised protocol (right). This is for the discrepant hERG model case; the data-generating hERG model is the Lu model, while the Beattie 37°C is used for model fitting. In each plot, we include a green dashed line at 10^4 below the largest maximised log-likelihood and a red dashed line at 10^5 below the largest maximised log-likelihood. We also shade, in grey, around the data-generating binding model.

true data-generating hERG model, but (wrongly) assuming that the 37°C Beattie model is the correct hERG model when fitting the drug-binding models to the data. We appear to get similar results to the non-discrepant case, with the optimised protocol once again spreading out the values of the log-likelihoods and pointing towards model 7 as the data-generating model. However, as we will see in Section 4.4, we find these methods are less effective for other data-generating drug-binding models when discrepancy is introduced. In Appendix B we include equivalent plots to Figures 4.2, 4.3, 4.5, 4.6 and 4.7 in Figures B.1-B.5 for the discrepant case.

4.4 Applying the OED methodology

In Section 4.3, we saw how our methodology performed for synthetic data generated under one drug-binding model (model 7) parameterised for one drug compound. We now repeat these methods across a range of models and drugs to demonstrate how the procedure performs in differing circum-

stances.

4.4.1 Verapamil: non-trapping dynamics

We begin by considering verapamil again, but this time we generate synthetic data for each of the fifteen drug-binding models (initially using the Beattie model in Figure 4.1a as our underlying hERG model). Using the previous fits to real verapamil data from Lei et al. (2024) to generate the synthetic data, in the top half of Figure 4.10 we plot heatmaps of the log-likelihoods obtained via the Section 4.3 methodology. This is for the case with no hERG model discrepancy. The y -axis represents which drug-binding model is used to generate the synthetic data, and we then get a corresponding log-likelihood for each fitted drug-binding model on the x -axis. We plot both the log-likelihoods obtained for the Milnes protocol data fits (top left), and for the optimised protocol data fits (top right). For each row, we highlight (with green squares) the fitted models within 10^4 of the largest log-likelihood in that row. We then also highlight (with red squares) the fitted models that are within 10^5 of the largest log-likelihood (that are not already highlighted in green). Black dots are plotted down the diagonal, indicating where the fitted model corresponds to the data-generating model. The first notable observation from this plot is that the number of models within the 10^4 and 10^5 thresholds is significantly reduced in the optimised protocol case compared to the Milnes case. We also observe that, in both cases, the full diagonal is within the 10^4 threshold.

Note that there is a large amount of model nesting between the fifteen drug-binding models, which we illustrate in Figure 4.11. We consider some model, model A, to be nested within another model, model B, if, by fixing one or more parameters in model B, we can reduce model B to model A. For

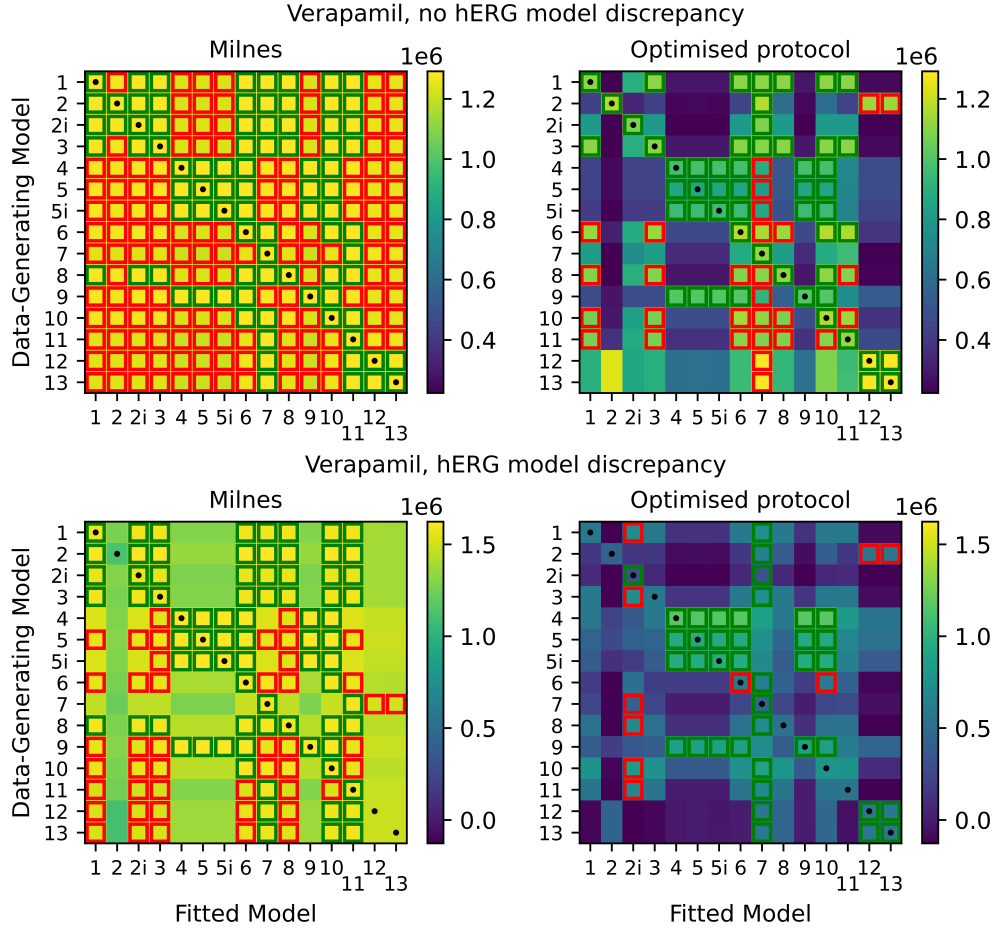


Figure 4.10: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes (left) and optimised protocol (right) synthetic verapamil data. The fitted models within 10^4 of the largest log-likelihood (very good fits) in each row are highlighted in green squares. The fitted models that are between 10^4 and 10^5 below the largest log-likelihood (reasonable fits) in each row are highlighted in red squares. Black dots are plotted down the diagonal of each heatmap where the fitted model corresponds to the data-generating model. Note that our optimal protocol results in a far bigger spread of maximised log-likelihoods. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data. The introduced hERG model discrepancy makes identifying the ‘true’ data-generating drug-binding model difficult.

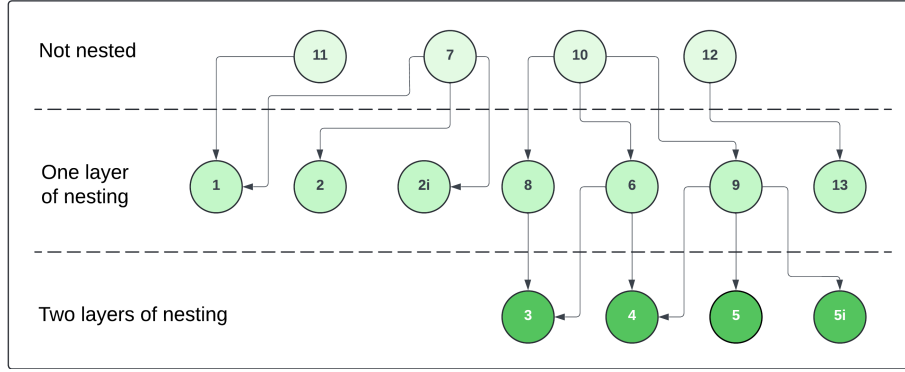


Figure 4.11: Graph illustrating nesting between drug binding models. An arrow pointing from model A to model B indicates that model B is nested within model A.

this reason, we expect that (in this non-discrepant hERG case) if model A is nested within model B, and model A is the true data-generating model, then model B should also be able to fit the data at least equally well (perhaps even slightly better given higher complexity and more parameters), so the maximised log-likelihoods for Model A and B should be roughly equivalent. At the top of the tree diagram are the models that are not nested within any others: models 7, 10, 11 and 12. If we consider the heatmap in the top right of Figure 4.10 corresponding to the optimised protocol, we see that for synthetic data generated under models 7, 10, and 11, the only fitted model with a log-likelihood within the 10^4 threshold is the true data-generating model. Model 12, the other model with no nesting, only has one other model within this 10^4 threshold (model 13). Considering the nested models in Figure 4.11, we can explain some of the cases where multiple models sit within the 10^4 threshold. In the optimised protocol case in the top right of Figure 4.10, the 10^4 threshold highlighted squares in the heatmap rows corresponding to models 2, 2i, 8, and 12 can all be explained by model nesting. On the other hand, data generated under models 1, 3, 4, 5, 5i, 6, and 9 all have at least one model within the 10^4 threshold that cannot be explained by nesting.

We now switch to the discrepant hERG model case, where synthetic data were generated using the Lu physiological hERG model, but we use the Beattie model in fitting. In the bottom half of Figure 4.10, we plot equivalent heatmaps to those at the top of the plot, but for the discrepant case. The purpose of considering this case is to address the more realistic scenario where we do not have a perfect model of hERG channel dynamics. We notice that the number of highlighted 10^4 and 10^5 threshold squares is once again significantly reduced in the optimised protocol case. However, in most cases, the best-fitting binding model is no longer the one that generated the data. As a result, the diagonal is highlighted significantly less than in the non-discrepant case, with only eight models (2i, 4, 5, 5i, 7, 9, 12 and 13) sitting within the 10^4 threshold. Clearly, the hERG model discrepancy is causing issues in identifying the true binding mechanisms.

4.4.2 Bepridil: trapping dynamics

We can repeat the process described in the previous section for a different drug, bepridil, this time with observed trapping behaviour (Kamiya et al., 2006; Stork et al., 2007). In Figure 4.12, we plot examples of model output under different binding dynamics. The left and middle subplots illustrate the differences between non-trapping and trapping dynamics for an example drug. Note how, in the presence of the example trapping drug, the channel does not fully open (i.e. proportion open equal to 1) between pulses.

Considering the trapping drug, bepridil, in the top half of Figure 4.13 we plot the heatmaps of maximised log-likelihoods for the non-discrepant case as previously. The optimised protocol heatmap is very similar to that obtained with verapamil; the highlighted squares in the rows corresponding to models 1, 2, 2i, 3, and 8 are identical, while there are only a couple of

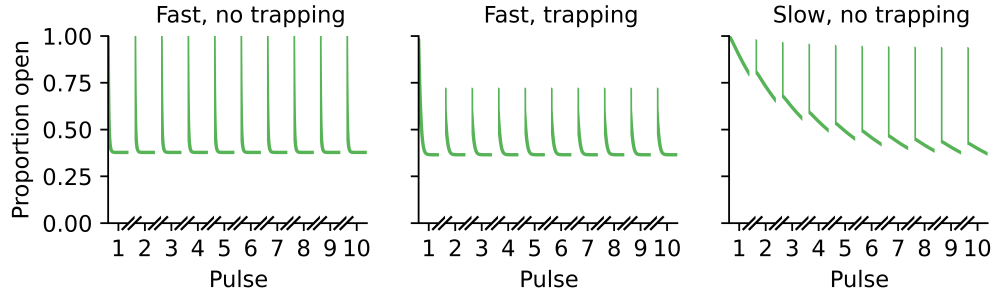


Figure 4.12: Model output for example drugs with different binding dynamics under 10 sweeps of the Milnes protocol. The left plot illustrates fast binding dynamics with no trapping behaviour; the drug quickly reduces the proportion open to approximately 0.4 on each pulse, but returns to being completely open between pulses. The middle plot also shows fast binding, but with trapping behaviour; this time, the channel does not completely open between pulses. In the right plot, we illustrate slow binding with trapping behaviour; about seven pulses of the Milnes protocol are required for the open proportion to reach an equilibrium at approximately 0.4.

differences for each of the other models. We can also again consider the discrepant hERG case, and we plot the heatmaps for this in the bottom half of Figure 4.13. Again, this shows relatively similar results to those seen with verapamil; when we have a discrepancy in the hERG model, it becomes difficult to determine the true data-generating drug-binding model. In Appendix B, we also include heatmaps in Figure B.6 for results obtained for chlorpromazine, a fast-binding drug with a suspected open-binding preference (over inactive-binding). These results appear similar to those obtained for verapamil and bepridil.

4.4.3 Dofetilide: slow-binding dynamics

Bepridil and verapamil both have relatively fast binding dynamics, so we also consider a slow-binding drug, dofetilide (Kiehn et al., 1996). In the left and right subplots of Figure 4.12, we include example model output comparing fast and slow binding dynamics.

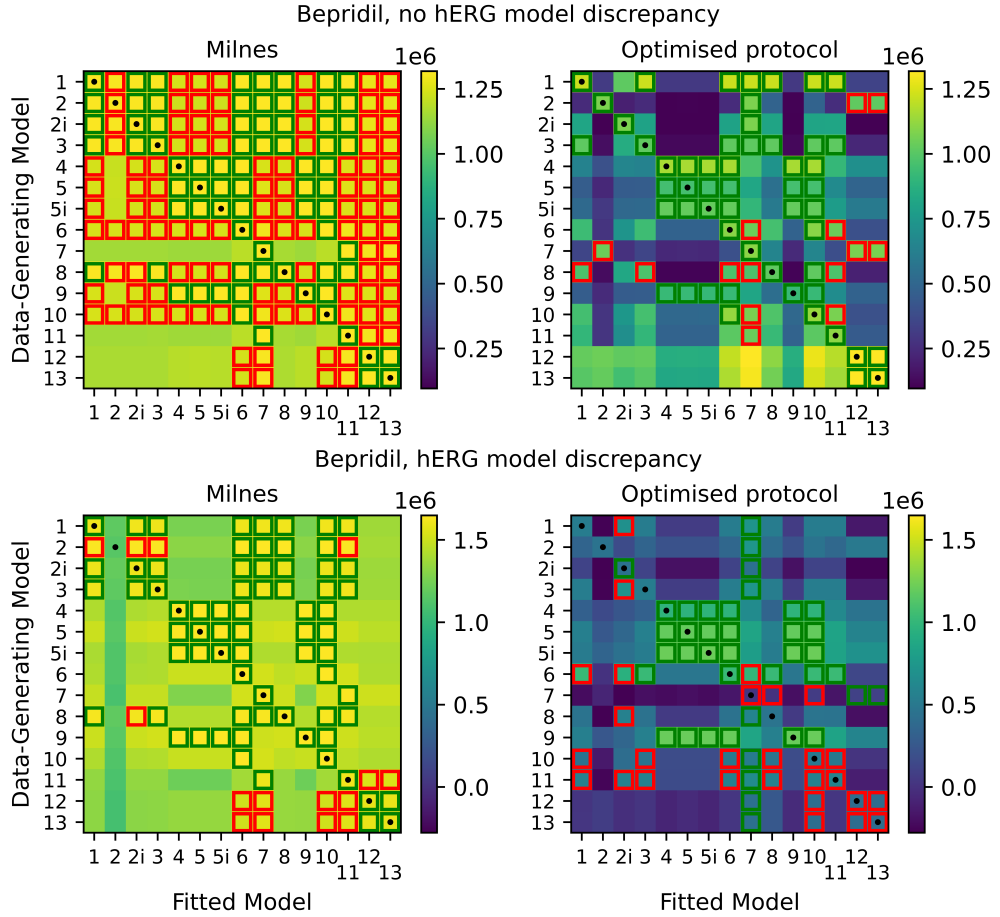


Figure 4.13: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic bepridil data. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data.

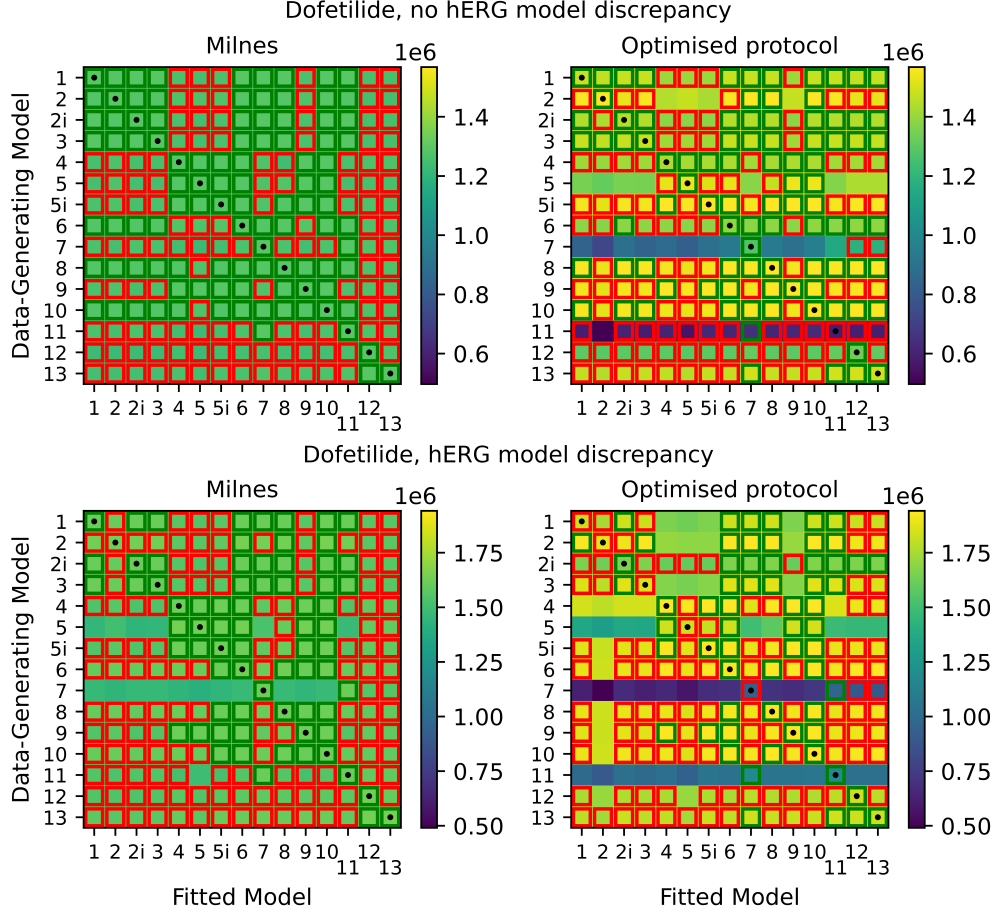


Figure 4.14: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic dofetilide data. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data.

Once again, we plot the heatmaps of maximised log-likelihoods as shown in Figure 4.14, and we see that nearly all models can fit the Milnes data well. Unlike the fast dynamic drugs, with dofetilide, we get much less of a reduction in 10^4 and 10^5 threshold squares with the optimised protocol (in both the non-discrepant and discrepant cases). It is not obvious what causes this difference between fast and slow-binding drugs, and it may indicate that our median optimisation objective function is less effective in the slow-binding case. As we will observe in Chapter 5, extending the design space to allow shorter steps in the voltage protocol appears to aid the OED methodology for slow-binding drugs. In Appendix B, we also include,

in Figures B.7-B.10, heatmaps for each of the four considered drugs for the case where all model fitting is done assuming the Lu model is the true model of hERG dynamics (as opposed to the Beattie model). These plots show similar overall results and indicate that the results obtained in the main text are not exclusive to the Beattie hERG model.

4.5 Discussion

In this chapter, we have outlined a methodology for generating optimised voltage protocol designs to assist in distinguishing between models of drug-binding dynamics. By undertaking a synthetic data study, we have seen the potential benefits of this methodology when the true physiological model of hERG is known. Log-likelihoods of models fitted to data generated under our optimised protocols indicate a divergence in the quality of fits when compared to fits to data generated under a simple Milnes protocol. This suggests that this OED procedure could assist in establishing the true binding dynamics of a compound. The method was less effective when considering synthetic data emulating a slow-binding drug, dofetilide, when compared to drugs with faster dynamics such as verapamil and bepridil.

We also considered how discrepancies in the hERG model used to fit the data (compared to the data-generating hERG model) influenced the outcomes of our methodology. We found that when we used the Lu hERG model to generate synthetic data and the Beattie 37°C hERG model to fit models to this data, log-likelihoods of model fits indicated that establishing the true data-generating drug-binding model became more difficult. This suggests that the underlying hERG model does play an important role when fitting drug-binding models to data and stresses the importance

of continuing to improve basic models of physiological channel behaviour. A well-calibrated model of hERG current, that approximately matches the observed dynamics in real experimental data, would reduce the influence of model discrepancy on the outcomes of the OED procedure. In their 2019 paper, Clerx et al. (2019a) detail the benefits and limitations of several methods for calibrating models of ion channels. Our proposed methodology could perhaps be improved by considering fitting a hERG model to control currents before fitting the drug-binding models (or fitting both the hERG and drug-binding models simultaneously).

We note that our results depend on the initial drug-binding model parameterisations for each of the four considered drug compounds, which come from model fits obtained by Lei et al. (2024). The quality of these model fits was quite variable from model to model and from compound to compound. This represents a limitation of a synthetic data study and motivates trialling the methodology on real data.

The methods used to fit the drug-binding models in this chapter were developed to improve on those used by Lei et al. (2024) and others (Windley et al., 2017; Li et al., 2019). While we have similarly fitted our models to the proportion of hERG in the open state, the log-likelihoods derived from the normal ratio PDF described in Equation (3.11) differ from the simple weighted sum of squares method used previously. As discussed in Chapter 3, this ratio likelihood fitting method provides a more realistic noise model for the data-generating process (in this synthetic scenario), avoiding problems associated with small control currents leading to large noise on the ‘proportion open’ trace. In Lei et al. (2024), the weighted sum of squares method required that low currents at the start of each 10 s pulse were cropped out to prevent noisy open proportion data from biasing the fit; our method allows us to fit the binding models to the full data sweep.

After some consideration and testing, the max-med optimality criterion was chosen over other possible alternatives, such as maximising the mean or the minimum of the pairwise sum of squares difference between model current traces. Using the mean or minimum, as opposed to the median, tended to result in one or two model pairwise differences biasing the objective function score, leaving many of the other current traces indistinguishable from each other. It would be useful to perform more rigorous comparisons between optimality criteria to see if there are scenarios where alternatives are more effective.

In Section 4.4.1, we noted the presence of nesting between the fifteen binding models. This nesting suggests that perhaps reducing our optimisation problem to consider only pairwise differences between the non-nested models (7, 10, 11, and 13) could be an alternative starting point, given that all other nested models are simplified versions of these. Another avenue to explore would be the use of multiple different protocols, each optimised to emphasise particular drug-binding properties perhaps, and examining the way different models need to compromise to fit data from each protocol has proved instructive in providing a lower bound on model discrepancy (Shuttleworth et al., 2024).

To conclude, the proposed OED methodology shows promise in determining the true binding dynamics, but care must be taken to ensure that we have a well-calibrated model of control hERG current if applied to a real data study.

Chapter 5

Determining unknown drug binding dynamics using an OED methodology

Preamble

In the previous chapter, we trialled a proposed OED methodology on synthetic patch-clamp data. In this synthetic data setting, the methodology proved effective in distinguishing between a set of candidate models of drug-binding dynamics in the hERG channel. The method was particularly effective when assuming perfect knowledge of the underlying model of hERG gating, while the introduction of discrepancy between the data-generating hERG model and the assumed gating dynamics highlighted the importance of a well-calibrated hERG model. In this chapter, we extend this methodology and apply it to real experimental patch-clamp data to assist in determining the unknown binding dynamics of several drugs. With the assistance of Monique Windley and Mohammed Atif, I ran initial trial

patch-clamp experiments at the Victor Chang Cardiac Research Institute (VCCRI) in Sydney, Australia, during visits in 2023 and 2024. These trial experiments informed the experiments described in this chapter, which were run by Mohammed Atif at the VCCRI between August and November 2025, based on my experimental designs. Using the SyncroPatch 384PE to automate patch-clamp experiments, we collected drug block data for five compounds with known hERG binding affinity. To reduce the impact of any hERG model discrepancy, and in contrast to the previous chapter, in this chapter we additionally fit the parameters of the hERG gating model at each stage of the OED methodology. By optimising drug-specific protocol designs, we are able to substantially reduce the set of feasible models for each drug considered, in doing so, providing new insights and reinforcing previous suggestions as to the likely binding mechanisms of each of the considered drugs. While the OED methodology proves an effective tool for model discrimination, we suggest that a further iteration of experimental design and data collection would be useful to verify any of the drug-specific insights gained.

5.1 Introduction

As described in Section 2.1, a healthy heart relies on regular, coordinated contractions of cardiomyocytes to pump blood around the body (Priest and McDermott, 2015). We are interested in the $K_v11.1$ ion channel encoded by hERG, as it plays a crucial role in cardiomyocyte recovery from excitation. Pharmaceutical drug compounds can bind to the hERG channel, causing an unintended block effect, often resulting in a reduction in I_{K_r} , which can, in turn, lengthen the cardiac action potential and increase the risk of cardiac arrhythmia.

Drug block of hERG can be measured using patch-clamp electrophysiology experiments as detailed in Section 2.1.6. These experiments involve measuring I_{Kr} in response to some voltage protocol, V_m . The block effect of a given drug compound can then be evaluated by comparing I_{Kr} in a control setting under some voltage protocol to I_{Kr} in the presence of the drug compound under the same protocol. The data produced by these experiments can then be used to build, fit, and validate mathematical models of both ion channel currents and drug binding dynamics.

In Chapter 4, we introduced an OED methodology to assist in determining the hERG channel binding dynamics for a given compound, specifically by discriminating between a set of candidate models. To investigate the feasibility of the method, we considered synthetic patch clamp data. We found that the method was effective in separating the quality of model fits, suggesting its use as a practical approach to aid drug binding model selection. In this chapter, we extend these methods and apply them in a real data scenario. Working with real data introduces several additional challenges. Firstly, we must ensure we have effective Quality Control (QC) to filter out data that are particularly noisy, or that represent unphysiological behaviour resulting from experimental errors or artefacts. To ensure we retain sufficient data, several experimental artefacts can be accounted for by introducing post-processing measures that correct for any leak or contaminating currents. Another consideration in working with real data is the variability introduced by both biological and experimental factors. In the previous chapter, we considered synthetic currents intended to represent the behaviour in a single cell. In this chapter, we collect data for several cells for each drug and concentration combination, introducing additional considerations in post-processing and model-fitting procedures.

To test the effectiveness of the OED procedure with real experimental data,

we consider five known hERG-blocking drugs that represent a range of binding dynamics. These include two compounds with slow binding dynamics (bepridil and verapamil) and three compounds with fast binding dynamics (quinidine, diltiazem, and chlorpromazine). For each of these compounds, the OED procedure assists in discriminating between the set of proposed binding models introduced in Chapter 2 and illustrated in Figure 2.6. However, we recommend that an additional iteration of the OED procedure is necessary to verify these results and further rule out remaining feasible models.

5.2 Initial data collection

Before we could generate optimally designed protocols, it was necessary to first retrieve some baseline data to initially parameterise the hERG gating and binding models. Patch-clamp data were collected (by Mohammed Atif at the VCCRI) under a simple modified Milnes protocol as used in the Comprehensive in-vitro Proarrhythmia Assay (CiPA) initiative (Milnes et al., 2010; Li et al., 2019). This protocol is commonly used to measure the effect of drug block and is a sensible choice of baseline protocol design. We now describe the process by which these data were collected.

5.2.1 Cell culture and harvesting

Regarding cell harvesting, the same methods were used as described by TeBay et al. (2022) and Windley et al. (2022). Chinese hamster ovary (CHO) cells stably expressing hERG channels were purchased from the American Type Culture Collection (ATTC reference PTA-681). The cells were cultured in Hams F12 nutrient mix (ThermoFisher Scientific, Waltham,

USA) supplemented with 5% fetal bovine serum (Sigma-Aldrich, Sydney, Australia) and kept in a 37 °C humidified incubator at 5% CO₂. Cells were passaged every 2-3 days and were harvested for experiments ~ 48 hours after passaging. Before harvesting, the cells were grown in t175 tissue culture flasks and were washed twice in phosphate-buffered saline (PBS, Mg/Ca²⁺ free, ThermoFisher Scientific, Waltham, USA) before being incubated with Accumax (Merck Life Science) at 37 °C for five minutes to allow for cell detachment. The cells were then washed in a 4 °C external solution and spun at 250 g for five minutes, before the supernatant was removed and the cells were resuspended in divalent-free SyncroPatch solution to a density of approximately 250,000 cells/ml. This cellular suspension was incubated for a further 45-60 minutes at 4 °C before being transferred to the shaking SyncroPatch cell platform.

5.2.2 High throughput patch-clamp set-up and solutions

All patch-clamp experiments were run using the SyncroPatch 384PE (Nanion, Munich, Germany). We ran 384 whole-cell, patch-clamp recordings simultaneously on a single Nanion, 1 hole, 384T-L type SyncroPatch plate. The SyncroPatch automated all cell catching, sealing, whole-cell breakthrough, and capacitance compensation processes. While performing experiments at physiological temperatures (i.e. ~ 37° C) is the gold standard for obtaining physiologically meaningful results, running SyncroPatch experiments at higher temperatures generally results in a lower experimental success rate. We, therefore, ran all experiments at room temperature (assumed to be 25 ±1° C), and we caveat any insights into drug-binding mechanisms as specific to these room temperature conditions.

5.2.3 Drug compounds

To investigate the effectiveness of the OED methodology (as introduced in Chapter 4 and extended in Section 5.4) in aiding model discrimination in a real data setting, across a range of binding dynamics, we considered five different hERG blocking drug compounds with a range of hypothesised/observed behaviours. We used two slow-binding drugs, bepridil and verapamil, and three fast-binding drugs, quinidine, diltiazem, and chlorpromazine. Given that their binding dynamics occur on longer timescales, it is generally easier to measure the drug-block effect of slower binding drugs, and so we have some prior knowledge of the expected dynamics of bepridil and verapamil. For example, Stork et al. (2007) found evidence of bepridil exhibiting trapping behaviour, while Windley et al. (2017) showed verapamil to be a non-trapping drug. The binding dynamics of the fast-binding drugs are less well known.

For each of the five drugs, we considered three different concentrations chosen to give a low, medium, and high block effect. In total, two SyncroPatch plates were required to get sufficient data for each drug — one plate for the two slow binding drugs, and another for the three fast binding drugs. In Figure 5.1, we include a schematic detailing which wells each of these drugs/concentrations were added to on the SyncroPatch plate. Note, we also included three columns of wells in which we added Dimethyl Sulfoxide (DMSO). Each of our blocking drugs were dissolved in DMSO before they were added to the SyncroPatch plate, so including some wells that contain only DMSO is an effective control case to ensure that any I_{Kr} block-effect observed in the drug wells does, in fact, result from the presence of these drugs.

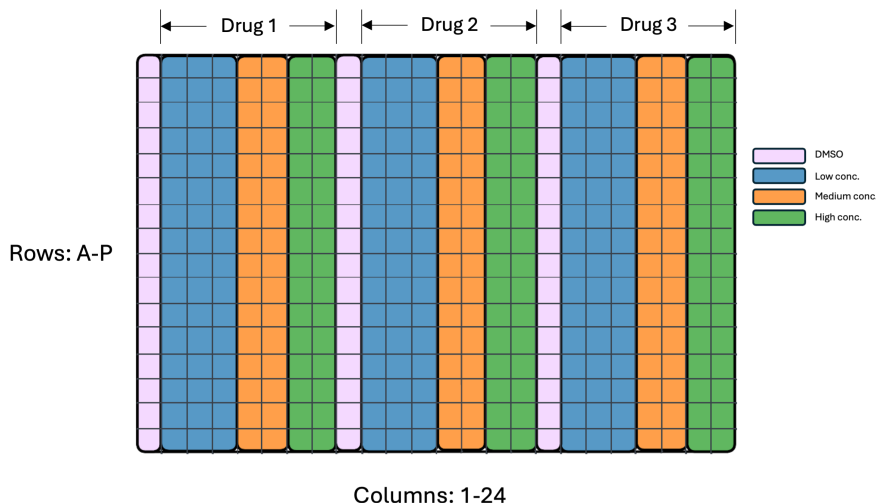


Figure 5.1: Layout of drug compounds added to the SyncroPatch plate. Each plate, containing 384 wells, was divided into three sections corresponding to three different drugs. For each drug, we included three columns (in blue) of a low concentration, and two columns at medium (in orange) and high (in green) concentrations. We also included three columns (in pink) containing DMSO, which acts as a control (non-blocking) compound.

5.2.4 Protocol procedure

We ran the protocol procedure illustrated in Figure 5.2 in each well to provide the data necessary to fit both our underlying hERG model and drug-binding models to isolated I_{K_r} current data. The procedure begins with two repeats of a ‘staircase’ protocol as described in Lei et al. (2019b). In Section 5.3.2 we detail how the hERG currents measured under this protocol can be used to fit the Beattie hERG model parameters. The staircase sweeps are followed by a minimum of five sweeps of the Milnes protocol to give a pre-drug baseline. Next, the drug of interest is added, before a minimum of ten sweeps of the Milnes protocol are run to assess the drug channel block. Dofetilide at $1\mu\text{M}$ is then added, before a wash-in protocol consisting of a regular activation stimulus induces an (almost) full block of I_{K_r} . We then run a further five sweeps of the Milnes protocol. During post-processing, one of these *full-block* Milnes sweeps can be subtracted from the pre-, and post-drug, leak-corrected Milnes sweeps to minimise contamina-

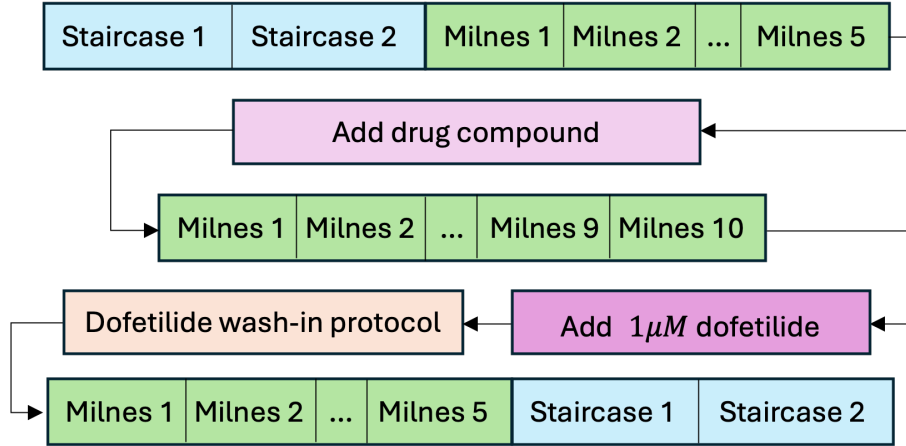


Figure 5.2: Schematic illustrating the protocol procedure used for SynchroPatch experiments. Before adding a drug, we begin with two sweeps of the staircase protocol, followed by five sweeps of the Milnes protocol. Next, we add the drug compound of interest, before running ten sweeps of the Milnes protocol to assess the drug block effect. We then add a high concentration ($1\mu\text{M}$) of dofetilide and run a simple wash-in protocol with a regular activation stimulus to fully block hERG. Finally, we run five more sweeps of the Milnes protocol and two more staircase protocol sweeps. While we specify the Milnes protocol in this schematic, instances of ‘Milnes’ here can be replaced with ‘optimised protocol’ to describe the protocol procedure when working with optimised protocols. Note, however, that for the optimised protocols we run fewer sweeps; three sweeps before adding the drug, six sweeps after the drug is added, and three sweeps after the dofetilide full-block.

tion from any other currents, effectively isolating I_{Kr} . Finally, two further repeats of the staircase protocol are applied, which can be used to perform a *full-block subtraction* on the initial two staircase sweeps. Further details of the leak-correction and full-block subtraction post-processing procedures are included in the following section.

5.2.5 Post-processing and quality control

After collecting patch-clamp SyncroPatch data, the data were subjected to post-processing and QC. As discussed in Section 2.1.7 of the literature review, there are several causes of experimental artefacts in patch-clamp experiments, particularly in an automated setup. Some artefacts, such as those resulting from an unsuccessful cell catch or seal, are unavoidable, and the data recorded in these cases are filtered out during the QC process. However, other artefacts, such as *leak currents* and contaminating background currents, can be accounted for in post-processing. Leak currents are currents that escape the tight seal between the cell membrane and the pipette (in manual patch clamp) or micro-pore (in the SyncroPatch plate). Contaminating currents, on the other hand, are any voltage-dependent currents, other than I_{Kr} , that have some effect on the observed current, I_{observed} . To account for these artefacts, we can define I_{observed} as follows

$$I_{\text{observed}} = I_{Kr} + I_{\text{leak}} + I_{\text{contaminating}}, \quad (5.1)$$

where I_{leak} and $I_{\text{contaminating}}$ represent the currents resulting from linear leak and contamination artefacts, respectively. In Figure 5.3A, we plot, in blue, the observed current, I_{observed} , for a single sweep of the Milnes protocol in an example well. For each I_{observed} sweep, we can approximate the corresponding I_{leak} via *leak ramps* included at the beginning of each

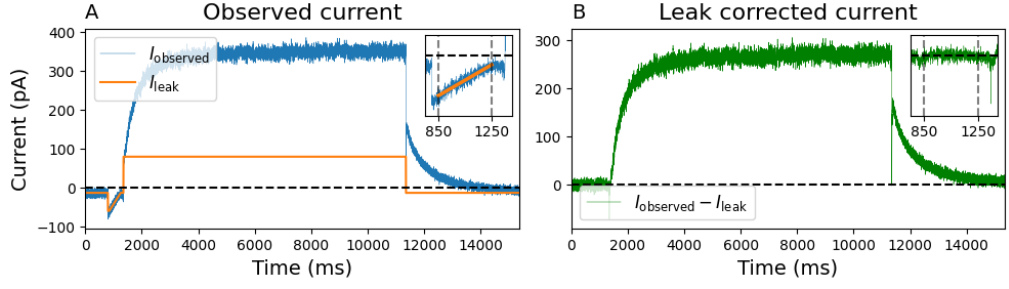


Figure 5.3: An illustration of the leak correction procedure. A: In blue, we plot the observed currents, I_{observed} , in an example well under one sweep of the Milnes protocol. By fitting a linear regression model to the currents recorded during a leak ramp, we can approximate the leak currents, I_{leak} , which we plot in orange. B: The leak-corrected current, $I_{\text{observed}} - I_{\text{leak}}$.

sweep of the applied voltage protocol. These leak ramps involve a linear increase in voltage from -120 mV to -80 mV over 400 ms. For our example sweep in Figure 5.3A, the I_{observed} corresponding to the leak ramp is highlighted in the top-right inset. It is assumed that at -80 mV and below, I_{Kr} is fully closed and, given no experimental artefacts, we would expect to observe zero current. It follows that we can approximate the magnitude of any currents resulting from leak by fitting a linear leak current model ($I_{\text{leak}} = aV + b$) to the current recorded during the leak ramp. We plot this fitted I_{leak} in orange in Figure 5.3A, indicating the linear regression fit to I_{observed} in the top-right inset. We can then subtract this estimated I_{leak} from the observed currents to effectively remove the leak artefact. In Figure 5.3B, in green, we plot the example leak corrected current. Note that the current during the leak-ramp step is now effectively zero; we have successfully accounted for the leak artefact.

To account for $I_{\text{contaminating}}$, as detailed in Section 5.2.4, we also ran full-block sweeps for both the Milnes and staircase protocols. Ideally, we would expect these full-block sweeps to exhibit $I_{\text{Kr}} = 0$ such that the observed current under full-block, $I_{\text{observed}}^{\text{full block}}$, can be described entirely as a combination

of leak and contamination artefacts

$$I_{\text{observed}}^{\text{full block}} = I_{\text{leak}} + I_{\text{contaminating}}. \quad (5.2)$$

It follows that by leak correcting the currents observed during these full-block sweeps, we can calculate an estimate for $I_{\text{contaminating}}$. We can then subtract this $I_{\text{contaminating}}$ estimate from all preceding leak-corrected sweeps (of the same protocol) in that well to minimise any remaining non- I_{Kr} currents.

With the data post-processed, we then applied several QC criteria to retain only clean and consistent data for model fitting. First, we applied the QC1 and QC2 described by Lei et al. (2019b) to the staircase protocol sweeps. QC1 applies strict upper and lower bounds on seal resistance, cell capacitance, and series resistance, effectively filtering any wells in which the cell catch or seal failed. QC2, meanwhile, ensures good signal-to-noise. For this Milnes protocol dataset, we also noticed that in some wells, including those where DMSO was applied, there was a slow, unexpected increase in current from sweep to sweep. In Figure 5.4, we plot five consecutive control currents for a well where this effect is evident (B), in comparison to a well, under identical experimental conditions, where this sweep-to-sweep current increase is not observed (A). We suspect that this sweep-to-sweep increase is an undesirable experimental artefact, possibly as a result of a slight, gradual temperature increase in the cell. To avoid cases where this was a particularly prevalent effect, we introduced a QC filter that removed wells where the normalised gradient of this increase (or decrease) across the final five control sweeps was greater than 0.15 per sweep (or less than -0.15 per sweep). To calculate this normalised gradient, we first fitted basis splines to each of the final five control sweeps and took the

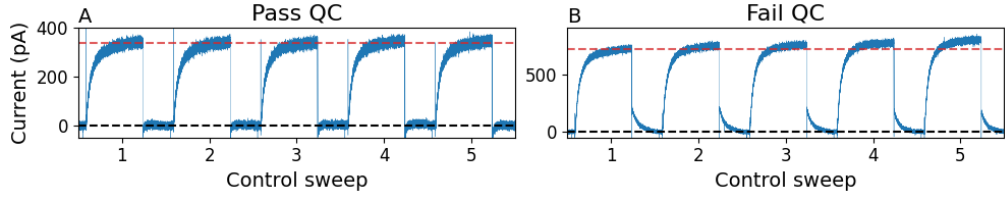


Figure 5.4: An illustration of the observed sweep-to-sweep drift artefact. A: Five consecutive control current sweeps (under the Milnes protocol) in a well that passes QC. Currents have been leak-corrected and full-block subtracted. The data were collected with 10 s at resting potential between sweeps — we do not plot the recorded currents during these 10 s intervals here, instead concatenating only the 15 s recorded during each Milnes sweep. A red horizontal dashed line is plotted to illustrate a consistent maximum current reached during each Milnes sweep. B: An equivalent five sweeps for a well under identical experimental conditions in which we observe a gradual sweep-to-sweep increase in current. This well was filtered during QC.

final value of the fitted spline as an approximation for the current at the end of each Milnes sweep. Similarly to sections 3.5 and 3.6 in Chapter 3, these basis splines were fitted using the `scikit-fda` Python package. To produce smooth spline fits of currents during each sweep, we used 4th-order polynomials defined between six evenly spaced knots along the current trace measured during the 0 mV Milnes step. We then fitted a linear regression model where the independent variable was the control sweep number (i.e. $\mathbf{x} = [1, 2, 3, 4, 5]$) and the dependent variable was the final value of the fitted spline for the corresponding control sweep. Finally, we took the fitted linear regression gradients for each well and normalised them by dividing each by an approximation of the standard deviation of the noise in that well. This noise standard deviation was approximated by evaluating the standard deviation of the observed currents in the first 0.8 s of each protocol sweep — a region where the protocol is at resting potential and the currents tend to remain stable at 0 pA.

Further QC involved manually filtering out several wells based on particularly noisy or inconsistent Milnes protocol sweeps. In Appendix C,

we plot examples of post-processed currents in three wells that pass QC (Figure C.1) and three wells that were manually filtered out during QC (Figure C.2).

The final post-processing step was to put our data in the desired form for model fitting. We first cropped each data sweep to include only currents corresponding to the 0 mV pulse of the Milnes protocol (where the channel opens, and hence, we observe measurable current). We then divided each drug sweep by a control sweep from the same well to get a measure of the proportion of channels in the open state, concatenating ten of these successive open proportion pulses. Finally, for each drug compound and concentration, we took the average open proportion across all wells that passed QC. This average open proportion quantity is then what we use to fit the various candidate drug binding models.

5.3 Model fitting with initial data

5.3.1 Mathematical models

In this chapter, we again model hERG gating mechanisms with the four-state Beattie ODE model as introduced in Chapter 2, illustrated in Figure 2.5c, and used throughout this thesis. We then, once again, consider the binding model extensions introduced in Figure 2.6.

5.3.2 Fitting the hERG model

In the previous chapter, we detailed a method for fitting binding models to the open proportion quantity using a normal ratio log-likelihood. This

method assumed we had a known, well-calibrated, underlying hERG physiological model. This was a straightforward assumption in the context of synthetic data; however, with real data, we must be careful to ensure that our hERG model is satisfactory. In theory, we could fit our Beattie hERG model parameters to the processed open proportion data while simultaneously fitting the drug-binding model parameters. However, when we have fifteen binding models, this becomes quite computationally expensive. As an alternative, we can use the leak-corrected and full-block subtracted staircase protocol sweeps to fit the hERG model parameters once, fixing these, before separately fitting our drug-binding parameters. Given our focus on model selection for drug binding dynamics, this approach additionally ensures that the fitted binding parameters all correspond to identical hERG gating behaviour, providing an effective comparison between binding models.

While we do observe a certain level of well-to-well variability in hERG current, characterising this variability was previously performed by Shuttleworth et al. (2025) and is beyond the scope of this thesis. In fact, there is reason to believe that a significant amount of any observed variation in hERG dynamics results from experimental artefacts as opposed to any significant cell-to-cell kinetic differences (Lei et al., 2020). We assume, then, that fitting to a single staircase current trace in a representative ‘median’ well should give a fairly good approximation of hERG dynamics for a given plate. We determine this ‘median’ well by first taking the observed post-processed currents, $I_{Kr,w}$, recorded during the first staircase sweep in each well, w , that passes QC. To reduce the variability introduced by different channel conductances (g_{Kr}), we then divide $I_{Kr,w}$ by the observed current during the -40 mV step beginning at 10.9 s in the staircase protocol (where we generally observe the largest current) to get normalised staircase sweeps,

$\hat{I}_{\text{Kr},w}$. Next, we calculate the point-wise average staircase trace for all $\hat{I}_{\text{Kr},w}$, and determine the well that is closest in Euclidean distance to this average as described by the objective

$$\min_w \left\| \hat{I}_{\text{Kr},w} - \frac{1}{N} \sum_n \hat{I}_{\text{Kr},n} \right\|_2, \quad (5.3)$$

where N is the total number of wells that pass QC for a given plate. For the well, w , that minimises Equation (5.3), we consider the staircase current, $I_{\text{Kr},w}$, to be the best representation of hERG dynamics across the plate. We then fit the hERG parameters to $I_{\text{Kr},w}$ by running ten repeats of the CMA-ES (Hansen, 2006) optimisation algorithm with an L_2 -norm objective function. For each repeat, $i \in [1, 10]$, we initialise the model parameters by randomly perturbing the hERG parameters used by Lei et al. (2024), which we will denote $\boldsymbol{\theta}_0$. This perturbed initialisation, $\boldsymbol{\theta}_{0i}$, is determined by the following equation

$$\boldsymbol{\theta}_{0i} = \boldsymbol{\theta}_0 \times 2^{\phi_i}, \quad (5.4)$$

where ϕ_i is drawn from a normal distribution $\mathcal{N}(0, 1)$, resulting in subtle differences in each initialisation, while retaining the biophysically realistic behaviour characterised by $\boldsymbol{\theta}_0$. We then set bounds on the hERG parameters to ensure that they remain within a biophysically realistic range throughout the optimisation. Specifically, given the $a_j \exp(b_j V)$ form (introduced in Equation (2.12) in Section 2.2.2) of the transition rates k_j for $j \in [1, 4]$, we define initialisation bounds for a_j of $[1 \times 10^{-7}, 1 \times 10^3]$, and for b_j , $[1 \times 10^{-7}, 0.4]$. Using the default PINTS (Clerx et al., 2019b) implementation of CMA-ES, we then run each fitting repeat, i , with a maximum of 200 unchanged iterations (with a threshold in objective change of 1×10^{-3}). In Figure 5.5, we plot the best hERG model fit against the

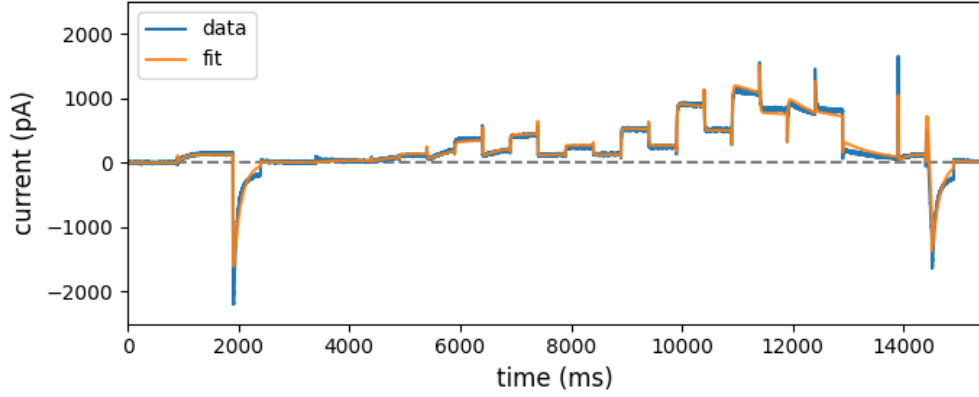


Figure 5.5: The Beattie hERG model fit compared to staircase data. In blue, we plot the post-processed current under the staircase protocol in a ‘median’ representative well on the SyncroPatch plate containing fast-binding drugs (Milnes protocol data). Parameters of the Beattie hERG model were fitted to this data, and in orange we plot this fitted model.

representative staircase current data for the SyncroPatch plate containing fast-binding drugs (see Figure C.3 in Appendix C for an equivalent plot for data from the plate containing slow-binding drugs). While the fit is not perfect, the model and data do match relatively well. In Section 4.3.5 in the previous chapter, we considered a synthetic data example in which we introduced a considerable discrepancy between the data-generating Beattie hERG model and a dynamically different Lu model (as illustrated in Figure 4.8). The hERG model fit in Figure 5.5 appears to express dynamics that are significantly more alike than this previously considered Beattie vs. Lu discrepancy, giving us confidence that we are not in a scenario where the hERG model discrepancy pollutes any attempt to gain useful insights into drug binding dynamics. For this reason, we consider these hERG model fits sufficient approximations of the hERG binding dynamics across each plate, and we will use these parameterisations moving forward.

5.3.3 Fitting the drug-binding models

With a hERG model parameterised, we next fit the parameters of the fifteen proposed binding models to the post-processed open proportion traces via the methods introduced in Section 4.3.2 in Chapter 4. In contrast to Chapter 4, the open proportion data considered here are the average of several wells. To be precise, in each well, w , we have a drug current trace, $x_{w,c}(t)$ (for each drug concentration c) and a control current trace, $y_w(t)$. The data we use for model fitting can then be described by

$$\bar{z}_c(t) = \frac{1}{N} \sum_w \frac{x_{w,c}(t)}{y_w(t)}, \quad (5.5)$$

where N is the number of wells that pass QC (for a given drug and concentration combination). Clearly, we are no longer working directly with the ratio of two normal variables, and the normal-ratio error model is no longer necessarily valid. However, for simplicity, we assume that this average quantity, $\bar{z}_c(t)$, can be considered roughly equivalent to the ratio of two random i.i.d. normal variables with equal standard deviation. With this assumption in place, we can then use the log-likelihood described in Equation (4.3) to fit our drug-binding models to $\bar{z}_c(t)$. Calculating this likelihood requires the quantity $\delta(t, \sigma) = \sigma / \mu_y(t)$, where μ_y is the mean control current. To approximate μ_y , we fit a spline to the mean recorded control current data across all wells that passed QC for a given drug (i.e. $\bar{y}(t) = \frac{1}{N} \sum_{w=1}^N y_w(t)$). This spline-fitting is done using the same methods described in Section 5.2.5, increasing the number of evenly spaced knots from six to 100 to ensure the shape of the full (mean) control sweep is accurately approximated. Using this approximated μ_y , we can then fit each of the fifteen binding models to $\bar{z}_c(t)$, $\forall c \in C$. We run ten repeats of CMA-ES optimisation with each repeat starting from a random parame-

ter initialisation drawn from the boundaries described in Lei et al. (2024). The $\hat{\theta}_m$ corresponding to the largest of the ten log-likelihoods is designated the MLE for the parameters of model m , with the fitted model trace described by $\beta_m(t, c, \hat{\theta}_m)$. All binding model fitting results are described in Section 5.7.

5.4 Optimising protocols

With the drug-binding models fitted to the Milnes protocol data, the next step involves developing an improved experimental design to assist with model selection. For some experimental design, $\mathcal{D}$, parameterised by ϕ , we wish to adjust $\mathcal{D}$ via ϕ to maximise some optimality criterion. As described in Chapter 4, we use an optimality criterion that seeks to maximise the median pairwise sum-of-squares differences between each of the fitted binding model currents. This ‘max-med’ objective is defined in Equation (4.5). Considering $\mathcal{D}(\phi)$, we must then decide on a design space over which we want to optimise our objective function. Taking experimental and computational practicalities into account, we use the same design space as described in Chapter 4 with a few extensions. In Figure 5.6, we illustrate this extended design space, indicating the regions of the voltage protocol that can vary during optimisation. Firstly, the 10 s pulse at 0 mV in the Milnes protocol is split into three sections, each of which is allowed to vary in duration (between 20 ms and 5 s) and voltage (between -120 mV and 50 mV). In Chapter 4, these voltage steps were fixed in length and could only vary between -50 mV and 40 mV. We use an updated voltage range here, that remains within a biologically realistic membrane potential range, and additionally allows for channel closing (at voltages of -80 mV and below). Meanwhile, allowing the voltage steps to vary in length is an

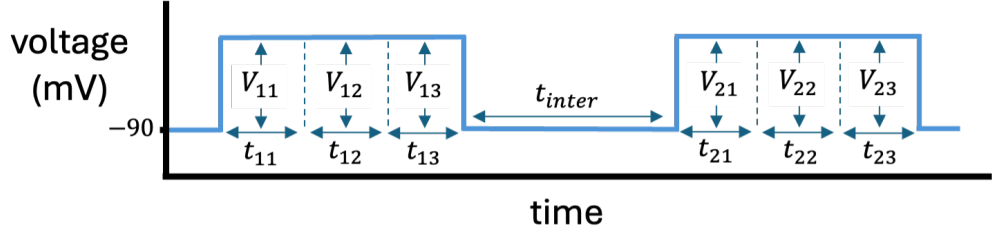


Figure 5.6: Schematic of the proposed design space for protocol optimisation. Voltages, V_{ij} , represent the membrane potential at each optimised step, while times, t_{ij} , represent the corresponding time at each step. V_{ij} can vary between -120 mV and 50 mV, and t_{ij} can vary between 20 ms and 5 s. The interpulse time, t_{inter} , can vary between 1050 ms and 21 s.

obvious extension to increase the design space. This design space is further increased by including two of the three-section pulses per sweep, both of which can vary independently of each other. The time at holding potential between the two pulses is then also allowed to vary between 1050 ms and 21 s. As noted in Section 5.2.5, we include a leak ramp at the start of each protocol sweep. So as not to disturb the leak-correction procedure, and to ensure that the hERG channel has sufficient time to close, we, therefore, fix the time at holding potential at the end of each sweep to 15 s. While -80 mV is generally considered a satisfactory holding potential (Windley et al., 2018), data collected in trial experiments suggested instances where the channel had not fully closed during these -80 mV sections. We, therefore, also lowered the holding potential from -80 mV to -90 mV to ensure the channel was fully closed at the end of each sweep. The described design space has thirteen degrees of freedom, and we can define $\phi = [V_{11}, V_{12}, V_{13}, V_{21}, V_{22}, V_{23}, t_{11}, t_{12}, t_{13}, t_{21}, t_{22}, t_{23}, t_{inter}]$. Here V_{ij} is the voltage (in mV) in the j th section of the i th pulse, t_{ij} is the duration (in ms) in the j th section of the i th pulse, and t_{inter} is the time between pulses at holding potential.

With a ϕ defined, we can then optimise our max-med criterion subject to the voltage and duration constraints. We run 20 optimisation repeats, and

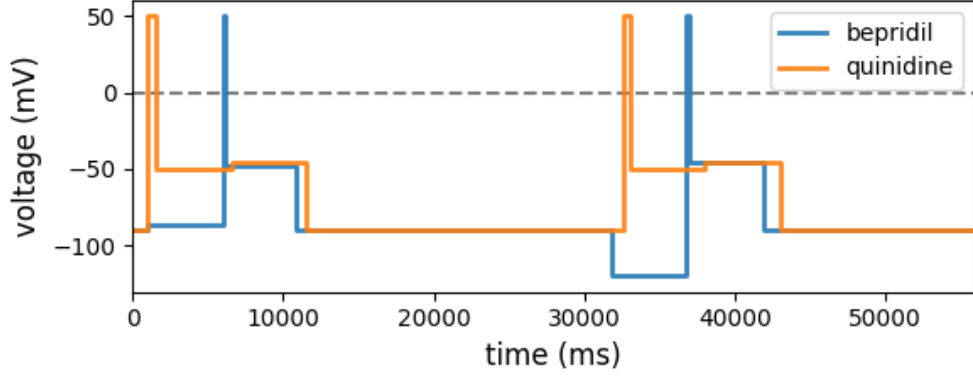


Figure 5.7: Optimised protocols for a slow-binding drug, bepridil, and a fast-binding drug, quinidine. The optimisation methods are as detailed in Section 5.4.

for each of these repeats, we begin by sampling 2000 instances of ϕ from the defined bounds. For each repeat, we take the ϕ corresponding to the largest max-med objective of the 2000 samples and use this as the initialisation for the CMA-ES optimisation algorithm. In Figure 5.7, we plot the optimised protocols for bepridil and quinidine based on the respective $\hat{\theta}_m$ ($\forall m \in M$) fits obtained when fitting the binding models to the Milnes protocol block data, $\bar{z}_c(t)$, for each drug. There are notable differences between these two protocols; however, it is difficult to draw any immediate conclusions. The protocols generated for the other three drugs are included in Figure C.4 in Appendix C, and, in comparison to the differences illustrated in Figure 5.7, all appear remarkably similar.

5.5 Optimised protocol data collection

With optimised protocols generated for each of the five considered drugs, we could then repeat the experimental procedure as described in Section 5.2, replacing the Milnes protocol with the drug-specific optimised protocol. Given that the optimised protocol is roughly equivalent in length to two

sweeps of the Milnes protocol, we ran a minimum of only three sweeps of the optimised protocol before the drug was added, and a minimum of six sweeps after the drug compound was added. We also made a few notable changes to our post-processing and QC procedure.

5.5.1 Post-processing and quality control

Leak correction and full-block subtraction were done as per Section 5.2.5. However, in contrast to the initial dataset, we used all six QC criteria specified in Lei et al. (2019b); the data collected under the optimal protocols were far cleaner and more consistent, and we were able to apply the additional criteria while retaining a similar number of passed wells. The gradual sweep-to-sweep current increase observed in the initial dataset was not as prevalent in the optimal protocol dataset, so the corresponding QC criterion was removed for these new datasets. We then, once again, manually filtered a few remaining wells with excessive noise or sweep-to-sweep inconsistencies. Next, we cropped each data sweep to remove sections recorded at the resting potential (-90 mV), leaving only the current during the six optimised voltage steps (i.e. steps corresponding to t_{11} , t_{12} , t_{13} , t_{21} , t_{22} , and t_{23} in Figure 5.6). Using a control sweep in each well to calculate the open proportion for each drug sweep, we finally concatenated six successive sweeps of the optimised protocol for each drug compound and concentration, and took the average open proportion across the wells that passed QC as previously.

5.6 Model fitting with optimised protocol data

The model fitting procedure for the optimal protocol data is identical to the Milnes protocol data. Regarding the hERG model, we can, once again, fit the Beattie hERG model parameters to the post-processed staircase data first, fixing these parameters before fitting the binding model parameters separately. To gauge how well each binding model explains both the Milnes and optimised protocol datasets, we also fit each of the binding models to both datasets simultaneously. This ‘dual’ fitting is achieved by defining a likelihood function consisting of a sum of the two (normal ratio) log-likelihoods previously used as objectives when fitting each dataset separately. Because these log-likelihoods are each defined for differing hERG model parameterisations, the two respective hERG model fits for each dataset are used simultaneously when performing these dual fits. From a modelling perspective, this assumes that the hERG gating dynamics may vary slightly between plates, while the drug-binding dynamics remain consistent. While experimental artefacts (among other factors) result in inevitable variability in observed drug-binding dynamics between wells/plates, these dual fits are performed to determine how well a particular binding model can describe two datasets generated under effectively identical experimental conditions (besides the different voltage protocols applied).

5.7 Results

We now describe the results of the model fitting and what these fits tell us about the feasibility of each drug-binding model in capturing the dynamics of the considered compounds. The results for one slow-binding drug, bepridil, and one fast-binding drug, quinidine, are discussed in detail. Results for the other three drugs assessed (verapamil, diltiazem, and chlorpromazine) are summarised, with further details included in Appendix C.

5.7.1 Bepridil

As detailed in Section 5.3.3, before generating optimised protocols, we first fitted the fifteen proposed binding models to post-processed bepridil block data collected under the Milnes protocol. In the top of Figure 5.8, we plot the processed Milnes open proportion data beneath the fits for all fifteen binding models. Similarly to the results in Lei et al. (2024), all fitted model traces appear fairly alike, and approximate the dynamics in the data reasonably well. In the bottom left of Figure 5.8, we include a plot of the log-likelihoods corresponding to each of the model fits. Models 12 and 13 have the highest log-likelihoods (1.46×10^6), while model 2i has the lowest (1.00×10^6). This spread in log-likelihoods is not insignificant, and, as discussed in sections 2.5.3 and 4.3.2, a classical model selection method such as a likelihood-ratio test would definitively rule out some of these models based on this Milnes data alone. However, given the potential for experimental artefacts and model discrepancies still unaccounted for, we remain suspicious of the veracity of our log-likelihoods, and seek methods that further separate the quality of the model fits.

Based on these initial model fits, an optimised protocol was generated (see

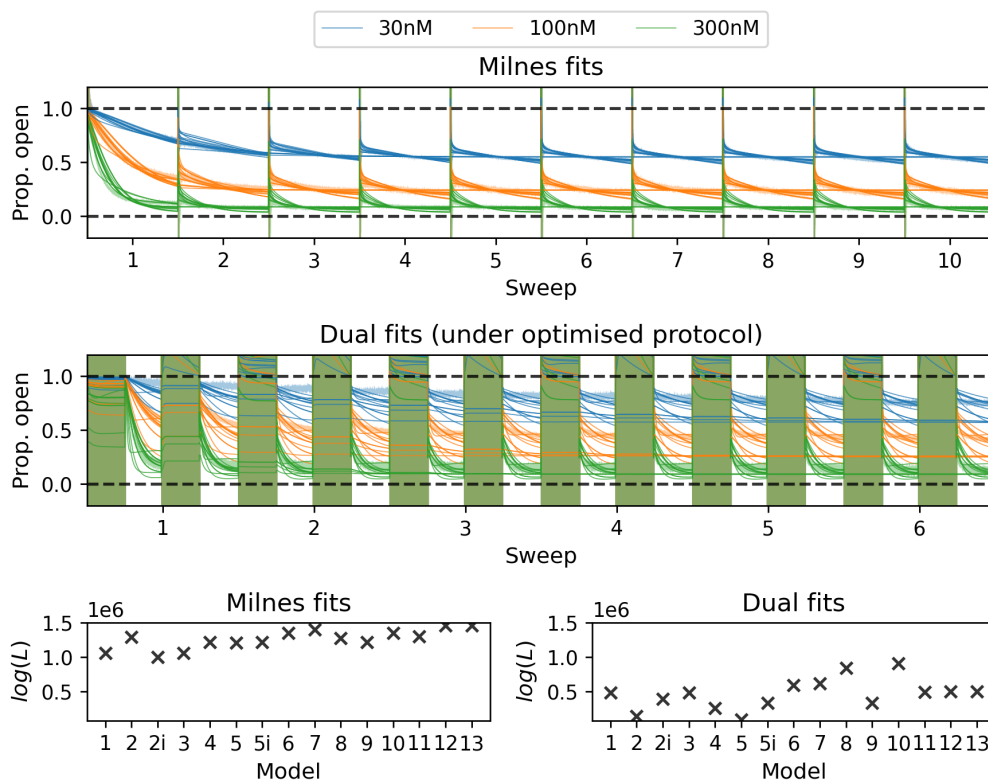


Figure 5.8: Fitting drug binding models to bepridil block data collected under the Milnes protocol and an optimised protocol. Top: Milnes data vs. model fits. Middle: Optimised protocol data vs. dual fits. The alternating green ‘bars’ correspond to regions of the generated protocol where currents are near zero, and hence the open proportion quantity is particularly noisy. The normal-ratio log-likelihood accounts for the increased noise in these sections, prioritising fitting the data in regions with higher signal-to-noise. Bottom left: Log-likelihoods for each binding model fit to Milnes data. Bottom right: Log-likelihoods for each binding model dual fit.

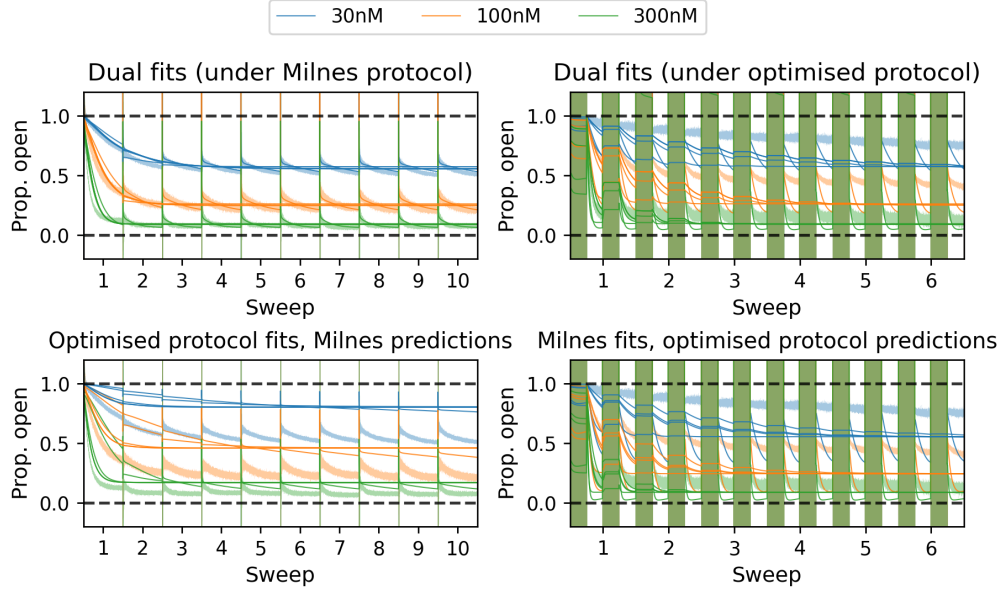


Figure 5.9: Binding model fits for the five lowest log-likelihood dual model fits when fitting to bepridil block data. The binding models plotted are 2, 4, 5, 5i, and 9. Top left: Milnes data vs. dual fits. Top right: Optimised protocol data vs. dual fits. Bottom left: Milnes data vs. predictions based on optimised protocol data fits. Bottom right: Optimised protocol data vs. predictions based on Milnes data fits.

Figure 5.7), and further bepridil block data were collected under this new protocol. In the middle row of Figure 5.8, we plot the model fits under the optimal protocol when fitting to both the Milnes data and the optimised protocol data simultaneously. We will, from now on, refer to these fits as the ‘dual’ fits. We now see a much greater variation in the quality of the model fits, with traces appearing to represent a wide array of dynamics. This is reinforced by the log-likelihood plot in the bottom right of Figure 5.8. Model 10 is now the best fit with a log-likelihood of 9.09×10^5 , compared to the worst fit for model 5 of 0.92×10^5 . To emphasise fits that no longer appear feasible, in the top half of Figure 5.9 we plot the five worst dual fits (models 2, 4, 5, 5i, and 9). The top left and right subplots contain the dual fits under the Milnes and optimised protocols, respectively. All five model fits fail to visually match the data, particularly in the optimised protocol case. While this is not definitive evidence, it strongly suggests that these

five models are unsuitable models of bepridil binding dynamics. We also consider how well a model fitted to one of the data sets can predict the other dataset. We plot these predictions for models 2, 4, 5, 5i, and 9 in the bottom half of Figure 5.9 and note again that the predictions for these models are consistently poor, failing to capture the dynamics observed in the data. It is interesting to note that of these five worst dual fits, the structures of models 4, 5, 5i, and 9 all assume identical hERG gating dynamics when the drug is trapped, compared to when no drug is present in the channel. This strongly suggests that, should bepridil exhibit trapping behaviour (as suggested by Stork et al. (2007)), then the presence of trapped bepridil alters the underlying hERG gating behaviour. The other conclusion to draw from these poor model fits is that bepridil seems to rely on some degree of inactive binding — the two models that exclusively allow open-state binding, models 2 and 5, both appear infeasible.

While the optimised protocols have effectively ruled out several models based on quality of fit, other insights can be gained by specifically analysing the fitted model parameters. We consider models 1 and 3. For these two models, we observe that for each fit to data (Milnes, optimised protocol, and dual), we return almost identical fits, including in the MLE parameters returned. While model 3 allows for gating between the ID and OD states and model 1 does not, by analysing the fitted binding parameters in comparison to the ID-OD hERG model transition rates, it becomes clear that the ID-OD gate in model 3 is redundant. For model 3 (and model 1), the dual fitted binding rate parameters are $k_{\text{on}} = 9.44 \times 10^{-7} \text{ ms}^{-1} \text{ nM}^{-n}$ and $k_{\text{off}} = 1.88 \times 10^{-5} \text{ ms}^{-1}$. In comparison, transition rates between the ID and OD states, defined by the fitted hERG model parameters, are determined by the following voltage-dependent functions $k_3 = 0.17 \times 10^{0.011V} \text{ ms}^{-1}$ and $k_4 = 0.015 \times 10^{-0.026V} \text{ ms}^{-1}$. Clearly, for any voltage in the range that we

are considering (-120 mV to 50 mV), the ID-OD transitions will be occurring at a much faster rate than any binding that is occurring. It follows that the ID-OD gate in model 3 is redundant, and model 3 is effectively reducing to model 1.

We next consider potential redundancies introduced by directly nested models. We refer the reader to Figure 4.11 in Chapter 4 for a visualisation of the nesting interactions between the set of fifteen candidate models. Binding model 12 nests model 13, and we notice that the dual fits for both of these models appear identical. As described in Figure 2.6, the on-binding rate for model 12 is defined by the following equation

$$E_{max}([D]) = \hat{k}_{on} \cdot \frac{[D]^n}{[D]^n + EC_{50}^n}, \quad (5.6)$$

where $[D]$ is the drug concentration, n is the Hill coefficient, EC_{50} is the concentration at which the drug binding rate is half of its maximum rate, and $\hat{k}_{on}$ is the binding rate with units ms^{-1} . In contrast, the equivalent on-binding rate for model 13 is given by $k_{on}[D]^n$, where k_{on} here has units in $ms^{-1} \cdot Nm^{-n}$. We find that the dual fitted EC_{50} parameter for model 12 is 9.99×10^8 , which dominates the $[D]^n$ term in the denominator of Equation (5.6), effectively reducing model 12 to model 13. To verify this, we confirm that the dual fitted parameters ensure that $\hat{k}_{on}/([D]^n + EC_{50}^n)$ for model 12 is equal to k_{on} for model 13.

Given the set of models that produce particularly poor fits, and the additional model redundancies discussed, this leaves eight remaining feasible binding models. In descending order of (dual-fit) log-likelihood, these models are 10, 8, 7, 6, 13, 11, 1, and 2i. Previous research by Stork et al. (2007) suggests that bepridil is a drug exhibiting trapping behaviour, so it is interesting to note that of these eight remaining models, only four represent

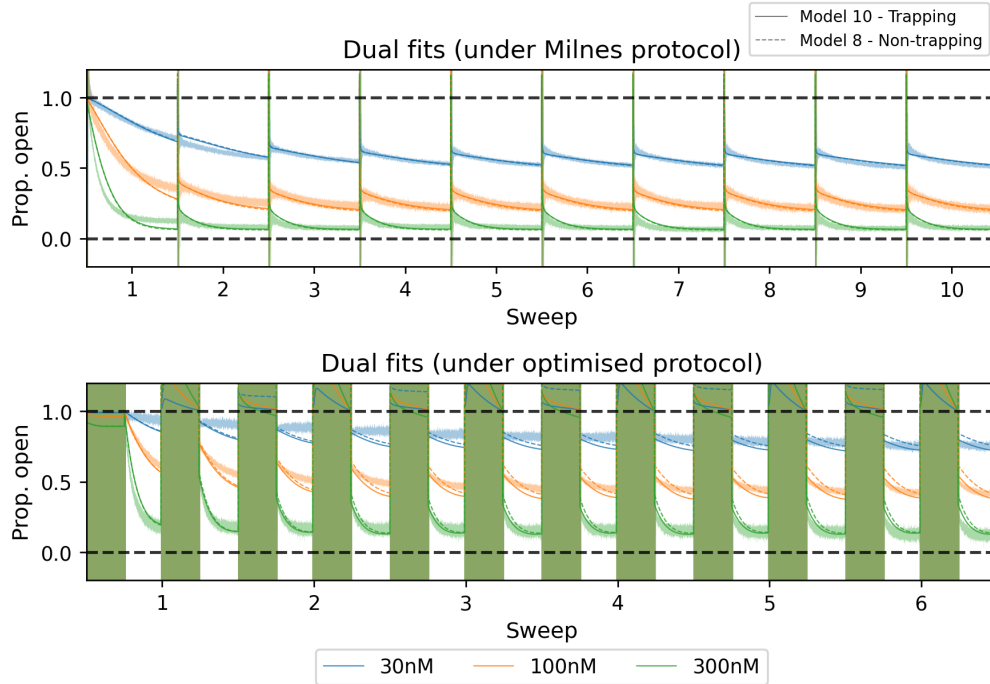


Figure 5.10: Binding model dual fits for the trapping and non-trapping models with the highest log-likelihoods when fitting to bepridil block data. The binding models plotted are model 10 (trapping) and model 8 (non-trapping). Top: Milnes data vs. dual fits. Bottom: Optimised protocol data vs. dual fits.

some form of trapping behaviour: 10, 6, 13, and 11. In Figure 5.10, we plot the dual model fits for the best trapping (model 10) and non-trapping (model 8) binding models in terms of dual fit log-likelihood. While it appears that the trapping model, model 10, fits the data better, as supported by the respective log-likelihoods (9.09×10^5 vs. 8.41×10^5), an additional round of protocol optimisation and data collection would be advisable to verify this.

Ultimately, the data collected under the optimised protocol has provided us with significant information to aid binding model selection for bepridil. Seven of the fifteen models that were previously deemed feasible when fitting to just the Milnes protocol data, are now either unable to simultaneously capture the dynamics in the Milnes and optimised protocol datasets, or can be ruled out based on redundancies and model nesting. Meanwhile,

the remaining eight models can reasonably fit the two datasets, although, as we have discussed, there are suggestions in the literature (Stork et al., 2007) that some of these remaining models could also be ruled out given further informative data.

5.7.2 Quinidine

We now turn our attention to a faster-binding drug, quinidine. As with bepridil, when we fit the candidate binding models to the Milnes protocol data, we gain few insights. In the top of Figure 5.11, we plot each of the model fits on top of the Milnes open proportion data. While some model fits (4, 5, 5i, and 9) fail to capture the apparent non-trapping behaviour observed in the data, it seems further data is required to definitively rule these in or out. In the bottom left of Figure 5.11, we also plot the log-likelihoods for each Milnes fit. The highest log-likelihood is for model 7 (1.46×10^6), while the lowest is for model 2 (1.15×10^6).

With the Milnes protocol data unable to provide any useful insights to assist in model selection, we once again consider the additional information provided by the optimised protocol data. In the middle of Figure 5.11, we plot the optimised protocol data and the dual model fits under this optimised protocol. These dual fits again result in a divergence in quality of fits, emphasised by the log-likelihood plot in the bottom right of Figure 5.11. While the best and worst fitting models remain the same, with model 7 being the best fit (2.40×10^6) and model 2 the worst (0.86×10^6), there is now a much greater spread in the range of log-likelihoods. In Figure 5.12 we illustrate the seven models with the lowest dual-fit log-likelihoods: models 2, 4, 5, 5i, 9, 12, and 13. We note that the dual fits for these models now all appear infeasible. The top left subplot shows the dual fits under the Milnes

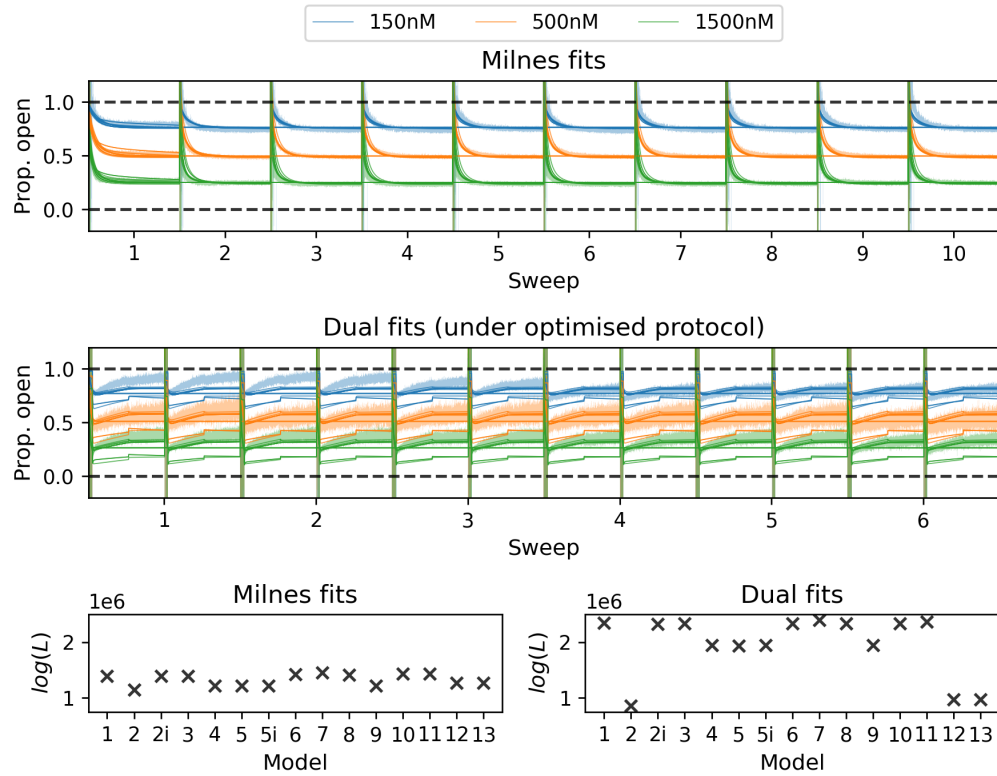


Figure 5.11: Fitting drug binding models to quinidine block data collected under the Milnes protocol and an optimised protocol. Top: Milnes data vs. model fits. Middle: Optimised protocol data vs. dual fits. Bottom left: Log-likelihoods for each binding model fit to Milnes data. Bottom right: Log-likelihoods for each binding model dual fit.

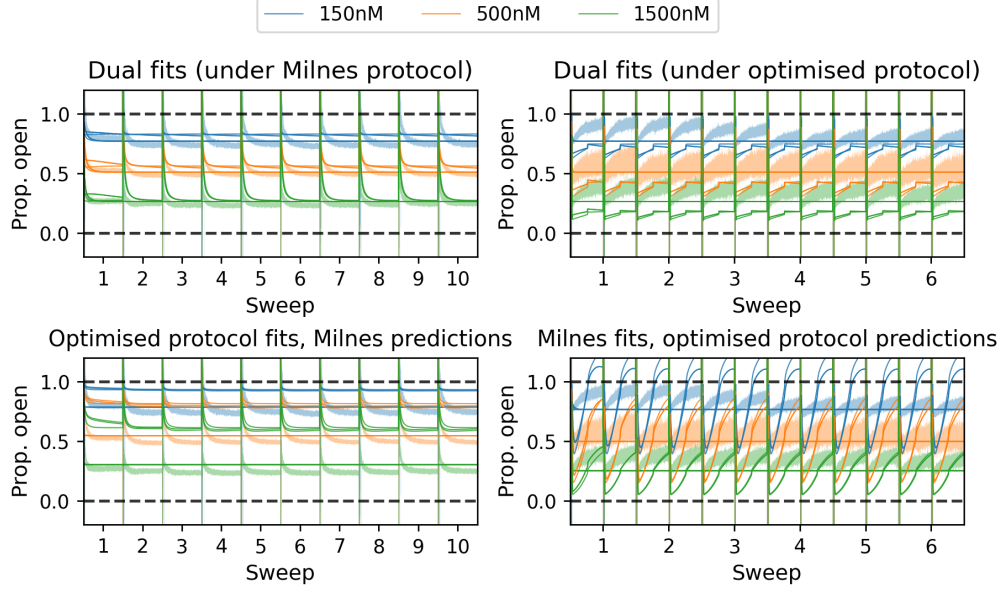


Figure 5.12: Binding model fits for the seven lowest log-likelihood dual model fits when fitting to quinidine block data. The binding models plotted are 2, 4, 5, 5i, 9, 12, and 13. Top left: Milnes data vs. dual fits. Top right: Optimised protocol data vs. dual fits. Bottom left: Milnes data vs. predictions based on optimised protocol fits. Bottom right: Optimised protocol data vs. predictions based on Milnes data fits.

protocol in comparison to the Milnes data, while the top right shows the dual fits under the optimal protocol in comparison to the optimal protocol data. In the bottom subplots of Figure 5.12, we also plot the predictions (for models 2, 4, 5, 5i, 9, 12, and 13) based on single dataset fits (similarly to Figure 5.9), and these also consistently fail to match the data. Based on these seven worst model fits, we can draw similar conclusions to bepridil; it is likely that quinidine binds in the inactive state, and that any trapping behaviour alters the underlying hERG gating mechanisms. Additionally, it is interesting that the two binding models based on the CiPAv1.0 model from Li et al. (2017), models 12 and 13, now appear infeasible.

We next consider whether any further nesting or redundancies can assist in reducing the number of feasible models. Considering models 6 and 10, we notice that the fitted trapping parameter, k_{trap} , is in the order of $\mathcal{O}(10^{-6})$ for both of these models, effectively removing the trapping component,

and reducing them to their equivalent non-trapping nested models 3 and 8, respectively.

We also note that, as with bepridil, despite the faster binding of quinidine, the ID-OD gate in model 3 again becomes redundant, reducing model 3 to model 1. This time, the dual fitted binding parameters are $k_{\text{on}} = 7.83 \times 10^{-6} \text{ ms}^{-1} \text{ nM}^{-n}$ and $k_{\text{off}} = 2.92 \times 10^{-3} \text{ ms}^{-1}$ and the voltage-dependent ID-OD transition rates are $k_3 = 0.30 \times 10^{0.018V} \text{ ms}^{-1}$ and $k_4 = 0.016 \times 10^{-0.018V} \text{ ms}^{-1}$. Clearly, the ID-OD transitions are again sufficiently faster than the binding transitions, resulting in this redundant model mechanism.

Based on the discussed redundancies and poor model fits, we are left with five remaining binding models, which, in order of decreasing log-likelihood, are: 7, 11, 1, 8, and 2i. Of these, only model 11 is capable of capturing trapping behaviour. Results from Sánchez-Chapula et al. (2003) suggest that quinidine is a non-trapping drug, supporting our findings. In Figure 5.13, we plot the dual fits for model 11 and the best non-trapping model, model 7 (in terms of dual fit log-likelihood). Both fits appear visually alike, perhaps suggesting, via an assumption of parsimony, that quinidine is a non-trapping drug.

5.7.3 Other compounds

We now summarise the results for the remaining three drug compounds considered. We refer the reader to Appendix C for further plots of fitting results for these drugs.

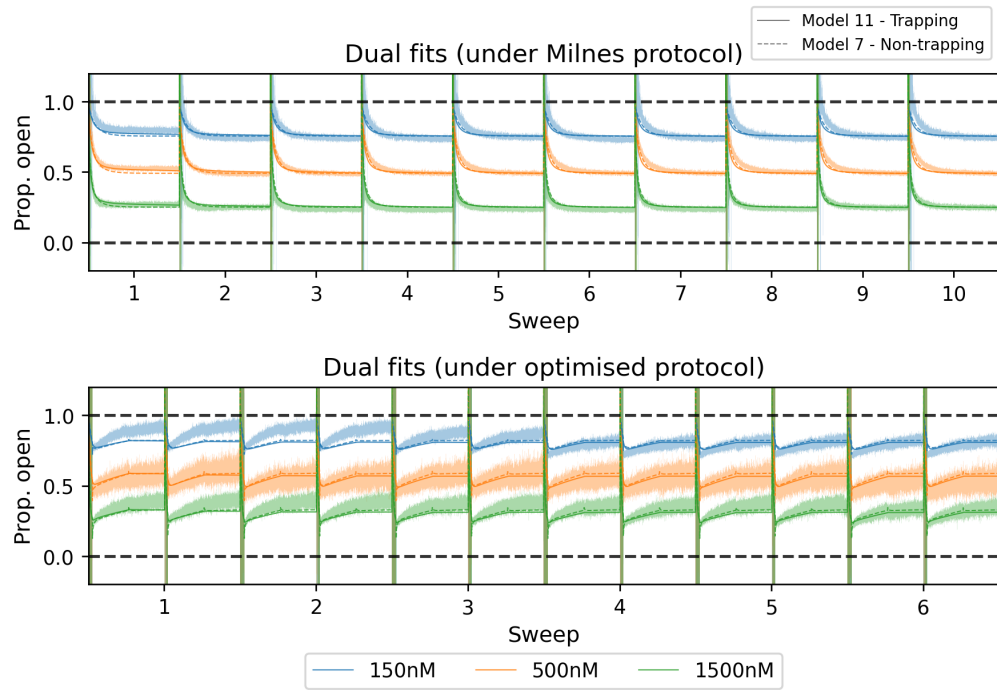


Figure 5.13: Binding model dual fits for the trapping and non-trapping models with the highest log-likelihoods when fitting to quinidine block data. The binding models plotted are model 11 (trapping) and model 7 (non-trapping). Top: Milnes data vs. dual fits. Bottom: Optimised protocol data vs. dual fits.

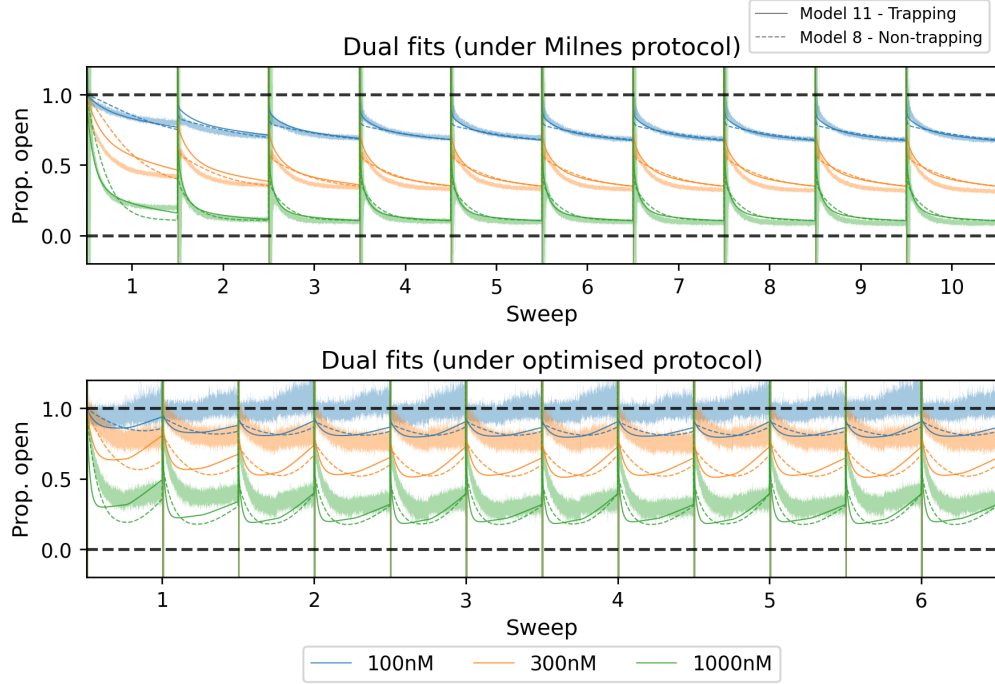


Figure 5.14: Binding model dual fits for the trapping and non-trapping models with the highest log-likelihoods when fitting to verapamil block data. The binding models plotted are model 11 (trapping) and model 8 (non-trapping). Top: Milnes data vs. dual fits. Bottom: Optimised protocol data vs. dual fits.

Verapamil

Verapamil is a slow-binding drug with previously observed non-trapping dynamics (Windley et al., 2017). Similarly to bepridil and quinidine, dual fits result in several models appearing infeasible as they are unable to simultaneously capture the dynamics expressed in both the Milnes and optimised protocol datasets. As with quinidine, we can then additionally rule out models 6 and 10 as their respective fitted trapping parameters (k_{trap}) are in the order of $\mathcal{O}(10^{-7})$, effectively reducing these models to their non-trapping counterparts (3 and 8). Once again, we can then also remove model 3 due to the redundant ID-OD gate that effectively reduces model 3 to model 1. This leaves five feasible models, which we list in order of decreasing log-likelihood: 8, 11, 7, 2i, and 1. In Figure 5.14, we plot the

best trapping (11) and non-trapping (8) model fits. Neither fit matches the data particularly well, especially in the optimised protocol case. Another round of experiments would be useful to separate the remaining models and determine whether any of these models are suitable, or if further model development is required.

Diltiazem

Next, we consider diltiazem, a fast-binding drug. Again, we can rule out several models whose dual fits appear infeasible. Given a fitted k_{trap} in the order of $\mathcal{O}(10^{-5})$, we can also neglect models 6 and 10 as they reduce to models 3 and 8 respectively. Unlike any of the previously considered compounds, we can also rule out model 7 by considering its nesting relationship with model 2i. Model 7 accommodates open and inactive binding, while model 2i represents only inactive binding. We find that the dual fitted inactive binding rates in model 7 are almost identical to those of model 2i, while the fitted model 7 open binding rate parameters are $k_{\text{on}} = 1 \times 10^{-9} \text{ ms}^{-1} \text{ nM}^{-n}$ and $k_{\text{off}} = 1 \text{ ms}^{-1}$, effectively resulting in no open-binding action. We are left with the remaining feasible models in order of decreasing log-likelihood: 2i, 11, 8, 3, and 1. In Figure 5.15, we plot the best trapping (11) and non-trapping (2i) model fits. Model 2i, in particular, appears to be a good fit for both data sets, perhaps suggesting that diltiazem is a non-trapping drug that binds to hERG only in the inactive state. Again, a further iteration of protocol optimisation and data collection would be useful to draw more definitive conclusions.

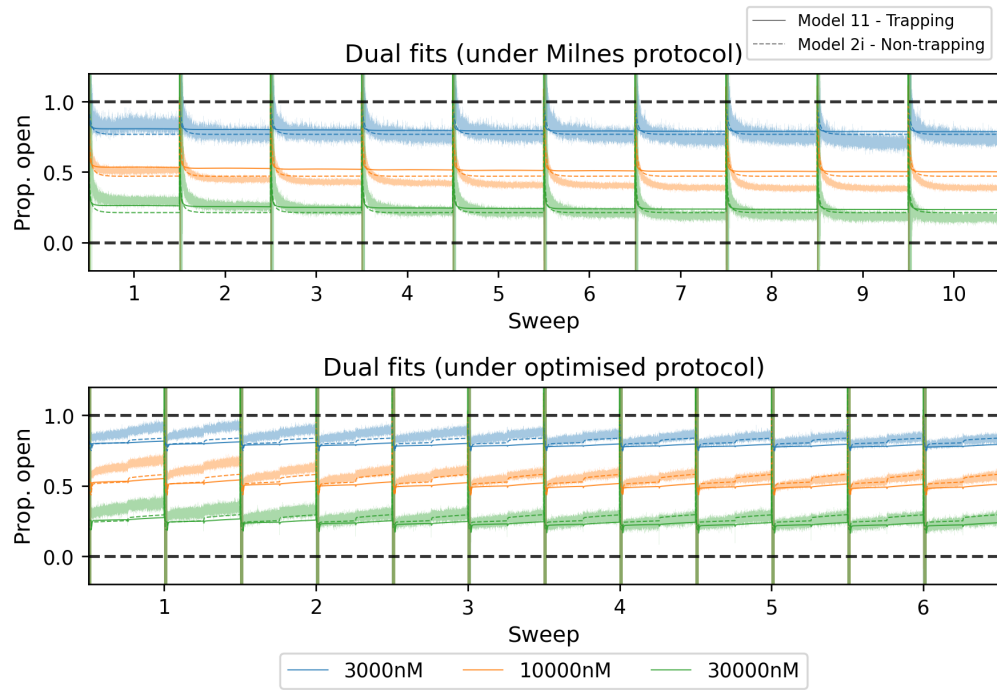


Figure 5.15: Binding model dual fits for the trapping and non-trapping models with the highest log-likelihoods when fitting to diltiazem block data. The binding models plotted are model 11 (trapping) and model 2i (non-trapping). Top: Milnes data vs. dual fits. Bottom: Optimised protocol data vs. dual fits.

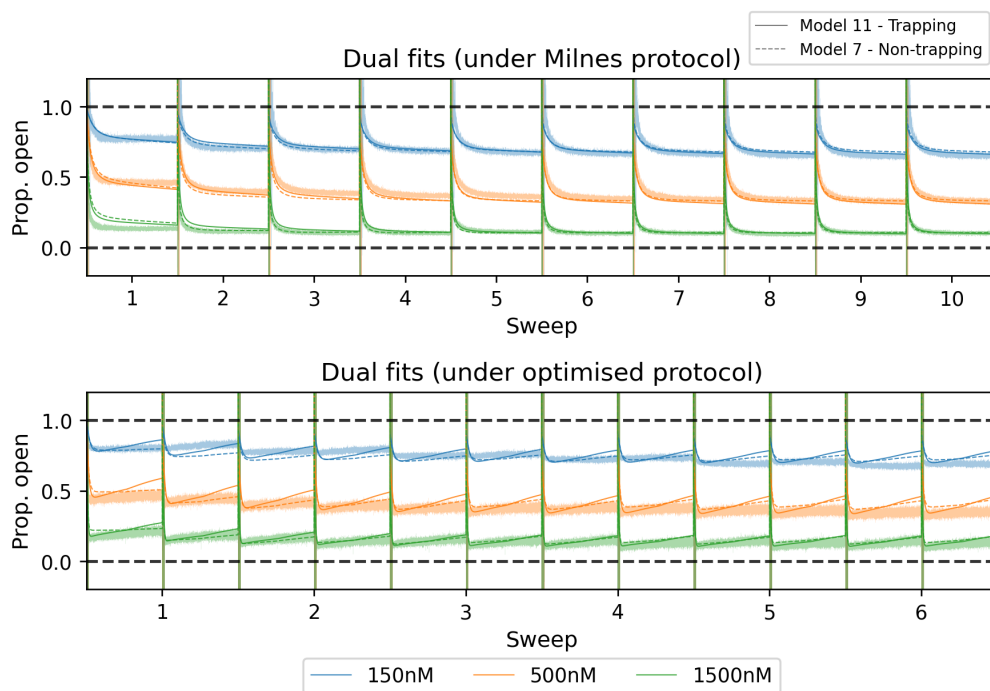


Figure 5.16: Binding model dual fits for the trapping and non-trapping models with the highest log-likelihoods when fitting to chlorpromazine block data. The binding models plotted are model 11 (trapping) and model 7 (non-trapping). Top: Milnes data vs. dual fits. Bottom: Optimised protocol data vs. dual fits.

Chlorpromazine

Finally, we turn to the results for chlorpromazine, another fast-binding drug. We find that many of the dual fits are unable to capture the data, leaving only four models remaining. These are, once again in order of decreasing log-likelihood, models 11, 7, 10, and 6. Of these remaining models, only model 7 is a non-trapping model. In Figure 5.16, we plot the best trapping (11) and non-trapping (7) model fits. Both fits appear visually similar and approximate the dynamics captured in the data fairly well. However, as with the other drugs considered, a further round of experiments would be necessary to draw definitive conclusions.

5.8 Discussion

In this chapter, we sought to apply the previously developed OED methodology (Chapter 4) to investigate its feasibility and effectiveness when working with real experimental patch-clamp data. Specifically, we designed optimal protocols, and collected data under these protocols, to assist in determining the underlying binding dynamics of five known hERG-blocking drugs. By fitting a set of candidate models to this collected data, we were able to infer drug-specific mechanistic insights. In Figure 5.17, we plot a summary of these results, indicating, for each drug considered, the model with the highest dual fit log-likelihood, and the models deemed infeasible. Taking bepridil as an example, our results suggest that an accurate model of bepridil binding requires the inclusion of an inactive binding mechanism. Kamiya et al. (2006) observed an open-binding preference in bepridil hERG binding, so it is of interest that our model fitting suggests that some degree of inactive binding also occurs. Our results also imply that, should bepridil become trapped, there is some degree of altered hERG gating rates (e.g. via the k_{trap} parameter in models 6 and 10). Model 10 appears to be the most appropriate model for bepridil binding, fitting both collected datasets well, and agreeing with previous results suggesting that bepridil does exhibit trapping behaviour. The results for quinidine, on the other hand, suggest that quinidine is a non-trapping drug, reinforcing the results from Sánchez-Chapula et al. (2003). Based on the log-likelihoods of the dual fits, model 7 appears the most appropriate model of quinidine dynamics, enabling binding to both inactive and open binding at differing rates, with no trapping mechanism.

We note that, in contrast to Chapter 4, there is no clear difference between slow-binding and fast-binding drugs in the effectiveness of the OED

methodology. We suspect that the inclusion of step length variation in the protocol optimisation procedure (allowing very short voltage steps) improves the method’s ability to discriminate between fast-binding dynamics here.

The work in this chapter was motivated by the results from Lei et al. (2024). We can compare the models ruled in/out by Lei et al. with the results of our OED methods. For quinidine and diltiazem, Lei et al. determined that all fifteen candidate binding models were plausible given the available data, with only subtle differences in the Root-Mean-Square Difference (RMSD) of each model fit. In comparison, for both of these fast-binding drugs, our OED methodology has reduced the set of feasible models down to five. Similarly for chlorpromazine, all fifteen binding models were deemed ‘as plausible as CiPAv1.0’ by Lei et al. with minimal differences in the RMSD values, while our methods reduce the feasible set down to four. Our methodology appears less beneficial for the slower binding drugs, for which Lei et al. had more success in model discrimination. Based on the Milnes block data available, Lei et al. were able to rule out several models as implausible for both verapamil and bepridil. For verapamil, we have reduced the size of the feasible set from eleven to five, while for bepridil, we have actually retained a few models (1, 2i, and 8) that were previously ruled implausible by Lei et al. These retained models are all non-trapping models, which, as we have noted, do not reflect the previously observed trapping behaviour of verapamil. As we will discuss shortly in further detail, the experiments in this chapter were performed at room temperature, while the Milnes data used by Lei et al. were collected at physiological temperature. This perhaps explains the discrepancies here; it is possible that trapping behaviour is more pronounced at physiological temperatures, aiding in ruling out non-trapping models.

Several assumptions and experimental choices were made to ensure practical and effective data collection, protocol optimisation, and model fitting. Firstly, as we have just noted, all SyncroPatch experiments were performed at approximately room temperature conditions. Cell catching and sealing processes in the automated patch-clamp environment tend to be less reliable at physiological temperatures, so working at lower temperatures enables a higher experimental success rate, enriching the quality of collected data. It is well known, however, that temperature differences result in significant changes in hERG gating mechanisms Lei et al. (2019a), inevitably affecting any observed binding dynamics. We, therefore, provide the caveat that any conclusions on drug binding dynamics are specific to room temperature conditions, and propose further experiments at physiological temperatures as verification. We also note that, while all experiments were performed in the same lab, under similar conditions, the effective experimental temperature is subject to change, resulting in potential subtle differences in experiments performed at different times on different days.

We employed several post-processing procedures to remove experimental artefacts introduced by the automated patch-clamp procedure. Leak correction and full block subtraction are both effective methods for accounting for leak and other contaminating current artefacts. However, these methods can only partially correct for their respective artefacts. For example, when performing leak correction, we are only able to *estimate* the leak currents via a fitted linear regression model, and this estimate is fairly sensitive to noise and other artefacts. Lei et al. (2020) introduce an *artefact model*, modelling the various experimental artefacts introduced by patch-clamp experiments. Further work applied this artefact model to resolve artefacts in sodium current patch-clamp recordings (Lei et al., 2025). It would be

useful to apply this artefact model to the data processed in this chapter to reduce the influence of any remaining artefacts.

QC methods were used to filter out data from wells with unphysiological behaviour and/or significant artefacts. We noted that, particularly in the Milnes data, an artefact, possibly caused by a gradual temperature increase, was producing unusual results in some wells. Given that we had sufficient data in other wells, we could filter these temperature-drift wells via QC. However, further investigations could consider retaining these wells by modelling the temperature drift using the temperature dependence relationships introduced by Lei et al. (2019a).

When fitting the drug-binding models to the average proportion block, we assumed, via the normal-ratio log-likelihood, that this average block was the ratio of two normally distributed random variables with identical standard deviation, σ . Given the form of our averaged block data, $\bar{z}_c(t)$, described in Equation 5.5, we suspect that the normal-ratio log-likelihood is inaccurate in this context. However, we suggest that the normal-ratio log-likelihood represents an improvement on the previously considered RMSD method used by Lei et al. (2024) when also working with average block data. Extensions to the work in this chapter could consider deriving a more appropriate likelihood, that takes the averaging process into account.

The results in Chapter 4 suggest that the effectiveness of the OED method is improved by a reduction in hERG model discrepancy. For this reason, in this chapter, we fitted the hERG model parameters to a ‘median’ I_{Kr} trace for each dataset (corresponding to a SyncroPatch plate). This additional hERG fitting step undoubtedly reduces any discrepancies between the data-generating hERG dynamics and our model; however, inevitable remaining discrepancies limit the reliability of any conclusions regarding specific drug-

binding dynamics.

Besides the recommendation of a further iteration of protocol design, there are several clear extensions to the work detailed in this chapter. Firstly, while the thirteen degrees of freedom considered in the protocol design process enable a fairly large scope for potential designs, the selection of this design space was rather arbitrary, motivated by a desire to produce improved designs while balancing computational expense. We suggest that, given sufficient computational resources, extending the scope of this design space would be a sensible next step to allow a larger diversity in protocol designs. Additionally, this design space could be tailored to the specific iteration of the OED procedure, depending (algorithmically or based on user discretion) on the dynamics of the remaining binding models. On a related note, the form of the objective function could also be altered depending on the OED iteration and available experimental budget. While we have observed the max-med objective ruling out at least half of the candidate models (as intended), subsequent iterations on a smaller subset of candidate models could favour separating a single model from the rest, especially if nested/redundant models have been eliminated from the design process. This could be achieved, for example, by maximising the maximum pairwise difference between fitted models currents.

We note that of the five drugs considered, four of the optimised protocols were very similar. This protocol form involved a short step up to a high voltage (around 50 mV), followed by two longer steps at lower voltages (around -50 mV). While the Milnes protocol is a useful starting point for observing binding dynamics, we suggest that, if these OED methods are implemented in a drug safety pipeline, a more informative starting protocol could involve the optimised protocol form we have observed, perhaps in combination with a 0 mV Milnes step, in effect, reducing the number of

OED iterations required and consequently reducing experimental cost.

The results summarised in Figure 5.17 relied on a manual process of ruling models in and out based on a variety of factors. As we noted in Section 5.7.1, it is not practical to use traditional model selection methods here (such as AIC or the likelihood-ratio test) given the high data sampling frequency and presence of experimental artefacts. However, we suggest that our manual ‘model selection’ process could be (partially) automated by specifying particular thresholds on parameters for which we expect certain nested models to reduce to others. This could aid the development of a fully automated iterative OED pipeline, improving the ease of implementation of these methods into existing cardiac safety assessment procedures.

In summary, we have illustrated the effectiveness of the previously detailed OED methodology (Chapter 4) in the context of real experimental patch-clamp data. By considering goodness-of-fit, model nesting, and model redundancies, we have substantially reduced the set of feasible binding models for five known hERG-blocking drugs, providing information to support the classification of binding dynamics for these drugs. A further iteration of protocol design is an obvious next step to validate these results and further discriminate between the remaining feasible models. The methods show promise as a practical tool for uncovering hERG binding dynamics during drug development and in aiding cardiac safety assessment.

Chapter 6

Learning binding dynamics from data using neural networks and sparse regression

Preamble

In the previous two chapters, we have assumed that the true model of drug binding dynamics is one of those illustrated in Figure 2.6. It has then effectively been a question of model selection — which of these proposed models best explains the data? In this chapter, we instead consider the possibility that the true binding dynamics cannot necessarily be modelled by the proposed dynamics in Figure 2.6. Using model discovery techniques that employ sparse regression and Machine Learning (ML) methods, the key question becomes: can we learn the underlying drug binding model structure from data? We rely on the assumption here that the binding dynamics can be captured with an ODE extension to our existing models of hERG gating. We begin by considering Universal Differential Equation (UDE)

methods to learn a neural network surrogate for the missing dynamics. We can then apply Sparse Identification of non-linear Dynamics (SINDy) methods to the learnt UDE to convert the uninterpretable UDE into an interpretable system of ODEs. Alternatively, it is possible to bypass the UDE step and apply SINDy methods directly to data. By considering several synthetic data examples, we observe that these model discovery methods can effectively return missing dynamics given sufficient informative data and ample prior knowledge. However, if data and/or prior knowledge are insufficient, system-dependent identifiability issues can be introduced, reducing the effectiveness of the proposed model discovery approaches.

6.1 Introduction

While mechanistic models of biological processes take biological assumptions and directly map these to model equations, this modelling process inherently relies on knowledge of the underlying mechanisms, while also introducing the potential for model misspecification. Alternatively, data-driven model discovery methods can bypass some of these biological assumptions, learning model equations directly from data (North et al., 2023). Hybrid models, combining mechanistic components with unknown parts learnt from data, allow us to balance the potential benefits of both approaches, ensuring that we can ‘bake in’ as much prior knowledge as feasible, while acknowledging the gaps in our understanding of the dynamics being modelled. In a field such as cardiac electrophysiology, technological advances in high-throughput data collection methods make model discovery methods all the more feasible. In this chapter, we investigate the feasibility of using Universal Differential Equations (UDEs) to learn missing dynamics in an ion channel drug-binding model. Combining ODEs with some Universal

Approximator (UA), often in the form of a neural network, UDEs offer a hybrid modelling approach that takes advantage of large-scale data and ML advances (Rackauckas et al., 2020). Although capable of capturing a vast diversity of dynamics, UDEs do suffer from a lack of interpretability; the complexity of the learnt neural network can make it difficult to derive mechanistic insights. For this reason, we also consider SINDy, a model discovery method that employs sparse regression techniques, such as the LASSO, to promote sparsity in the governing equations of a dynamical system (Brunton et al., 2016). By striking a balance between accuracy and parsimony, hyperparameters in the SINDy method can be fine-tuned to output a suitably interpretable set of equations that explain missing dynamics in a set of differential equations. We refer the reader to the literature review (Chapter 2) for further details on both UDEs and the SINDy method. Previous work (Rackauckas et al., 2020; Wu et al., 2025) has considered using UDEs in conjunction with SINDy to learn missing dynamics in the Lotka–Volterra model of predator-prey dynamics. As an extension of this work, we assess whether neglecting the UDE training step and applying SINDy directly to synthetic Lotka–Volterra data is a beneficial approach in terms of identifiability and computational efficiency. Consideration is also given to the case where data represent only a partially observed system of ODEs. We then apply both the UDE+SINDy and direct SINDy approaches in the context of drug-binding models to assess the abilities of the respective methods in learning missing binding dynamics from synthetic drug block data.

6.2 Learning missing dynamics in the Lotka–Volterra system

6.2.1 A fully observed system

We begin by considering the Lotka–Volterra system, often used for modelling predator-prey dynamics. Suppose x represents the population density of some prey species (e.g. rabbits) at time t . If we wish to model the competitive population dynamics between the prey species and some predator species (e.g. foxes), represented by y , then we can use the following set of differential equations

$$\frac{dx}{dt} = \alpha x - \eta xy \tag{6.1}$$

$$\frac{dy}{dt} = -\delta y + \gamma xy. \tag{6.2}$$

The model parameters α and δ describe the growth and death rates of prey and predators, respectively, while η and γ describe the influence of predator-prey interactions. We can generate (noisy) synthetic data, $\mathbf{x}(t) = [x(t), y(t)]$, from this model representing measurements of x and y over time. We will refer to this scenario, where we have observations of all model states, for some range of times $t \in [0, T]$, as the fully observed case. In Section 6.2.2, we will consider the partially observed case where, for example, only the x state is observed. In Figure 6.1A, we plot fully observed synthetic Lotka–Volterra data, $\mathbf{x}(t)$, generated with fairly high sampling frequency (a step size of 0.01) over the time period of $t \in [0, 5]$. Normally distributed noise has been added with a standard deviation of 0.2. We use a parameterisation of $\alpha = 0.3$, $\eta = 0.9$, $\delta = 1.8$, $\gamma = 0.8$ and initial conditions $\mathbf{x}(0) = [3.146, 1.537]$.

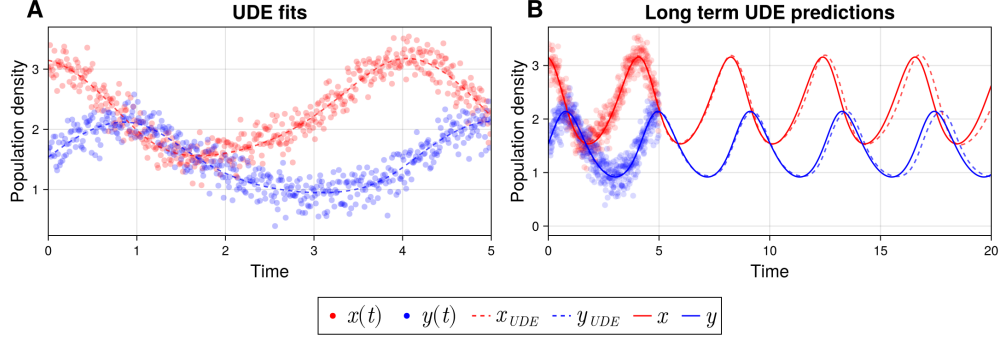


Figure 6.1: A: Synthetic Lotka–Volterra data with normally distributed noise (red and blue dots) compared to the five best UDE fits (red and blue dashed lines). Each UDE fit is visually indistinguishable. B: A comparison between the data-generating Lotka–Volterra dynamics (red and blue solid lines) and the five best UDE fits (red and blue dashed lines) over a longer time frame than the training data. We also replot the synthetic data from A.

Training a UDE

Now let us suppose that we know only part of the ODE system expressed in Equations (6.1) and (6.2), and we wish to learn the missing dynamics by training a neural network component, U , on the fully observed synthetic data, $\mathbf{x}(t)$. Specifically, let us assume that we know the αx and $-\delta y$ terms (and the values of α and δ), but the interaction terms, i.e. $-\eta xy$ and γxy , are unknown. We can set up this problem as follows

$$\frac{dx}{dt} = \alpha x + U_1, \quad (6.3)$$

$$\frac{dy}{dt} = -\delta y + U_2, \quad (6.4)$$

where U_1 and U_2 are the two outputs of a neural network which takes the current state of the system, x and y , as inputs. Next, we must consider the neural network architecture: how many hidden layers, and nodes per hidden layer, are required to capture the missing dynamics. When deciding on an appropriate level of complexity, traditional model fitting methods must consider the bias-variance trade-off to maintain a balance between

underfitting and overfitting the data. In contrast, for modern ML methods such as neural networks, it is widely accepted that, for higher levels of complexity, it is possible to overcome the bias-variance trade-off altogether. While the classical bias-variance trade-off can be illustrated by a U-shaped curve describing the loss in the test data for increasing levels of model complexity, Belkin et al. (2019) introduced a ‘double-descent’ curve for modern ML methods, extending the U-shaped curve to include a second reduction in loss for higher complexity models. Training larger neural network structures does, however, result in higher computational expense. It follows that, for the synthetic examples considered in this chapter, where the dynamics we are attempting to approximate with neural networks are fairly simple, we favour neural networks with fewer hidden layers and nodes to reduce computation times. We will begin with the simplest structure of one hidden layer with one node, and add nodes and/or layers as required. The neural network is then parameterised by ‘weights’ and ‘biases’ in the equations below, which we will represent with some parameter vector ϕ . We must also specify an activation function to ensure the neural network can exhibit non-linear behaviour. Based on the literature, a suitable activation function for UDE applications is the tanh function (Karlik et al., 2011), which we will use throughout this chapter. With a neural network of this simple structure, we can write out the equations explicitly as

$$U_1(x, y) = w_{2,1}h + b_{out_1} \quad (6.5)$$

$$U_2(x, y) = w_{2,2}h + b_{out_2}, \quad (6.6)$$

where h is defined as

$$h = \tanh(w_{1,1}x + w_{1,2}y + b_h), \quad (6.7)$$

and our weight (w) and bias (b) parameters are

$$\phi = [w_{1,1}, w_{1,2}, w_{2,1}, w_{2,2}, b_h, b_{out_1}, b_{out_2}]. \quad (6.8)$$

We can then fit the UDE defined in Equations (6.3) and (6.4) by training the neural network component on the synthetic data, $\mathbf{x}(t)$. We detail the UDE training methods in Section 6.4.1. In Figure 6.1A, we plot the five best UDE fits over the synthetic data. We note that all five fits appear indistinguishable and fit the data well. As described in Equations (6.5) and (6.6), the neural network structure for these fits was a single hidden layer with one node and a tanh activation function. Given this simple network structure, it is understandable that we see little variation in our fitted UDE solutions.

Next, we consider how these UDE fits perform over longer time frames. In Figure 6.1B, we plot a comparison between the ‘true’ data-generating dynamics and the UDE predictions over a range of $t \in [0, 20]$. We see that, while the predictions are close to the truth, there is a gradual drift apart. We suspect that this drift results from noise in the data, and, to test this, we repeat the UDE fitting process with noiseless synthetic data. Figure D.1 in Appendix D shows how the gradual drift is no longer present when fitting to this noiseless data.

We investigate the quality of the UDE fits further in Figure 6.2A by changing the initial conditions. Even within the original training time frame, $t \in [0, 5]$, this shift in initial conditions causes the fitted UDE to quickly diverge from the data-generating model. To emphasise that this divergence is not a noise-related issue, in Figure D.2 in Appendix D, we include an equivalent plot to Figure 6.2A for noiseless synthetic data, in which we observe the same effect. Clearly, the UDE has not captured the missing

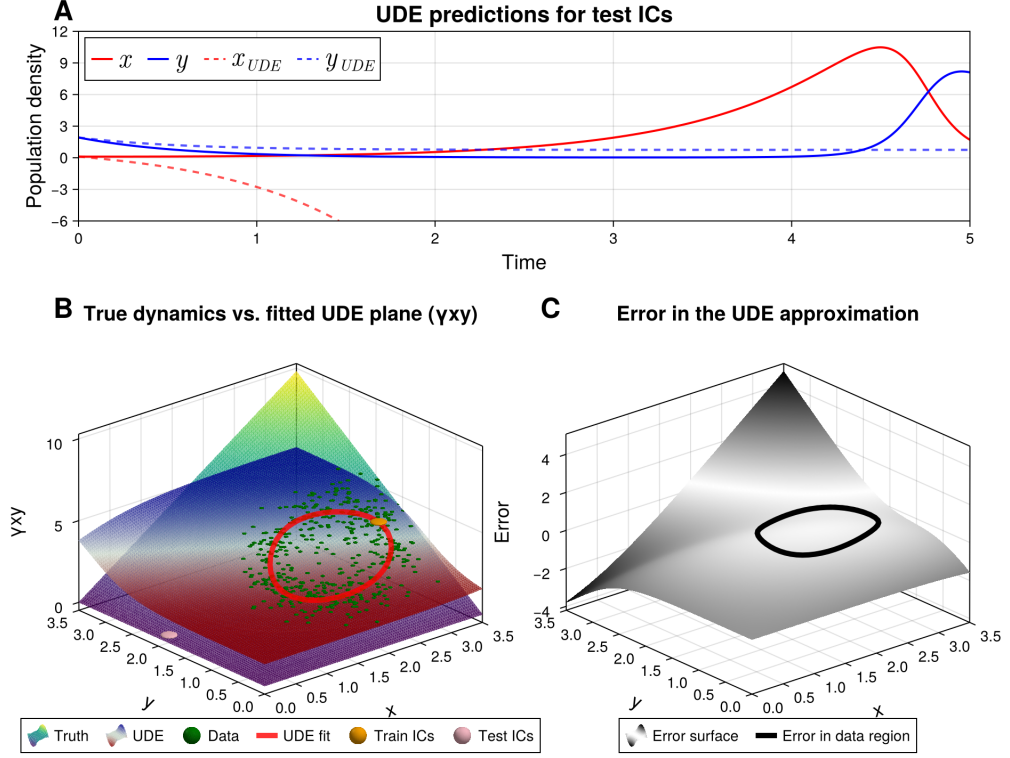


Figure 6.2: A: A comparison between the data-generating Lotka–Volterra dynamics and the five best UDE fits for the test data set with different initial conditions to the training data. B: A comparison between the best fit U_2 plane (red-blue) and the plane representing the data-generating γxy Lotka–Volterra dynamics (yellow-green). We also plot the synthetic data (green spheres) and the trajectory of the fitted UDE in the data region (red line). In orange, we indicate the initial conditions of the training data, and in pink, the initial conditions of the test data set. C: Error in the UDE approximation. The grey plane is evaluated by subtracting the best-fit U_2 plane from the plane illustrating the true dynamics, γxy . We emphasise, in black, the error in the region corresponding to the data.

dynamics as well as Figure 6.1B would suggest; despite its ability to extrapolate forward in time, it has no ability to extrapolate to a different region of the phase-space associated with different initial conditions. To see why, in Figure 6.2B, we plot a comparison between the planes defined by γxy and the neural network output U_2 , evaluated over a grid of x and y values. The two planes overlap nicely in the region where data is observed; however, the planes quickly diverge away from the data region. This is emphasised in Figure 6.2C, where we plot a plane representing the error in the UDE approximation. This error is calculated by simply subtracting U_2 from the true dynamics, γxy , over the plotted range of x and y values. Clearly, as we move away from the data-generating region (plotted as a black trace), the error quickly increases in magnitude. This nicely illustrates the potential pitfalls of relying solely on a fitted UDE; the predictive quality of our UDE is constrained by both the quality of our data and the range of the observed dynamics. These issues, in combination with the lack of interpretability of neural networks (especially for more complex network structures), naturally lead us to consider sparse regression methods such as SINDy.

Applying SINDy to our trained UDE

With a UDE trained successfully (at least in the region of observed data), we now have access to a surrogate for the missing dynamics, which we will call $\hat{U}$. We can take advantage of this surrogate and define a LASSO-style loss function to perform SINDy. To do this, we must first define a basis of candidate functions, which we will call $\Phi(X)$, where X is a matrix

containing the recorded data at N time points for each state (x and y)

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_N) \end{bmatrix}. \quad (6.9)$$

If we let $\Phi(\mathbf{X})$ consist of constants, $\mathbf{X}$, and quadratic and cubic combinations of $\mathbf{X}$, we can define it as

$$\Phi(\mathbf{X}) = \begin{bmatrix} 1 & x(t_1) & y(t_1) & x(t_1)y(t_1) & x(t_1)^2 & \cdots & x(t_1)^3 & y(t_1)^3 \\ 1 & x(t_2) & y(t_2) & x(t_2)y(t_2) & x(t_2)^2 & \cdots & x(t_2)^3 & y(t_2)^3 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & x(t_N) & y(t_N) & x(t_N)y(t_N) & x(t_N)^2 & \cdots & x(t_N)^3 & y(t_N)^3 \end{bmatrix}. \quad (6.10)$$

The optimisation problem can then be defined as

$$\min_{\boldsymbol{\beta}} \left\{ \frac{1}{N} \|\hat{\mathbf{U}} - \Phi(\mathbf{X})\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \quad (6.11)$$

where $\boldsymbol{\beta}$ is a vector of coefficients,

$$\boldsymbol{\beta} = \begin{bmatrix} \beta_{1,1} & \beta_{1,2} \\ \beta_{1,1} & \beta_{1,2} \\ \vdots & \vdots \\ \beta_{10,1} & \beta_{10,2} \end{bmatrix}, \quad (6.12)$$

that dictate which basis functions are active in our dynamical system. $\boldsymbol{\beta}$ has two columns corresponding to the two state variables, and ten rows corresponding to the ten basis terms in $\Phi(\mathbf{X})$. We specify that the L_1 -

norm here is an entry-wise norm such that

$$\|\beta\|_1 = \sum_{i,j} |\beta_{i,j}|. \quad (6.13)$$

We can then specify some regularisation parameter, λ , that promotes model sparsity via β . To ensure that each term in $\Phi(X)$ is treated ‘equally’, and hence that the coefficients in β are of the same scale, we also standardise $\Phi(X)$ by dividing each column by its L_2 -norm to give $\bar{\Phi}(X)$. For column $j \in [1, 10]$ of $\Phi(X)$, the standardised column is defined as

$$\bar{\Phi}(X)_j = \frac{\Phi(X)_j}{\|\Phi(X)_j\|_2}. \quad (6.14)$$

The selection of an appropriate λ is commonly done via cross-validation-based methods (Chetverikov and Liao, 2016). For this straightforward example, however, we can instead simply take a log-spaced grid of λ values and look for the ‘elbow’ in the model accuracy vs. complexity curve where we get a good trade-off between the two (similarly to the bias-variance U-shaped curve previously discussed). For a log-spaced range of λ_i for $i \in [1, 100]$ we determine an optimal β , which we label $\hat{\beta}_{\lambda_i}$. Further details of this β optimisation are included in Section 6.4.2. Using a withheld test data set, we then evaluate the loss for each $\hat{\beta}_{\lambda_i}$, and plot these in Figure 6.3A. Based on this plot, we select an optimal $\hat{\lambda}$ of 0.005. As detailed in Section 6.4.2, we threshold the coefficients in $\hat{\beta}_{\lambda_i}$, and run a further optimisation step with the remaining non-zero coefficients. We note that for $\hat{\lambda}$, we return the true data-generating dynamics, i.e. there remain two non-zero coefficients in $\hat{\beta}_{\hat{\lambda}}$ corresponding to the two missing xy terms. These non-zero $\hat{\beta}_{\hat{\lambda}}$ coefficients have the values of -0.9 and 0.8 , perfectly matching the data-generating parameters. Figures 6.3B and 6.3C illustrate the fitted model perfectly matching the true data-generating

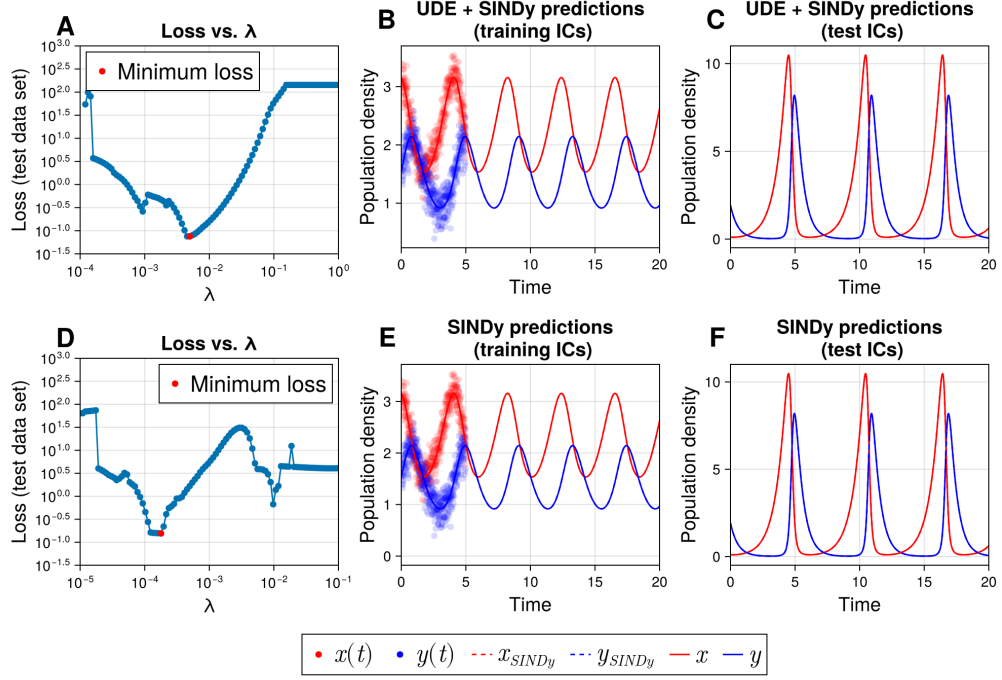


Figure 6.3: A: Comparison between the LASSO regularisation parameter, λ , and the resulting loss for a withheld test data set when applying SINDy to a fitted UDE. The λ resulting in minimum test data loss is highlighted in red. B: A comparison between the data-generating Lotka–Volterra dynamics and the best UDE+SINDy fit over a longer time frame than the training data. This fit is obtained using the optimal regularisation parameter, $\hat{\lambda}$. Note that the fitted model is visually identical to the data-generating model. C: A comparison between the data-generating Lotka–Volterra dynamics and the best UDE+SINDy fit (using $\hat{\lambda}$) for different initial conditions to the training data. The fitted model is again visually identical to the data-generating model. D: Comparison between the LASSO regularisation parameter, λ , and the resulting loss for a withheld test data set when applying SINDy directly to data. The λ resulting in minimum test data loss is highlighted in red. E: A comparison between the data-generating Lotka–Volterra dynamics and the best direct SINDy fit (using $\hat{\lambda}$) over a longer time frame than the training data. The fitted model is visually identical to the data-generating model. F: A comparison between the data-generating Lotka–Volterra dynamics and the best direct SINDy fit (using $\hat{\lambda}$) for different initial conditions to the training data. Again, the fitted model is visually identical to the data-generating model.

model. There is obviously nuance in the selection of λ , but we have shown how a sparse regression method is capable of returning the true missing dynamics in an interpretable form. As we noted when training a UDE, the quality of the UDE fit is determined by the data quality. In a similar manner, for this sparse regression case, the quality of the sparse regression model is determined by the quality of the neural network fit in our UDE. A neural network that does not capture the missing dynamics will mislead any subsequent sparse regression. For this reason, we also consider whether we can apply SINDy directly to our observed data, bypassing the UDE fitting step entirely.

Applying SINDy directly to our observed data

In the case where we wish to apply SINDy directly to our observed data, we can redefine our system of ODEs as follows

$$\frac{dx}{dt} = \alpha x + \bar{\Phi}(X)\beta_{*,1}, \quad (6.15)$$

$$\frac{dy}{dt} = -\delta y + \bar{\Phi}(X)\beta_{*,2}, \quad (6.16)$$

where $\beta_{*,1}$ and $\beta_{*,2}$ are, respectively, the 1st and 2nd columns of β . The LASSO objective then becomes

$$\min_{\beta} \left\{ \frac{1}{N} \|X - \hat{X}(\bar{\Phi}(X), \beta)\|_2^2 + \lambda \|\beta\|_1 \right\}, \quad (6.17)$$

where $\hat{X}(\bar{\Phi}(X), \beta)$ is the solution to the system of ODEs (in Equations (6.15) and (6.16)) at the N time-points corresponding to the data in X . To evaluate the objective in Equation (6.17), we, therefore, must substitute the standardised basis of candidate functions, $\bar{\Phi}(X)$, and the vector of coefficients, β , into the system of ODEs and solve at each iteration of the

optimisation algorithm (i.e. for each newly proposed β). In comparison to applying SINDy to the fitted UDE, this direct SINDy approach introduces a substantial increase in computation time in the sparse regression step, especially for larger systems simulating over longer time frames; however, this is a reasonable trade-off given that we have avoided training a neural network. In fact, for this simple Lotka–Volterra example, the two methods (UDE+SINDy and direct SINDy) are fairly comparable in computation time ($\sim 2 - 3$ minutes) when using small neural networks in the UDE step, while increases in this neural network size make the direct SINDy approach more favourable. We note that the original proposed SINDy methodology (Brunton et al., 2016) involves determining some numerical approximation of the gradients (e.g. via fitting differentiable splines to the data), consequently avoiding solving any ODEs. While this represents a computationally cheaper approach, our methods avoid making these gradient approximations, which, given that we can afford to solve the ODEs, is preferable. We apply the same LASSO methodology as detailed in the previous section and Section 6.4.2 and return a sparse solution. This time, we return a $\hat{\lambda}$ of 0.00018, and the remaining non-zero $\hat{\beta}_{\hat{\lambda}}$ coefficients once again correspond to the data-generating model. These non-zero $\hat{\beta}_{\hat{\lambda}}$ coefficients again perfectly match the data-generating parameter values of -0.9 and 0.8 . These results are illustrated in Figures 6.3E and F, where the long-term model predictions, and model predictions under different initial conditions, once again, perfectly match the data-generating model.

6.2.2 A partially observed system

In practical applications, it is often not possible to observe all states of a dynamical system. For this reason, we now consider an extension of the

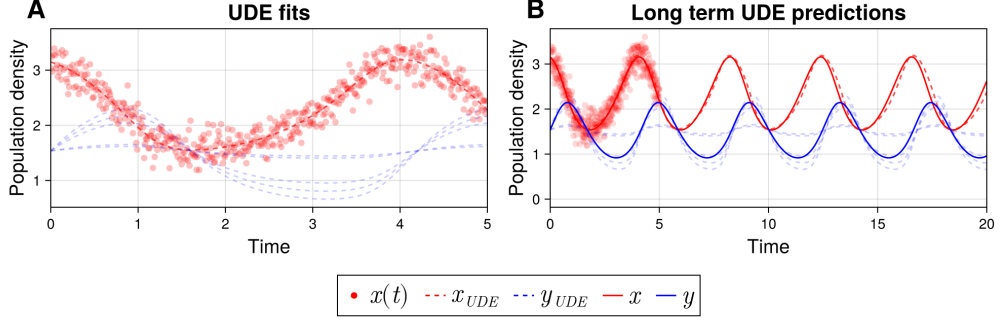


Figure 6.4: A: Synthetic Lotka–Volterra data with normally distributed noise (red dots) compared to the five best UDE fits (red and blue dashed lines). B: A comparison between the data-generating Lotka–Volterra dynamics and the five best UDE fits over a longer time frame than the training data.

previous section in which only one of the two Lotka–Volterra state variables is observed. We begin by generating synthetic data, with the same sampling frequency and noise as in the fully observed case. However, in this partially observed case, we only generate synthetic data for the state x . In a later section (Section 6.3), we will consider training a UDE on data from a partially observed hERG/drug-binding system. In the case of drug-binding ODEs, it is often feasible to approximate the initial conditions of all model states, even if we can only effectively observe one state. For this reason, we will assume here that the initial conditions of both x and y are known, despite not otherwise observing y . We can then fit a UDE to this partially observed data, $x(t)$, in a similar manner to the fully observed case (see Section 6.4.1 for further details). Figure 6.4A shows the output of the best five UDE fits (red and blue dashed lines) in comparison to the synthetic data (red dots). The UDE solutions representing the x state appear to match the data. In contrast, the y state solutions vary drastically. It seems that we are observing some form of functional non-identifiability. In fact, further investigation illustrates exactly where this identifiability issue arises. Let us define some transformation on our unobserved y variable $y^* = m(y - y_0) + y_0$ for $m \in \mathbb{R}$, where y_0 is the (assumed known) initial

condition of y . We can then redefine our Lotka–Volterra system as

$$\frac{dx}{dt} = \alpha x - \eta x y^*, \quad (6.18)$$

$$\frac{dy^*}{dt} = -\delta y^* + \gamma x y^*. \quad (6.19)$$

By solving this system for x and y^* , and transforming the y^* solution back to y via

$$y = \frac{y^* - y_0}{m} + y_0, \quad (6.20)$$

it is fairly trivial to see that $\forall m \in \mathbb{R}, m \neq 0$, solving for x gives identical solutions, while the y solution can vary infinitely. This *scaling symmetry* (Villaverde and Massonis, 2021), causing the solution for x to be invariant to a transformation in y , is clearly resulting from a structural identifiability issue — increases in sampling frequency, or reductions in noise, will not improve the fitted UDE. It follows that applying SINDy to a UDE fitted to partially observed data is not sensible in this scenario. Equivalently, direct application of SINDy will also fail to reliably capture the missing dynamics. While this result does not bode well for applying these methods to a partially observed system of ion channel drug-binding dynamics, we note that this is a system-specific result. In real-world applications, such as ion channel drug binding, while we are often unable to observe all ODE system states, we sometimes have access to additional model inputs (e.g. voltage protocol) and outputs (e.g. currents at different drug concentrations) which can potentially aid in reducing identifiability issues. Additionally, if we are only interested in modelling/predicting the observed state (e.g. x here), then we could feasibly include an additional regularisation constraint on the unobserved quantities (e.g. y) in our objective to ensure that we obtain a unique solution to the optimisation problem.

6.3 Learning missing drug-binding dynamics

In the previous section, we demonstrated the potential benefits and pitfalls of several methods for learning missing dynamics via a simple Lotka–Volterra example. We now return to models of drug binding dynamics. We wish to determine whether it is feasible to learn missing binding dynamics from data using the methods detailed in the previous section. Given the experimental practicalities of collecting ion channel drug binding data, we are only able to work with partially observed data; we can record I_{Kr} , which can be described as a function of the open state probability (see Equation (4.2)), but all other ion channel gating states are unobserved. In this section, therefore, we skip the practically unrealistic fully observed case and apply the model discovery methods to partially observed systems only.

6.3.1 Prior knowledge of state-specific binding

We will begin by considering an example where we have extensive prior knowledge of the missing dynamics. Specifically, we will assume that we know that a given ‘simulated’ drug binds to the hERG channel *only* in the inactivated state. To demonstrate this scenario, we must first generate synthetic data. We will use the Beattie model of hERG gating as illustrated in Chapter 2 in Figure 2.5c. If we then use binding model 2i from Figure 2.6, an example of solely inactive binding dynamics, we can define the following

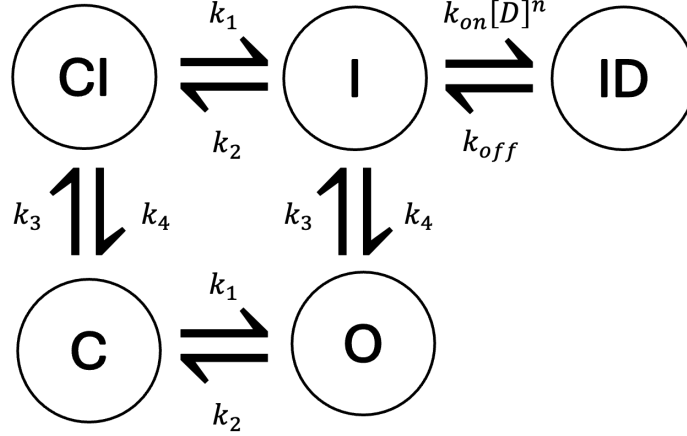


Figure 6.5: Markov diagram of the Beattie hERG model with a drug-binding component representing binding model 2i from Figure 2.6.

set of ODEs

$$\frac{dI}{dt} = -(k_2 + k_4)I + k_3O + k_1CI - k_{on}[D]^nI + k_{off}ID, \quad (6.21)$$

$$\frac{dO}{dt} = k_4I - (k_2 + k_3)O + k_1C, \quad (6.22)$$

$$\frac{dCI}{dt} = k_2I - (k_1 + k_4)CI + k_3C, \quad (6.23)$$

$$\frac{dC}{dt} = k_2O + k_4CI - (k_1 + k_3)C, \quad (6.24)$$

$$\frac{dID}{dt} = k_{on}[D]^nI - k_{off}ID. \quad (6.25)$$

In Figure 6.5, we also include the Markov diagram described by this set of ODEs. For the hERG model parameters (i.e. α_i and β_i defining each of the gating rates via $k_i = \alpha_i \exp(\beta_i V)$) and the binding parameters (k_{on} , k_{off}) we will use parameters fitted to the quinidine patch-clamp data described in Section 5.7.2 in Chapter 5. Specifically, the binding parameters are defined as $k_{on} = 7.027 \times 10^{-6} \text{ ms}^{-1} \text{ nM}^{-n}$ and $k_{off} = 2.339 \times 10^{-3} \text{ ms}^{-1}$. For simplicity, we will set the Hill parameter, n , to one; a biophysically reasonable starting point for models of ion channel block (Jarvis and Thompson, 2013). With this model setup, we generate control currents (i.e. $[D] = 0 \text{ nM}$) and drug currents for three different concentrations,

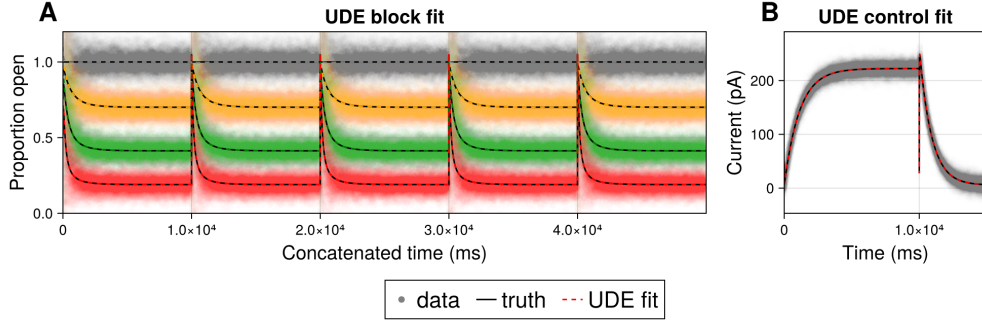


Figure 6.6: Fitting a UDE to drug block data. A: Simulated proportion open output from a UDE model fitted to synthetic data. Noisy data are plotted as grey, orange, green and red dots representing increasing drug concentrations of 0 nM, 150 nM, 500 nM, and 1500 nM, respectively. The equivalently coloured dashed lines indicate the best UDE fit. We also plot, as black lines, the true data-generating dynamics. B: Simulated control currents from the best UDE fit (red dashed line) over top of synthetic control current data (grey dots) and the true data-generating control current (black line).

$[D] \in \{150 \text{ nM}, 500 \text{ nM}, 1500 \text{ nM}\}$. Throughout this chapter, we evaluate hERG channel currents, I_{Kr} , as previously described in Equation (4.2), and reproduced here

$$I_{Kr} = g_{Kr} \cdot O \cdot (V - E_K). \quad (6.26)$$

For each concentration, we simulate five sweeps of the Milnes protocol with 10 s at resting potential between each sweep. We save the I_{Kr} data with a step size of 0.5 ms and add Gaussian noise with a standard deviation of 10 pA. We then divide each drug sweep, $x_c(t)$ (indexed by concentration $c \in \{150, 500, 1500\}$), by a control sweep, $y(t)$, to obtain the proportion block quantity, $z_c(t) = x_c(t)/y(t)$, used throughout this thesis. In Figure 6.6, we plot this synthetic data. Figure 6.6A shows the proportion block data for five sweeps of the Milnes protocol with no drug (grey), and with the drug at the three concentrations (orange, green, red). For each sweep, we concatenate the section of the proportion block data corresponding to the 10 s of the protocol at which $V = 0 \text{ mV}$. In Figure 6.6B, we plot the control current (in grey) for the full 15 s of the Milnes protocol.

Training a UDE

With synthetic data generated, we next define the structure of the UDE we will fit to this data. Assuming we know that the data represent an inactive binding drug, we can set up the UDE as follows

$$\frac{dI}{dt} = -(k_2 + k_4)I + k_3O + k_1CI + U_1, \quad (6.27)$$

$$\frac{dO}{dt} = k_4I - (k_2 + k_3)O + k_1C, \quad (6.28)$$

$$\frac{dCI}{dt} = k_2I - (k_1 + k_4)CI + k_3C, \quad (6.29)$$

$$\frac{dC}{dt} = k_2O + k_4CI - (k_1 + k_3)C, \quad (6.30)$$

$$\frac{dID}{dt} = -U_1. \quad (6.31)$$

Here, U_1 is the output of a neural network, and represents the unknown binding dynamics. In alignment with the data-generating model, the defined UDE structure enforces symmetry in the drug-binding dynamics; any additional mechanism learnt in the inactive state, I , has an equal and opposite effect in the inactive drug-bound state, ID . While this symmetry helps to promote conservation of mass in our ion channel system, we note that given there are no non-negativity constraints on U_1 , we could, in theory, obtain learnt mechanisms that are biophysically infeasible. However, given the proposed UDE form, the output of U_1 must be allowed to take both positive and negative values if we wish to accurately learn the data-generating dynamics. We also note that we have assumed that any drug-binding behaviour can be described by a single drug-bound state, ID . While this was the case for the data-generating model, other drug-binding models considered previously in this thesis (Figure 2.6) contain two or more drug-bound states. We suggest that the ‘universal’ property of UDEs ensures that a single drug-bound state (such as in Equation 6.31) should accommodate

the behaviour of these more complex models, provided the neural network is sufficiently large. Alternative UDE setups are discussed further in Section 6.5.

We assume that the inputs to the neural network component of the UDE are the two states that we anticipate the binding action is occurring between, I and ID, as well as the concentration, $[D]$, of the drug responsible for this binding. Given the form of the synthetic data, $z_c(t)$, we use the normal-ratio PDF $f_Z(z_c; b, \delta_y)$ from Equation (3.13) to define a normal-ratio log-likelihood function for training the UDE. In Section 6.4.1, we describe this log-likelihood and detail the UDE training methods. Over top of the noisy data in Figure 6.6A, we plot the best fit UDE (as dashed coloured lines) and the true model output (solid black lines). The fit appears to closely match the data-generating truth. We investigate further by comparing the control current generated by the fitted UDE to the true control current in Figure 6.6B. Again, it seems the UDE is a good approximation for the true dynamics.

In the Lotka–Volterra example, we probed the UDE fit by adjusting the initial conditions and considering model predictions over longer time-frames. In this drug-binding example, we instead consider whether the fitted UDE accurately captures the model dynamics under a different set of drug concentrations and a different protocol. These are both realistic experimental changes that should push the binding dynamics into new regions of the phase space. In Figure 6.7, we plot the fitted UDE output under a new range of drug concentrations, $[D] \in \{300 \text{ nM}, 1000 \text{ nM}, 3000 \text{ nM}\}$, and a new protocol. The new protocol is of the same length as the Milnes protocol; however, rather than a single 10 s step to 0 mV, the 10 s step is broken into three sections (3.34, 3.33 and 3.33 s respectively). Across these three sections, the voltage goes to 50 mV, then -60 mV , and finally 0 mV, before

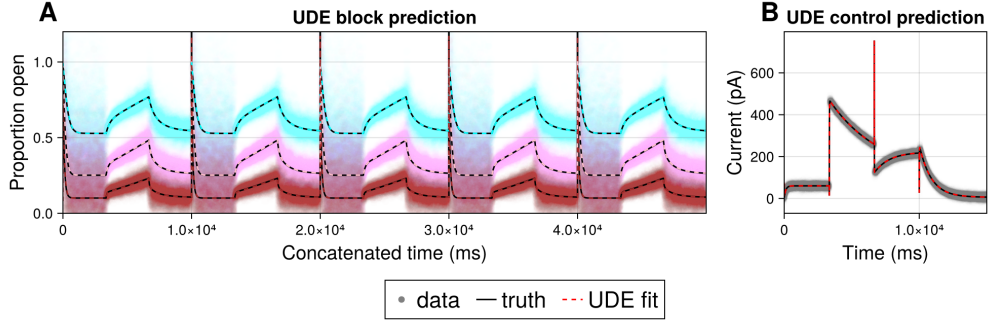


Figure 6.7: Assessing the quality of a UDE fit with test drug block data. A: Simulated proportion open output from a UDE model fitted to synthetic data. Noisy data are plotted as blue, pink, and brown dots representing increasing drug concentrations of 300 nM, 1000 nM, and 3000 nM, respectively. The equivalently coloured dashed lines indicate the prediction of the best UDE fit. We also plot, as black lines, the true data-generating dynamics. B: Predicted control currents under the best UDE fit (red dashed line) over top of synthetic control current data (grey dots) and the true data-generating control current (black line).

dropping back down to resting potential as per the Milnes protocol. As emphasised in Figure 6.7, this protocol induces differing dynamics (cf. the Milnes protocol output in Figure 6.6), which should be sufficient for assessing the quality of the UDE fit. The predictions appear to closely match the test data, reinforcing that the UDE is accurately capturing the missing dynamics. We can investigate this further by considering comparisons between the fitted UDE and the true dynamics across the phase space. If we fix the drug concentration to a given value (e.g. $[D] = 150 \text{ nM}$), then we can plot planes representing the fitted neural network output, U_1 , and the missing dynamics, $-k_{\text{on}}[D]^n I + k_{\text{off}} ID$, over a range of values of I and ID . In Figure 6.8A, we plot these two planes for $I, ID \in [0, 1]$. Considering conservation of mass in our equations, we expect $I + ID \leq 1$ at all times, so the plotted range covers the range of expected behaviours. We also plot, in red, the trajectory representing the region covered by the synthetic data (for $[D] = 150 \text{ nM}$). The UDE appears to match the true dynamics fairly well, only showing slight divergence towards the edges of the plotted region.

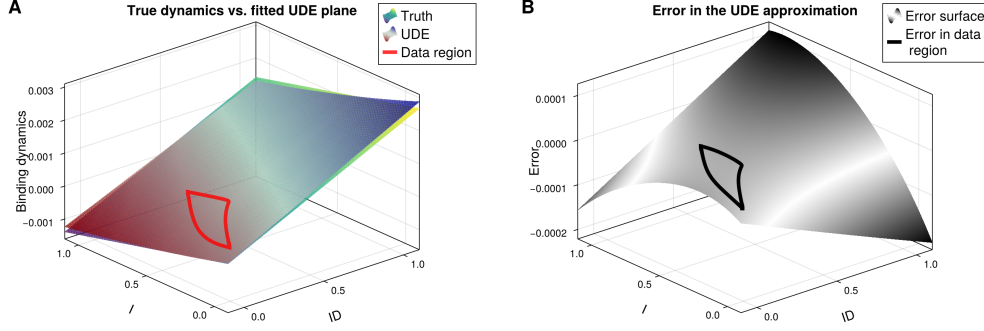


Figure 6.8: Comparisons between a fitted UDE and the data-generating dynamics. A: A comparison between the best fit U_1 plane (red-blue) and the plane representing the data-generating drug-binding dynamics (yellow-green) at a fixed drug concentration of 150 nM. We also plot the trajectory representing the true dynamics in the data region (red line). B: Error in the UDE approximation. The grey plane is evaluated by subtracting the best fit U_1 plane from the plane illustrating the true dynamics at a fixed drug concentration of 150 nM. We emphasise, in black, the error in the region corresponding to the data.

We also plot a plane representing the error in the UDE fit in Figure 6.8B. Calculated by subtracting the UDE plane from the true dynamics, we see the absolute error remaining, for the most part, below 1×10^{-4} , which should be suitably minimal to accurately apply the SINDy method to this fitted UDE.

Applying SINDy to our trained UDE

As we noted with the Lotka–Volterra example, a lack of interpretability makes it difficult to draw any mechanistic insights from the UDE alone. To address this interpretability issue, we again turn to SINDy. If we let $\hat{U}_1$ be the output (of dimension $N \times 1$) of the fitted neural network, then we can set up the following LASSO-style optimisation problem

$$\min_{\beta} \left\{ \frac{1}{N} \|\hat{U}_1 - \bar{\Phi}\beta\|_2^2 + \lambda \|\beta\|_1 \right\}. \quad (6.32)$$

where β is now simply the vector $\beta = [\beta_1, \beta_2, \dots, \beta_{10}]$. Once again, the basis of candidate functions then consists of polynomial combinations (up to cubic) of the neural network inputs. In this instance, we can define Φ as

$$\Phi = \begin{bmatrix} | & | & | & | & | & | & | & | & | \\ 1 & I & ID & [D] & I \times ID & I \times [D] & \dots & I^3 & [D]^3 \\ | & | & | & | & | & | & | & | & | \end{bmatrix}, \quad (6.33)$$

and the standardised $\bar{\Phi}$, $\bar{\Phi}$, is again defined equivalently to Equation (6.14) via

$$\bar{\Phi}_j = \frac{\Phi_j}{\|\Phi_j\|_2}. \quad (6.34)$$

Details of the optimisation methodology used to determine an optimal β are included in Section 6.4.2. In Figure 6.9A, we plot the normal-ratio log-likelihood for each optimised $\hat{\beta}_{\lambda_i}$ (over a log-spaced grid of λ_i) when calculating this likelihood with respect to some test data set (see Section 6.4.2 for details). We note that there are several discontinuities in Figure 6.9A corresponding to instances where the log-likelihood is evaluated as $-\infty$. When the difference between the test data and the fitted model is particularly large, we inevitably encounter underflow errors in calculating the normal-ratio PDF, f_Z (Equation (3.13)), used to evaluate the log-likelihood. These underflow errors do not necessarily pose a problem, provided we can still select an appropriate λ from those that do not result in a log-likelihood of $-\infty$. However, the discontinuities in the log-likelihood vs. λ plot do suggest potential instabilities in this SINDy approach when applied to a fitted UDE. Nevertheless, we indicate the optimal $\hat{\lambda}$ corresponding to the maximum log-likelihood, in red, as a value of 4.86×10^{-11} . After additional thresholding and optimisation steps (detailed in Section 6.4.2), for $\hat{\lambda}$, we

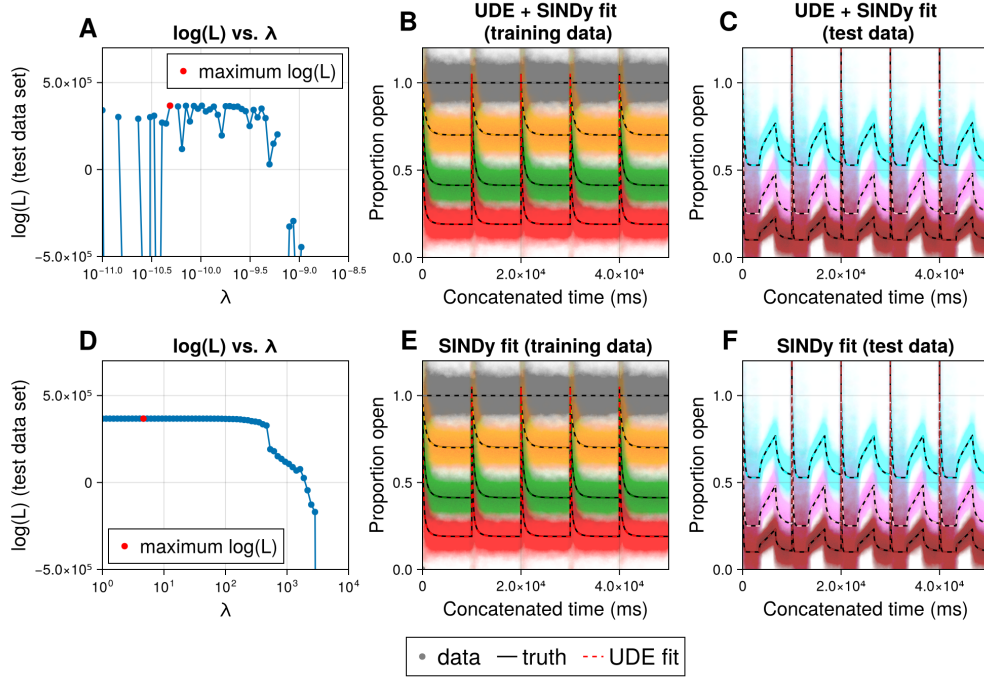


Figure 6.9: Using SINDy to learn missing drug-binding dynamics. A: Comparison between the LASSO regularisation parameter, λ , and the resulting log-likelihood for a withheld test data set when applying SINDy to a fitted UDE. The λ resulting in maximum test data log-likelihood is highlighted in red. B: A comparison between the data-generating drug-binding dynamics and the best UDE+SINDy fit. This fit is obtained using the optimal regularisation parameter, $\hat{\lambda}$. C: A comparison between the data-generating drug-binding dynamics and the best UDE+SINDy fit (using $\hat{\lambda}$) for a test data set. D: Comparison between the LASSO regularisation parameter, λ , and the resulting log-likelihood for a withheld test data set when applying SINDy directly to data. The λ resulting in maximum test data log-likelihood is highlighted in red. E: A comparison between the data-generating drug-binding dynamics and the best direct SINDy fit (using $\hat{\lambda}$). F: A comparison between the data-generating drug-binding dynamics and the best direct SINDy fit (using $\hat{\lambda}$) for a test data set.

return two non-zero coefficients of β corresponding to the ID and $I \times [D]$ columns of Φ . The coefficient values for these terms are -6.698×10^{-6} and 2.232×10^{-3} , closely matching the true data-generating dynamics of $-7.027 \times 10^{-6} [D]^n I + 2.339 \times 10^{-3} \text{ID}$. In Figure 6.9B, we plot the final model fits to the Milnes block data, while in Figure 6.9C we plot a comparison between model predictions and data under the test data experimental settings. The fits appear excellent, unsurprising given we have effectively returned the true model dynamics.

Applying SINDy directly to our observed data

While the SINDy method combined with a fitted UDE appears to be effective, we will try fitting SINDy directly to the Milnes block data to assess whether this is a practical alternative. Given that we are now fitting to the proportion open data, our LASSO objective function can be defined as

$$\min_{\beta} \left\{ \sum_{c \in C} \sum_{t \in T} \log \left(f_Z(z_c(t); b(t, c, \tilde{\Phi}\beta), \delta_y(t)) \right) + \lambda \|\beta\|_1 \right\}, \quad (6.35)$$

where $b(t, c, \tilde{\Phi}\beta)$ is the modelled open proportion given the (standardised) set of basis functions, $\tilde{\Phi}$, and the corresponding sparsity-inducing coefficients β . When applying SINDy to our fitted UDE, it was possible to solve the fitted UDE surrogate and return approximates for all model states that could then be used when standardising Φ . In applying SINDy directly to $z_c(t)$, we no longer have access to the UDE surrogate, and so we cannot evaluate $\|\Phi\|_2$ prior to optimisation. It is possible to evaluate $\|\Phi\|_2$ at each optimisation iteration and replace the $\|\beta\|_1$ term in Equation (6.35) with $\|(\beta \cdot \|\Phi\|_2)\|_1$. While this produces the desired effect of treating each coefficient of β as approximately equally scaled, it does, however, lead to increases in computation time in an already fairly computationally expen-

sive method. As a simple alternative, we can instead consider standardising only the columns of Φ that contain the drug concentration, $[D]$, which, in contrast to the I and ID state variables, is known throughout. The values of $[D]$ are on a significantly different scale ($[0, 1500]$) to I and ID ($[0, 1]$), so we expect that standardising the columns containing $[D]$ should at least provide some improvements in the stability of LASSO optimisation. We define, $\tilde{\Phi}$, as this partially standardised version of Φ .

Once again, we can then select an appropriate λ by taking a log-spaced range of λ_i , optimising the objective in Equation (6.35), and evaluating the normal-ratio log-likelihood for each $\hat{\beta}_{\lambda_i}$ with respect to a test data set. We plot these log-likelihoods in Figure 6.9D. While we no longer observe the discontinuities seen in Figure 6.9A, it is interesting to note that there is no obvious ‘elbow’ indicating the optimal trade-off between parsimony and model fit. In fact, for $\lambda \leq 100$, the test dataset log-likelihood shows very little change. This perhaps suggests that the missing dynamics in the test dataset are fairly similar to those observed in the training data. Consequently, for small λ , the trained models with higher levels of complexity (i.e. not penalised by the regularisation term in the objective) can also capture the dynamics in the test data well. It follows that it may be necessary to generate a more dynamically different test dataset. Regardless, we consider the solution, $\hat{\beta}_{\hat{\lambda}}$, corresponding to the $\hat{\lambda} = 4.64$ indicated in red in Figure 6.9D. After further thresholding and optimisation steps (see Section 6.4.2), we are left with only the ID and $I \times [D]$ terms, with coefficients of -7.168×10^{-6} and 2.372×10^{-3} , respectively. These values closely match the true data-generating parameters (-7.027×10^{-6} and 2.339×10^{-3}) and are marginally better estimates than those obtained via the UDE+SINDy method. In Figures 6.9E and 6.9F, we plot this good SINDy model fit against the training and test data, respectively.

6.3.2 No prior knowledge of state-specific binding

We now consider an extension to the previous section where we no longer have prior knowledge of whether the drug compound has an inactive or open binding preference. Regarding drug safety assessment, this is a more realistic level of prior knowledge, particularly for new drugs in development, where little is known of their binding dynamics. We can describe this scenario via the following UDE setup

$$\frac{dI}{dt} = -(k_2 + k_4)I + k_3O + k_1CI + U_1, \quad (6.36)$$

$$\frac{dO}{dt} = k_4I - (k_2 + k_3)O + k_1C + U_2, \quad (6.37)$$

$$\frac{dCI}{dt} = k_2I - (k_1 + k_4)CI + k_3C, \quad (6.38)$$

$$\frac{dC}{dt} = k_2O + k_4CI - (k_1 + k_3)C, \quad (6.39)$$

$$\frac{dD_b}{dt} = -U_1 - U_2, \quad (6.40)$$

where the neural network now has two outputs, U_1 and U_2 , corresponding to the inactive and open states, respectively. Note that we have renamed the ID state as D_b to indicate that this drug-bound state no longer exclusively represents inactive binding. To account for potential open binding interactions, the neural network now has an additional input, O , resulting in a total of four inputs (the others being I , D_b , and $[D]$). We then proceed with training the UDE on the synthetic data from Figure 6.6 as described in Section 6.4.1. We once again return a good fit to the synthetic data, as plotted in Figure 6.10. However, when assessing the predictions of the fitted UDE under the test data conditions, it is clear that we have not accurately captured the missing dynamics. In Figure 6.11, we plot, as dashed lines, the UDE predictions under the test protocol and concentrations as described in Section 6.3.1. We observe a significant divergence from the data-generating

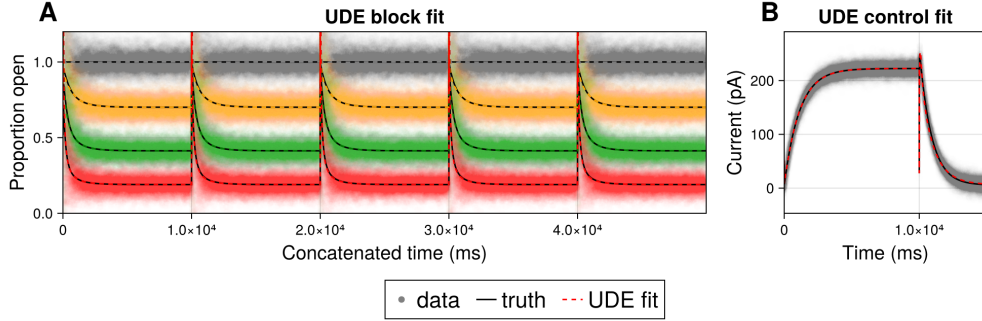


Figure 6.10: Fitting a UDE to drug block data with no prior knowledge of state-binding preference. Subplot descriptions are equivalent to those in Figure 6.6.

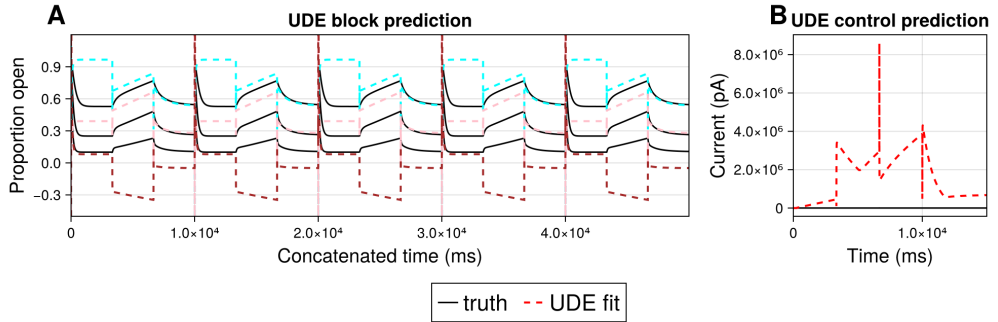


Figure 6.11: Assessing the quality of a UDE fit to test drug block data with no prior knowledge of state-binding preference. Subplot descriptions are equivalent to those in Figure 6.7. To aid visibility, and in contrast to Figure 6.7, we do not plot the noisy synthetic test data here. Note that the UDE control current predictions under the test protocol are significantly larger than the data-generating control currents.

truth (black lines) in both the modelled proportion open quantity (dashed coloured lines) and the control currents (dashed red line). In fact, the UDE prediction under the highest test concentration (plotted in brown) results in, at times, a negative proportion open — clearly, the fitted UDE is not an accurate approximation of the missing dynamics. It seems that some identifiability issue has been introduced, preventing the UDE from accurately learning the true binding dynamics. Given the form of our UDE, this is an unsurprising result; we are only observing one variable, O , while attempting to learn the missing dynamics, U_1 and U_2 , corresponding to two states, O and I . As discussed, we have also not specified non-negativity constraints on U_1 and U_2 , so it is likely that the state variables I , O , and D_b are straying from the feasible range of between 0 and 1.

Given that the UDE fails to accurately capture the missing dynamics, applying SINDy to the UDE outputs will inevitably prove ineffective. We can, however, attempt to use the direct SINDy approach. As described in Section 6.4.2, we select a $\hat{\lambda}$ and obtain an optimal solution, $\hat{\beta}_{\hat{\lambda}}$. However, we find that, possibly as a result of the identifiability issues observed in the UDE fit, there remain several non-zero terms in $\hat{\beta}_{\hat{\lambda}}$, none of which correspond to the true $I \times [D]$ and D_b terms. In Figure 6.12B, we plot a comparison between the data-generating dynamics and the SINDy fit to the training data, and, in Figure 6.12C, the predicted proportion open under the test data conditions. We note that the fit is fairly close to the true dynamics in the training data, but there are notable differences when predicting the test data dynamics, particularly for the highest test concentration. Considering the test data log-likelihoods plotted in Figure 6.12A, there are several discontinuities and a lack of smooth trend across the grid of λ values. We suspect that there are several local maxima in the optimisation space preventing SINDy from converging on the data-generating

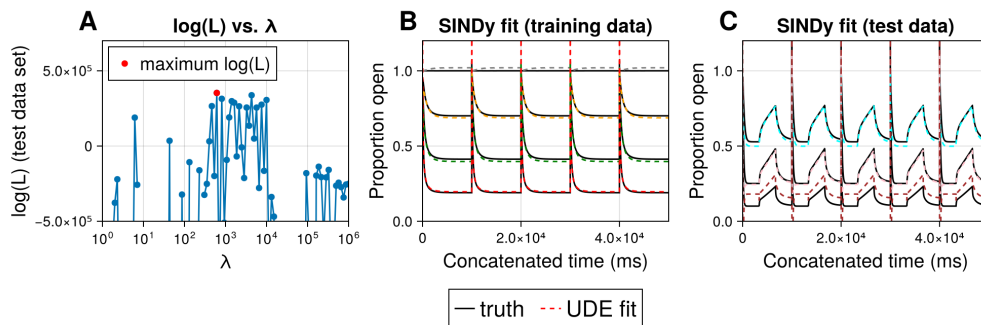


Figure 6.12: A: Comparison between the LASSO regularisation parameter, λ , and the resulting log-likelihood for a withheld test data set when applying SINDy directly to data. The λ resulting in maximum test data log-likelihood is highlighted in red. B: A comparison between the data-generating drug-binding dynamics and the best direct SINDy fit (using $\hat{\lambda}$). C: A comparison between the data-generating drug-binding dynamics and the best direct SINDy fit (using $\hat{\lambda}$) for a test data set.

dynamics. To illustrate this, we consider binding model 2 from Figure 2.6, a model representing exclusively open binding dynamics (in contrast to the exclusively inactive binding model 2i). If we fit model 2 to the model 2i synthetic Milnes block data (from Figures 6.6 and 6.10), we return a parameterisation (namely $k_{\text{on}} = -4.947 \times 10^{-6} \text{ ms}^{-1} \text{ nM}^{-n}$, $k_{\text{off}} = 0.00165 \text{ ms}^{-1}$) that can produce almost identical dynamics to the data generated under binding model 2i. In Figure 6.13A, we plot this model 2 fit over top of the data-generating model 2i dynamics; clearly, there are very minimal differences between these two models under the Milnes protocol, despite the obvious differences in model structures. In the presence of noise, it is not surprising that SINDy (and indeed the UDE) are unable to return the data-generating dynamics when it is apparent that several model structures can explain the dynamics in the data equally well. It seems that, by opening up the range of possible model structures (via a neural network or SINDy basis), we have only emphasised the issue motivating the OED methods applied in the previous two chapters. Once again, the synthetic data generated under the Milnes protocol contains insufficient information to learn the specific data-generating dynamics, and we instead observe instances of

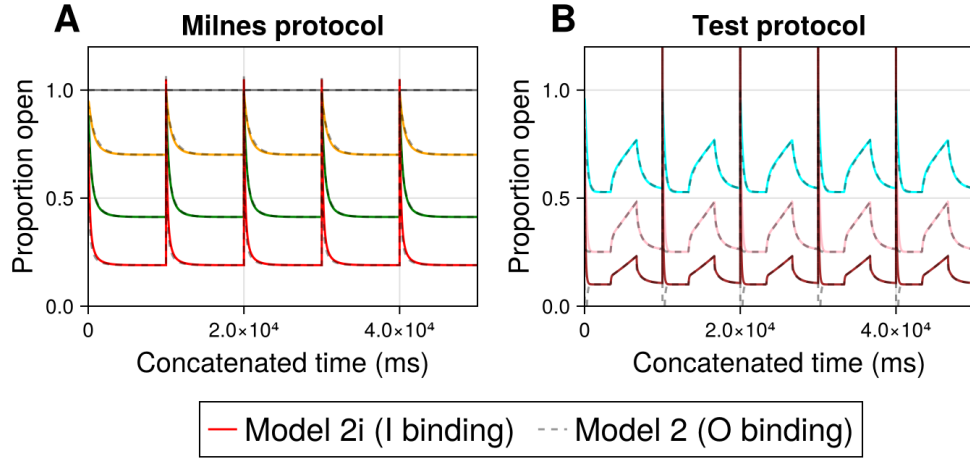


Figure 6.13: Demonstrating functional identifiability issues in drug-binding models. In both A and B, we plot the output of the data-generating model (model 2i) as coloured solid lines, and the fitted model (model 2) as dashed black lines. Model 2, has been fitted to the model 2i Milnes data from Figures 6.6 and 6.10.

functional non-identifiability. We note that, in Figure 6.13B, comparisons between model 2i and the fitted model 2 under the test protocol show some more noticeable differences, further emphasising how improved experimental designs can assist in separating model outputs as per Chapters 4 and 5.

6.4 Optimisation methods

For both Lotka–Volterra and drug-binding examples, we now provide further details of the methods used to train UDEs and optimise the SINDy objective.

6.4.1 UDE training

Lotka–Volterra system

To train the Lotka–Volterra UDEs from Section 6.2, we used a simple L_2 -norm loss function with the objective of minimising the pointwise sum of squared differences between the $\mathbf{x}(t)$ (or $x(t)$ in the partially observed case) data points and the UDE solution. By selecting some optimisation algorithm, we could then learn the neural network weights and biases, defined by ϕ (Equation (6.8)), that minimise the loss function. While traditional ML methods for training neural networks rely on gradient-based optimisation, given the small neural network structures we were working with here, we could opt for a derivative-free method. We used a multistart CMA-ES approach, which tended to outperform gradient-based methods such as ADAM and LBFGS in terms of computation time for these small networks. Specifically, our optimisation process involved running 50 distinct repeats in parallel. For each repeat, i , we sampled an initial parameter set, ϕ_i^0 , using the `Lux.jl` default neural network parameter initialisation. Starting from ϕ_i^0 , we then ran the CMA-ES optimisation algorithm, as implemented in `OptimizationCMAEvolutionStrategy.jl`, to return the optimal parameter set for the i th repeat, $\hat{\phi}_i$. When working with synthetic data, we could then verify that our neural network training had converged by comparing the minimised loss to the loss when running the ground truth model (i.e. when the model equations are fully known).

Drug-binding dynamics

The model complexity and increased data frequency in the drug-binding example from Section 6.3 prompted several adjustments to the UDE train-

ing methods (in comparison to the Lotka–Volterra example). To reduce training times and improve fitting stability, we ensured that all neural network inputs were on a similar scale. This was achieved by working with a standardised drug concentration input, $[\bar{D}]$, dividing each $[D]$ by the maximum concentration such that $[\bar{D}] = [D]/1500$. In terms of the neural network structure, after trialling with one and two nodes in the hidden layer, we found that three nodes were required to accurately capture the dynamics in the data. In contrast to the L_2 -norm loss function used in the Lotka–Volterra example, in the drug-binding example we used the normal-ratio log-likelihood, as introduced in Chapter 3. This was a suitable choice here, given that the synthetic data were generated by taking the ratio of two Gaussian random variables. If we have two independent Gaussian random variables, $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$, and we assume $\sigma_X = \sigma_Y$, then the PDF for $Z = X/Y$ can be defined by $f_Z(z_c; b, \delta_y)$ in Equation (3.13). For our synthetic open proportion data, $z_c(t)$, for concentrations $c \in C = \{150, 500, 1500\}$, the log-likelihood can be written as

$$\mathcal{L}(\phi|z_c(t)) = \sum_{c \in C} \sum_{t \in T} \log \left(f_Z(z_c(t); b(t, c, \phi), \delta_y(t)) \right). \quad (6.41)$$

Here, $b(t, c, \phi)$ is the output of the UDE at time t , for drug concentration c , given a set of neural network weights and biases ϕ . The $\delta_y(t)$ term, meanwhile, represents the signal-to-noise ratio in the ‘control’ hERG currents. We could have approximated these control currents using basis splines or a similar method; however, assuming no hERG model discrepancy, we instead simply used the true hERG model output (and true σ) to evaluate $\delta_y(t)$ in this synthetic data example. The final adjustment from the Lotka–Volterra neural network training methods pertains to the optimisation initialisation. As previously, we ran 50 repeats of CMA-ES in

parallel. However, to aid the optimisation, for each repeat, i , we sampled $j \in [1, 100]$ parameter initialisations, $\phi_{i,j}^0$, and used the $\phi_{i,j}^0$ corresponding to the largest log-likelihood (of these 100 samples) as our optimisation initialisation, ϕ_i^0 .

6.4.2 SINDy optimisation

Lotka–Volterra system

To perform the SINDy optimisation in Section 6.2, we began with a log-spaced range of λ_i for $i \in [1, 100]$. From a randomly sampled initialisation for β (each coefficient was sampled from a uniform distribution between -1 and 1), we then used the CMA-ES optimisation algorithm to minimise the LASSO objective in Equation (6.11) for each λ_i . We ran five CMA-ES repeats from randomly sampled initialisations and took the minimum loss, giving an optimal β for each λ_i , labelled $\hat{\beta}_{\lambda_i}$. We then used a withheld test data set to evaluate a ‘test loss’ for each $\hat{\beta}_{\lambda_i}$. These test data were generated identically to the training data; however, we used a different set of initial conditions. These test initial conditions were the same as those used to test the quality of our Lotka–Volterra UDE fit and are indicated in pink in Figure 6.2B. For the optimal λ , labelled $\hat{\lambda}$, we then performed an iteration of thresholded least-squares (similarly to Brunton et al. (2016)). This involved thresholding the coefficients in $\hat{\beta}_{\hat{\lambda}}$ by rounding to one decimal place and determining a least-squares solution for the remaining non-zero coefficients. Given the standardisation of $\bar{\Phi}(X)$, we then had to transform $\hat{\beta}_{\hat{\lambda}}$ back into the original scale by dividing through by the appropriate $\|\Phi(X)_j\|_2$ terms.

Drug-binding dynamics

As with the Lotka–Volterra example, in Section 6.3, we used a log-spaced grid of λ_i for $i \in [1, 100]$ to select an appropriate $\hat{\lambda}$ for SINDy. However, given the greater number of data points in this drug-binding example, a more thorough optimisation methodology was required. For each λ_i , we sampled 500 initialisations of β (with each coefficient again being sampled uniformly from between -1 and 1). We then took the β corresponding to the minimum loss as our starting point for CMA-ES optimisation, labelled β_0 . Next, we ran parallel repeats of CMA-ES, starting from β_0 , to determine an optimal $\hat{\beta}_{\lambda_i}$. While the UDE+SINDy approach involved ten CMA-ES repeats, for the direct SINDy approach, we limited the number of repeats to three, given the additional computation expense and the observed uniformity in the returned optimal solutions. We then, once again, used a withheld test data set to evaluate the test loss for each $\hat{\beta}_{\lambda_i}$. Specifically, these test data were the same as those considered in Figure 6.7. For the $\hat{\lambda}$ corresponding to the minimum test loss, we again thresholded the coefficients of $\hat{\beta}_{\hat{\lambda}}$, rounding to one decimal place, before reversing the normalisation transform. With the terms back in their original scale, we performed a final optimisation step on the remaining non-zero coefficients using an updated objective with no regularisation term. For the UDE+SINDy method, the optimisation problem could be written as

$$\min_{\beta} \left\{ \frac{1}{N} \|\hat{U}_1 - \bar{\Phi}\beta\|_2^2 \right\}, \quad (6.42)$$

while for the direct SINDy approach it was

$$\min_{\beta} \left\{ \sum_{c \in C} \sum_{t \in T} \log \left(f_Z(z_c(t); b(t, c, \tilde{\Phi}\beta), \delta_y(t)) \right) \right\}. \quad (6.43)$$

Based on these objectives, we then ran ten (three for direct SINDy) further repeats of CMA-ES, each beginning from the optimal values in $\hat{\beta}_{\lambda}$.

6.5 Discussion

In this chapter, we have investigated model discovery methods for learning models of ion channel drug binding dynamics from data. In contrast to the model selection methods from the preceding Chapters 4 and 5, these methods do not necessarily require the specification of a set of candidate models, enabling greater flexibility in the learnt model structures. With a specific focus on UDE and SINDy methods, we have demonstrated the feasibility of model discovery in several synthetic data examples using the Lotka–Volterra model of predator-prey dynamics, and a model of drug-binding dynamics. Ultimately, we have found that, of the two methods considered, a UDE+SINDy and a direct SINDy approach, both have their limitations, particularly when we are unable to observe all states of a dynamical system and/or we have minimal prior knowledge of the form of the missing dynamics we are attempting to learn.

Regarding the UDE+SINDy approach, we have illustrated how a fitted UDE is often only capable of capturing the missing dynamics in the region of the phase space that is characterised by the data. While the ‘universal approximator’ component of UDEs ensures that they are well-suited to approximating complex dynamics, they are generally less effective as a predictive tool for mechanistic behaviour not captured in the training data. In some cases, such as the fully observed Lotka–Volterra example (Section 6.2.1), applying SINDy to the fitted UDE ensured that the true dynamics were returned. However, this was not the case for the partially

observed Lotka–Volterra example (Section 6.2.2) and the drug binding example with no prior knowledge of state-specific binding (Section 6.3.2). In both of these cases, a lack of identifiability resulted in the trained UDE inaccurately capturing the dynamics in the observed data, consequently preventing the SINDy algorithm from reliably determining the ‘correct’ model equations. While neural networks are widely used to approximate the dynamics of physical systems, their predictive limitations in the context of UDEs, as emphasised in this chapter, are often not well appreciated. Similarly to Loman and Baker (2025), we suggest that an assessment of functional identifiability (e.g. via an ensemble of fitted UDE predictions) is an important consideration when fitting UDEs to data.

We have favoured the use of small neural network structures (with $\mathcal{O}(10)$ parameters) throughout this chapter. This choice was made to reduce computational expense, given that we were approximating relatively simple dynamics. When working with real data, in instances where the true dynamics are unknown, we suggest that employing larger networks is a more practical approach to increase the flexibility of the UDEs, ensuring more complex dynamics can be well approximated.

When fitting UDEs to synthetic drug-binding data, we made several choices with regard to the UDE setup. Notably, we attempted to fully capture the functions that describe the missing binding dynamics, as opposed to simply learning the functions/parameters defining the binding rates between model states. Consequently, to ensure the neural network outputs could accurately approximate the missing dynamics, we did not apply non-negativity constraints on the neural network output. However, given the identifiability issues observed when fitting UDEs to synthetic drug-binding data, it may be useful to reduce the scope of the fitted UDE, and instead assume we only wish to learn the specific rate functions characterising the

interactions between states (e.g. I-D_b and O-D_b). By incorporating this additional prior knowledge into our training regime, we may reduce the ability of the UDEs to characterise more complex dynamics. However, further prior knowledge would inevitably aid identifiability, assuming the missing dynamics can be accurately approximated by the specified state interactions.

Both the UDE+SINDy and direct SINDy approaches relied on some implementation of the SINDy algorithm. In the UDE+SINDy approach, a fitted UDE provided us with an approximation of the gradients of the missing dynamics, which could be used to optimise the SINDy objective at relatively low computational expense. This did, however, result in SINDy solutions that were sensitive to the quality of the UDE approximation. In contrast, our direct SINDy approach relied on solving the ODE system at each optimisation iteration, which removes potential issues introduced by the UDE approximation, but loses the computationally cheap properties of the standard SINDy algorithm. When applying direct SINDy to the fully observed Lotka–Volterra case, we could have alternatively approximated the gradients numerically by, for example, fitting (piece-wise polynomial) basis splines to the data and differentiating. However, when considering cases of partially observed data (e.g. in drug-binding data), approximating the gradients of unobserved states is less straightforward, and often infeasible. Ultimately, given the additional neural network training required by the UDE+SINDy approach, the two approaches were roughly equivalent in terms of computation times. We note, however, that the computational expense of both methods is dependent on the size of the network/basis used.

To conclude, the UDE+SINDy and direct SINDy methods presented in this chapter show promise when sufficient data and prior knowledge are avail-

able. However, both methods are prone to system-dependent identifiability issues. We recommend that, when applying to real experimental data, the considered system is first tested for functional identifiability to determine the possibility of reliably returning any missing dynamics. With regard to learning missing drug-binding dynamics, we suggest that these methods, in combination with the previously detailed OED methods in Chapters 4 and 5, could prove effective in learning dynamics from data, should sufficient prior knowledge be available to reduce identifiability issues.

Chapter 7

Discussion

Preamble

Mathematical models are invaluable tools for understanding complex dynamical systems. In a qualitative sense, models can provide deeper insights into the specific mechanisms that define a system, while quantitatively, they can aid prediction and accurately describe dynamics via the parameters that define them. In this thesis, we have investigated model selection and model discovery methods for improving mathematical models of drug binding mechanisms for biological ion channels. We have focused on the hERG channel, a channel that is both critical for healthy cardiac function and particularly prone to drug block. Our methods have uncovered both qualitative and quantitative insights into hERG binding dynamics, suggesting the feasibility of these methods in improving cardiac safety assessment procedures for pharmaceutical drug compounds. We now summarise the key contributions of this thesis, before considering several remaining open questions and future research directions.

7.1 Key contributions

After introducing the context and thesis aims in Chapter 1, we detailed the key biological and mathematical concepts in Chapter 2, while also summarising previous related research. In Chapter 3, we then introduced the framework for fitting drug-binding models to data. Previous research by Lei et al. (2024) fitted binding models to post-processed data representing the proportion of unblocked hERG channels. Through synthetic data experiments, we investigated whether this approach was sensible, with a particular focus on the benefits of this method as they pertain to model discrepancy. For two cases of (fairly extreme) model discrepancy, we demonstrated that, in comparison to fitting directly to the hERG current, fitting to the unblocked proportion provided more appropriate levels of uncertainty around model parameter estimates.

Another notable contribution of Chapter 3 was the introduction of a normal-ratio noise model when fitting binding models to data representing the unblocked proportion. In contrast to previous methods assuming Gaussian noise (Lei et al., 2024), this noise model more accurately describes the unblocked proportion as the ratio of two normally distributed random variables. We observed that applying this noise model to two synthetic model discrepancy cases resulted in a significant reduction in the error in parameter estimates, suggesting that this fitting method was an appropriate approach moving forward. Given these results, the normal-ratio fitting methods described in this chapter were used throughout the rest of this thesis.

In Chapter 4, we turned our attention to drug binding model identification, the key focus of this thesis. Motivated by model selection limitations previously addressed by Lei et al. (2024), we introduced an OED method-

ology for discriminating between a set of candidate drug binding models. Incorporating the normal ratio error model from Chapter 3, these methods demonstrated, via a synthetic data study, how a set of binding models could be effectively discriminated by developing improved voltage protocol designs. We also tested the method in the presence of model discrepancy, noting that a large discrepancy between the assumed hERG model and the data-generating hERG model led to a reduction in the method’s accuracy. This result suggested that effective application of the OED method in a real data setting would rely on a well-calibrated model of the underlying hERG gating mechanisms.

In Chapter 5, we applied the methods from Chapter 4 to produce optimised experimental designs and collect real experimental patch-clamp data. The specifics of the OED methodology were adjusted to extend the design space, while hERG model discrepancy issues were reduced by additionally fitting the parameters of the underlying hERG model. Based on model fits to the collected data, we were able to significantly reduce the set of feasible binding models for several known hERG-blocking drug compounds, demonstrating the effectiveness of the OED methods in aiding model selection in a real data context.

Finally, in Chapter 6, we considered model discovery methods, applying neural networks and sparse regression to learn missing dynamics in ODE systems from data. Through synthetic data examples, we demonstrated the potential limitations of these methods with regard to identifiability, particularly when not all model states are observable, or if insufficient prior knowledge is available.

7.2 Open questions and future directions

So, what questions remain? And what further research avenues do these questions prompt? Let us return to the overall motivation of this thesis. We sought to develop methods to improve models of drug-binding mechanisms in biological ion channels and aid the cardiac safety assessment of drug compounds. These problems were addressed through two distinct, yet related, approaches: model selection and model discovery. Regarding model selection, the proposed OED methodology introduced in Chapter 4, and trialled in Chapter 5, represents a promising tool for enhancing our understanding of drug binding mechanisms. The results in Chapter 5, in particular, suggest that implementing this methodology as part of routine cardiac safety assessment procedures could be an effective way of improving the classification of drug binding mechanisms, potentially assisting in uncertainty quantification of proarrhythmic risk. However, several considerations regarding the practical application of the proposed OED methodology remain. We have shown that, for a single iteration of the method, we can reduce the set of candidate models by at least half, but will further iterations produce the desired effect of eventually reducing the feasible set down to one? And is this necessary, given the motivations of this work? As a result of model nesting in the set of candidate models, we have seen that, for some binding dynamics, a more complex model can reduce to a simpler nested equivalent, making the separation of the two models practically impossible. In Chapter 5, we filtered out the more complex models in these nesting cases, but two non-nested models could feasibly produce identical outputs under all practical experimental setups. In such a case, provided that predictive outputs used for assessment of arrhythmic behaviour (e.g. action potentials) are also identical, this inseparability is perhaps not an issue; as long as we can use the binding models to accurately predict risk, we

do not necessarily need to know the exact underlying model mechanisms.

An additional factor is the model discrepancy introduced by various modelling assumptions and experimental artefacts, which inevitably limit the reliability of the likelihood function used to fit the binding models. This discrepancy affects the experimental designs produced, as well as any conclusions as to which binding models better explain the data. As discussed in Chapter 5, incorporating the artefact model introduced by Lei et al. (2025) would go some way to improving the accuracy of the likelihood. We could also introduce additional extensions to account for any temperature drift effects observed, while using a mixed-effect model to account for any well-to-well variability in the observed block effect (as opposed to simply taking the average across wells). With a more reliable likelihood, we could then more accurately quantify uncertainty in parameter estimates. This uncertainty could then feed forward through the protocol design procedure, improving the robustness of the designs and any consequent model fits.

While the proposed OED method is fairly straightforward to implement, several extensions could be made to automate the process and allow for more rapid iteration. As mentioned in Chapter 5, an improved starting protocol could provide more information than the Milnes protocol we have used. By specifically developing a protocol designed to emphasise a range of dynamics, the data collected in the first experimental run could provide significantly more information, potentially reducing the total number of iterations required. In its current form, for each drug considered, and at each OED iteration, optimising a new protocol design takes approximately 24 hours, so reducing the number of these iterations is preferable. To reduce computation time, we could, for example, generate representative synthetic data from all candidate models, for a wide range of parameterisations, and

train some form of emulator (e.g. Gaussian process) to rapidly generate new protocols for a given set of parameterised candidate models, without having to solve any ODEs. Alternatively, a set of pre-designed protocols could be developed for a wide range of feasible binding dynamics, allowing us to select a suitable protocol at each iteration via a lookup table or similar method. The manual analysis methods used to rule out nested or redundant models could also be automated by comparing the fitted parameters of specific model combinations and specifying thresholds to dictate when models could be ruled in or out. In combination with automated post-processing procedures, this could enable near real-time experimental design, generating newly optimised protocols immediately after collecting data.

Another consideration is whether the proposed set of candidate binding models in Figure 2.6 represents a comprehensive set of feasible binding dynamics; by considering this set of Markov models only, are we excessively restricting the model selection problem? We could extend the model set by enumerating a wider range of Markov models with additional states and dynamics; however, this could significantly increase computational expense in both the OED and model fitting steps. The suggestion of a restrictive set of candidate models motivated the model discovery approach considered in Chapter 6. In comparison to the OED methodology, the model discovery results were less promising due to identifiability-related limitations; however, we suggest that combining these model discovery methods with some form of OED methodology could prove fruitful. This could involve generating experimental designs based on a set of fitted UDEs, perhaps with the objective of optimising the explored phase space, similarly to the ‘space-filling’ objective considered by Mirams et al. (2024), with the aim of reducing identifiability issues.

7.3 Concluding remarks

To conclude, in this thesis we have investigated several methods for improving models of drug-binding in cardiac ion channels. By implementing these methods in the context of real experimental data, we have demonstrated their effectiveness in producing biological insights into the binding mechanisms of specific drug compounds. In doing so, we have illustrated the potential of these methods as practical tools for cardiac safety assessment in drug development. Existing *in vitro* and *in silico* safety assessment procedures are generally limited in their ability to distinguish between various models of drug binding, hence resulting in significant uncertainty in predictions of proarrhythmic risk. The OED procedure developed in this thesis could be incorporated as an additional safety assessment step, reducing uncertainty by determining an accurate model of binding dynamics.

While recent technological advances allow us to record high-frequency ion channel current data, these data are particularly prone to experimental artefacts, resulting in significant model discrepancies. Model discrepancy is often overlooked and can significantly bias parameter estimation, prediction, and any mechanistic insights derived from mathematical models. Throughout this thesis, we have placed a particular emphasis on accounting for model discrepancy, acknowledging its impact on the reliability of model fits and predictions. We encourage further consideration of model discrepancy in cardiac electrophysiology, and in other applications of mathematical biology.

Overall, we hope that the work presented in this thesis provides useful suggestions to improve the cardiac safety assessment of drug compounds, while raising interesting questions pertaining to modelling drug-binding mechanisms in biological ion channels.

Appendices

Appendix A

Model equations

A.1 hERG models

A.1.1 Beattie model

Given the state vector

$$\mathbf{x} = [\text{C}, \text{O}, \text{I}, \text{CI}], \quad (\text{A.1})$$

the full system of ODEs for the Beattie model (Beattie et al., 2018) can be defined as

$$\frac{\text{d}}{\text{d}t}\mathbf{x} = \begin{bmatrix} -k_1 - k_3 & k_2 & 0 & k_4 \\ k_1 & -k_2 - k_3 & k_4 & 0 \\ 0 & k_3 & -k_2 - k_4 & k_1 \\ k_3 & 0 & k_2 & -k_1 - k_4 \end{bmatrix} \mathbf{x}, \quad (\text{A.2})$$

for the voltage-dependent rates

$$k_1 = p_1 \exp(p_2 V), \quad (\text{A.3})$$

$$k_2 = p_3 \exp(-p_4 V), \quad (\text{A.4})$$

$$k_3 = p_5 \exp(p_6 V), \quad (\text{A.5})$$

$$k_4 = p_7 \exp(-p_8 V), \quad (\text{A.6})$$

parameterised by $\theta = [p_1, \dots, p_8]$.

A.1.2 Lu model

The Lu model (Lu et al., 2001) has the following state vector

$$\mathbf{x} = [\text{C1}, \text{C2}, \text{C3}, \text{O}, \text{I}], \quad (\text{A.7})$$

and the full ODE system can be defined as

$$\frac{d}{dt} \mathbf{x} = \begin{bmatrix} -a_{a0} & b_{a0} & 0 & 0 & 0 \\ a_{a0} & -b_{a0} - k_f & k_b & 0 & 0 \\ 0 & k_f & -k_b - a_{a1} - a_{ci} & b_{a1} & b_{ci} \\ 0 & 0 & a_{a1} & -b_{a1} - a_i & b_i \\ 0 & 0 & a_{ci} & a_i & -b_{ci} - b_i \end{bmatrix} \mathbf{x}, \quad (\text{A.8})$$

where k_f and k_b are voltage-independent rate parameters, and the remaining voltage-dependent rates are defined as follows

$$a_{a0} = p_1 \exp(p_2 V), \quad (\text{A.9})$$

$$b_{a0} = p_3 \exp(-p_4 V), \quad (\text{A.10})$$

$$a_{a1} = p_5 \exp(p_6 V), \quad (\text{A.11})$$

$$b_{a1} = p_7 \exp(-p_8 V), \quad (\text{A.12})$$

$$a_{ci} = p_9 \exp(p_{10} V), \quad (\text{A.13})$$

$$b_{ci} = p_{11} \exp(-p_{12} V), \quad (\text{A.14})$$

$$a_i = p_{13} \exp(p_{14} V), \quad (\text{A.15})$$

$$b_i = p_{15} \exp(-p_{16} V), \quad (\text{A.16})$$

for parameters $\theta = [p_1, \dots, p_{16}]$.

A.2 Drug binding models

For each binding model described in Figure 2.6, we now detail the ODEs for the additional binding states and the necessary extensions to the existing hERG model equations when assuming an underlying Beattie model. Instances of ‘ $\dots$ ’ indicate cases where the dO/dt or dI/dt equations have been extended from their respective Beattie model equations in Section A.1.1. Where unspecified, state equations and rate parameters are also as defined in Section A.1.1.

A.2.1 Model 1

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.17})$$

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.18})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O - k_{\text{off}} OD, \quad (\text{A.19})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I - k_{\text{off}} ID, \quad (\text{A.20})$$

for binding rate parameters k_{on} and k_{off} , drug concentration $[D]$, and Hill coefficient n .

A.2.2 Model 2

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.21})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O - k_{\text{off}} OD. \quad (\text{A.22})$$

A.2.3 Model 2i

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.23})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I - k_{\text{off}} ID. \quad (\text{A.24})$$

A.2.4 Model 3

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.25})$$

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.26})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O - (k_{\text{off}} + k_3) OD + k_4 ID, \quad (\text{A.27})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I + k_3 OD - (k_{\text{off}} + k_4) ID. \quad (\text{A.28})$$

A.2.5 Model 4

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.29})$$

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.30})$$

$$\frac{dCD}{dt} = -(k_1 + k_3) CD + k_2 OD + k_4 CID, \quad (\text{A.31})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O + k_1 CD - (k_{\text{off}} + k_2 + k_3) OD + k_4 ID, \quad (\text{A.32})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I + k_3 OD - (k_{\text{off}} + k_2 + k_4) ID + k_1 CID, \quad (\text{A.33})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 ID - (k_1 + k_4) CID. \quad (\text{A.34})$$

A.2.6 Model 5

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.35})$$

$$\frac{dCD}{dt} = -(k_1 + k_3) CD + k_2 OD + k_4 CID, \quad (\text{A.36})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O + k_1 CD - (k_{\text{off}} + k_2 + k_3) OD + k_4 ID, \quad (\text{A.37})$$

$$\frac{dID}{dt} = k_3 OD - (k_2 + k_4) ID + k_1 CID, \quad (\text{A.38})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 ID - (k_1 + k_4) CID. \quad (\text{A.39})$$

A.2.7 Model 5i

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.40})$$

$$\frac{dCD}{dt} = -(k_1 + k_3)CD + k_2 OD + k_4 CID, \quad (\text{A.41})$$

$$\frac{dOD}{dt} = k_1 CD - (k_2 + k_3)OD + k_4 ID, \quad (\text{A.42})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I + k_3 OD - (k_{\text{off}} + k_2 + k_4)ID + k_1 CID, \quad (\text{A.43})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 ID - (k_1 + k_4)CID. \quad (\text{A.44})$$

A.2.8 Model 6

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.45})$$

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.46})$$

$$\frac{dCD}{dt} = -(k_1 + k_3)CD + k_2 k_{\text{trap}} OD + k_4 CID, \quad (\text{A.47})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O + k_1 CD - (k_{\text{off}} + k_2 k_{\text{trap}} + k_3)OD + k_4 ID, \quad (\text{A.48})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I + k_3 OD - (k_{\text{off}} + k_2 k_{\text{trap}} + k_4)ID + k_1 CID, \quad (\text{A.49})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 k_{\text{trap}} ID - (k_1 + k_4)CID, \quad (\text{A.50})$$

where k_{trap} is a trapping rate factor parameter.

A.2.9 Model 7

$$\frac{dO}{dt} = \dots - k_{\text{on},O}[D]^n O + k_{\text{off},O} OD, \quad (\text{A.51})$$

$$\frac{dI}{dt} = \dots - k_{\text{on},I}[D]^n I + k_{\text{off},I} ID, \quad (\text{A.52})$$

$$\frac{dOD}{dt} = k_{\text{on},O}[D]^n O - k_{\text{off},O} OD, \quad (\text{A.53})$$

$$\frac{dID}{dt} = k_{\text{on},I}[D]^n I - k_{\text{off},I} ID, \quad (\text{A.54})$$

where $k_{\text{on},O}$, $k_{\text{off},O}$, $k_{\text{on},I}$, and $k_{\text{off},I}$ are state-dependent binding rate parameters.

A.2.10 Model 8

$$\frac{dO}{dt} = \dots - k_{\text{on},O}[D]^n O + k_{\text{off},O} OD, \quad (\text{A.55})$$

$$\frac{dI}{dt} = \dots - k_{\text{on},I}[D]^n I + k_{\text{off},I}^* ID, \quad (\text{A.56})$$

$$\frac{dOD}{dt} = k_{\text{on},O}[D]^n O - (k_{\text{off},O} + k_3) OD + k_4 ID, \quad (\text{A.57})$$

$$\frac{dID}{dt} = k_{\text{on},I}[D]^n I + k_3 OD - (k_{\text{off},I}^* + k_4) ID, \quad (\text{A.58})$$

where $k_{\text{off},I}^*$ is a function of the fitted binding rate parameters such that

$$k_{\text{off},I}^* = k_{\text{on},I} \frac{k_{\text{off},O}}{k_{\text{on},O}}, \quad (\text{A.59})$$

enforcing microscopic reversibility as per Colquhoun et al. (2004).

A.2.11 Model 9

$$\frac{dO}{dt} = \dots - k_{\text{on},O}[D]^n O + k_{\text{off},O} OD, \quad (\text{A.60})$$

$$\frac{dI}{dt} = \dots - k_{\text{on},I}[D]^n I + k_{\text{off},I}^* ID, \quad (\text{A.61})$$

$$\frac{dCD}{dt} = -(k_1 + k_3)CD + k_2 OD + k_4 CID, \quad (\text{A.62})$$

$$\frac{dOD}{dt} = k_{\text{on},O}[D]^n O + k_1 CD - (k_{\text{off},O} + k_2 + k_3)OD + k_4 ID, \quad (\text{A.63})$$

$$\frac{dID}{dt} = k_{\text{on},I}[D]^n I + k_3 OD - (k_{\text{off},I}^* + k_2 + k_4)ID + k_1 CID, \quad (\text{A.64})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 ID - (k_1 + k_4)CID. \quad (\text{A.65})$$

A.2.12 Model 10

$$\frac{dO}{dt} = \dots - k_{\text{on},O}[D]^n O + k_{\text{off},O} OD, \quad (\text{A.66})$$

$$\frac{dI}{dt} = \dots - k_{\text{on},I}[D]^n I + k_{\text{off},I}^* ID, \quad (\text{A.67})$$

$$\frac{dCD}{dt} = -(k_1 + k_3)CD + k_2 k_{\text{trap}} OD + k_4 CID, \quad (\text{A.68})$$

$$\frac{dOD}{dt} = k_{\text{on},O}[D]^n O + k_1 CD - (k_{\text{off},O} + k_2 k_{\text{trap}} + k_3)OD + k_4 ID, \quad (\text{A.69})$$

$$\frac{dID}{dt} = k_{\text{on},I}[D]^n I + k_3 OD - (k_{\text{off},I}^* + k_2 k_{\text{trap}} + k_4)ID + k_1 CID, \quad (\text{A.70})$$

$$\frac{dCID}{dt} = k_3 CD + k_2 k_{\text{trap}} ID - (k_1 + k_4)CID. \quad (\text{A.71})$$

A.2.13 Model 11

$$\frac{dO}{dt} = \dots - k_{\text{on}}[D]^n O + k_{\text{off}} OD, \quad (\text{A.72})$$

$$\frac{dI}{dt} = \dots - k_{\text{on}}[D]^n I + k_{\text{off}} ID, \quad (\text{A.73})$$

$$\frac{dOD}{dt} = k_{\text{on}}[D]^n O - (k_{\text{off}} + k_{\text{trap}}) OD + k_{\text{untrapped}} OD^*, \quad (\text{A.74})$$

$$\frac{dID}{dt} = k_{\text{on}}[D]^n I - (k_{\text{off}} + k_{\text{trap}}) ID + k_{\text{untrapped}} ID^*, \quad (\text{A.75})$$

$$\frac{dOD^*}{dt} = k_{\text{trap}} OD - k_{\text{untrapped}} OD^*, \quad (\text{A.76})$$

$$\frac{dID^*}{dt} = k_{\text{trap}} ID - k_{\text{untrapped}} ID^*, \quad (\text{A.77})$$

where k_{trap} and $k_{\text{untrapped}}$ are on/off transition rate parameters for trapped states OD^* and ID^* .

A.2.14 Model 12

$$\frac{dO}{dt} = \dots - E_{\text{max}}([D])O + k_{\text{off}} OD, \quad (\text{A.78})$$

$$\frac{dI}{dt} = \dots - E_{\text{max}}([D])I + k_{\text{off}}^* ID, \quad (\text{A.79})$$

$$\frac{dOD}{dt} = E_{\text{max}}([D])O - (k_{\text{off}} + k_{\text{trap}}) OD + k_{\text{trap}} X(V)C^*, \quad (\text{A.80})$$

$$\frac{dID}{dt} = E_{\text{max}}([D])I - (k_{\text{off}}^* + k_{\text{trap}}) ID + k_{\text{trap}} X(V)C^*, \quad (\text{A.81})$$

$$\frac{dC^*}{dt} = k_{\text{trap}}(OD + ID) - 2k_{\text{trap}} X(V)C^*. \quad (\text{A.82})$$

Here $E_{\text{max}}([D])$ is defined as a function of the half maximal effective concentration, EC_{50} , such that

$$E_{\text{max}}([D]) = \hat{k}_{\text{on}} \frac{[D]^n}{[D]^n + EC_{50}^n}. \quad (\text{A.83})$$

Note that we use $\hat{k}_{\text{on}}$ (as opposed to k_{on}) given the units are now ms^{-1} (c.f. $\text{ms}^{-1}\text{nM}^{-n}$). Microscopic reversibility is also enforced by defining

$$k_{\text{off}}^* = k_{\text{off}} \frac{k_3}{k_4}. \quad (\text{A.84})$$

We fix $k_{\text{trap}} = 3.5 \times 10^{-5} \text{ms}^{-1}$ and define $X(V)$ as

$$X(V) = \frac{1}{1 + \exp\left(\frac{V_{1/2,\text{trap}} - V}{6.789}\right)}, \quad (\text{A.85})$$

where $V_{1/2,\text{trap}}$ is a parameter that alters the degree of trapping.

A.2.15 Model 13

The equations for model 13 are identical to those of model 12 with an adjustment to the equation for $E_{\text{max}}([D])$ such that

$$E_{\text{max}}([D]) = k_{\text{on}}[D]^n. \quad (\text{A.86})$$

Appendix B

Supplementary material for Chapter 4

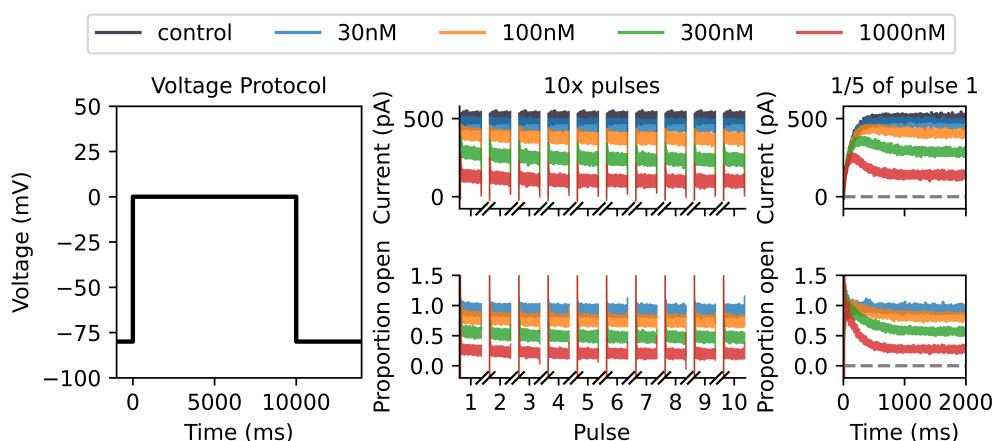


Figure B.1: Synthetic verapamil data under drug-binding model 7 generated under a Milnes protocol with the Lu hERG model. To the left, we plot a single sweep of the Milnes protocol. In the top middle, we plot the control and drug currents for 10 sweeps of the Milnes protocol. We only plot the currents that occur during the 10s pulse at 0 mV for each sweep. In the bottom middle, we plot the corresponding proportion open for each drug concentration, which is calculated by dividing each drug sweep by the control current. On the right, we include a zoomed-in look at the first fifth (2s) of the first pulse for both currents and proportion open.

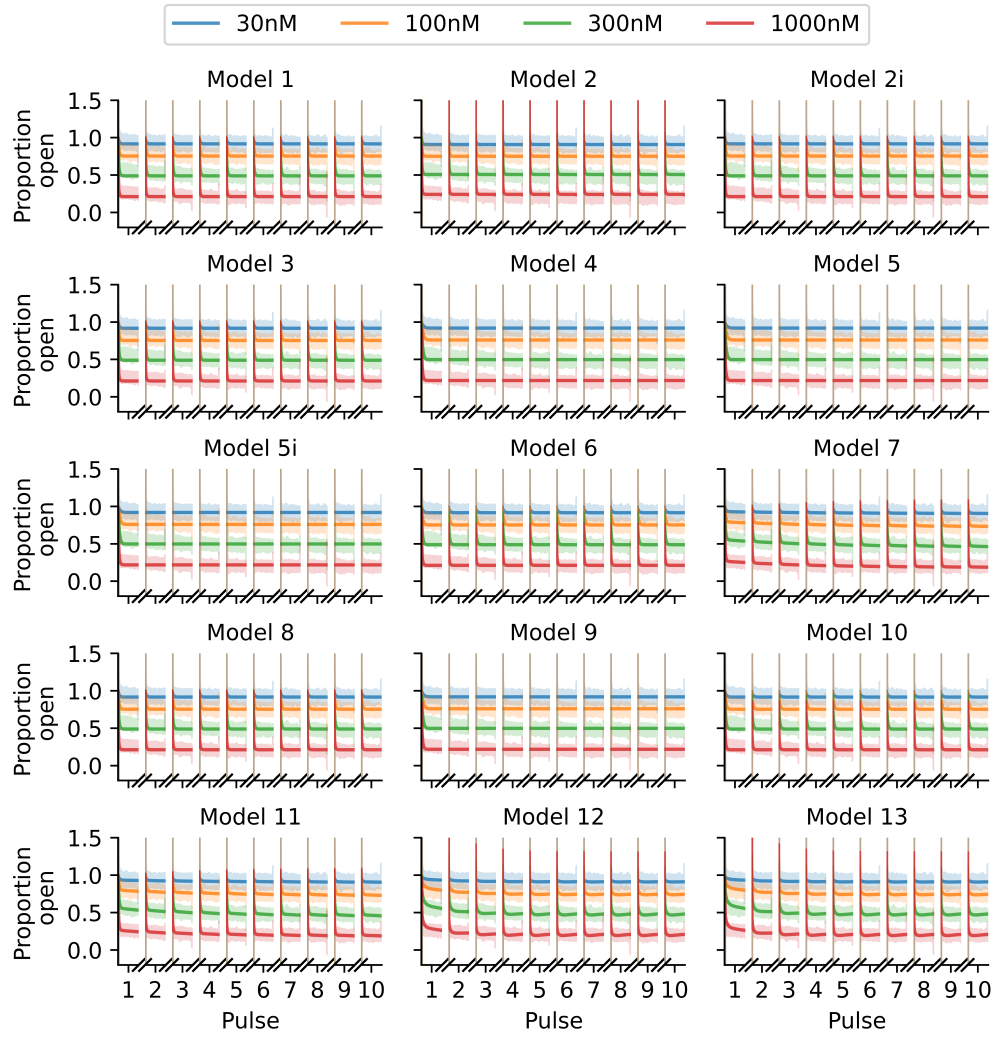


Figure B.2: Fits of the 15 binding models to the Milnes protocol proportion-open synthetic data shown in Figure B.1. The fitted models are shown with solid lines, and we plot the data slightly shaded out behind these fits. While the data is generated using the Lu hERG model, fits are obtained assuming the Beattie 37°C hERG model.

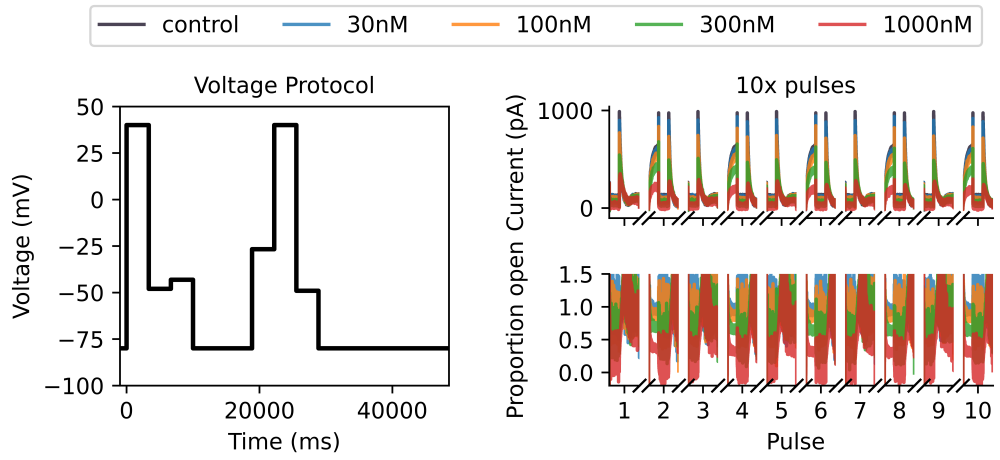


Figure B.3: Synthetic data for drug-binding model 7 generated under an optimised protocol with the Lu hERG model. To the left, we plot our optimised protocol with two 10s pulses, each with three optimised voltage steps. At the top right, we plot the control and drug currents for 5 sweeps of the optimised protocol. We only plot the currents that occur during the two 10s pulses per protocol sweep. At the bottom right, we plot the corresponding proportion open for each drug concentration, which is calculated by dividing each drug sweep by the control current.

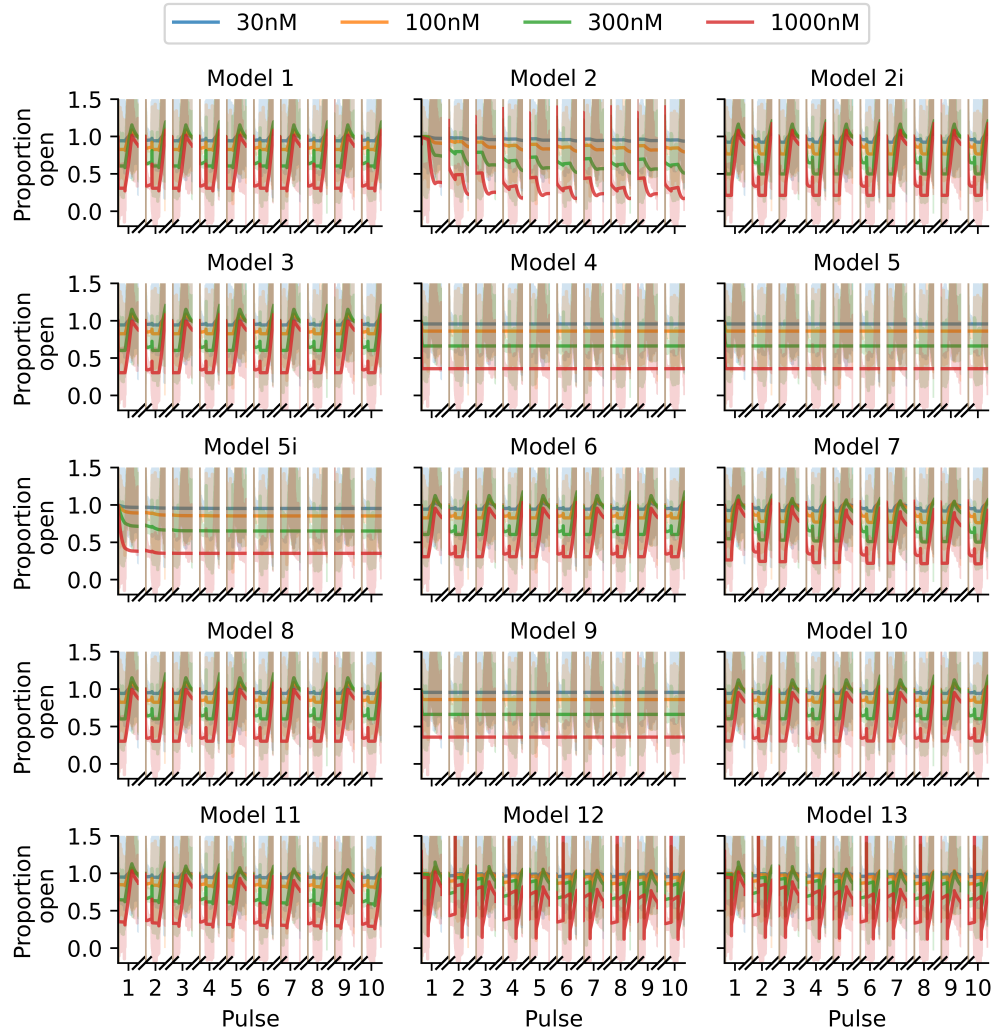


Figure B.4: Fits of the binding models to the optimised protocol proportion open synthetic data shown in Figure B.3, in the same style as Figure B.2. While the data is generated using the Lu hERG model, fits are obtained assuming the Beattie 37°C hERG model.

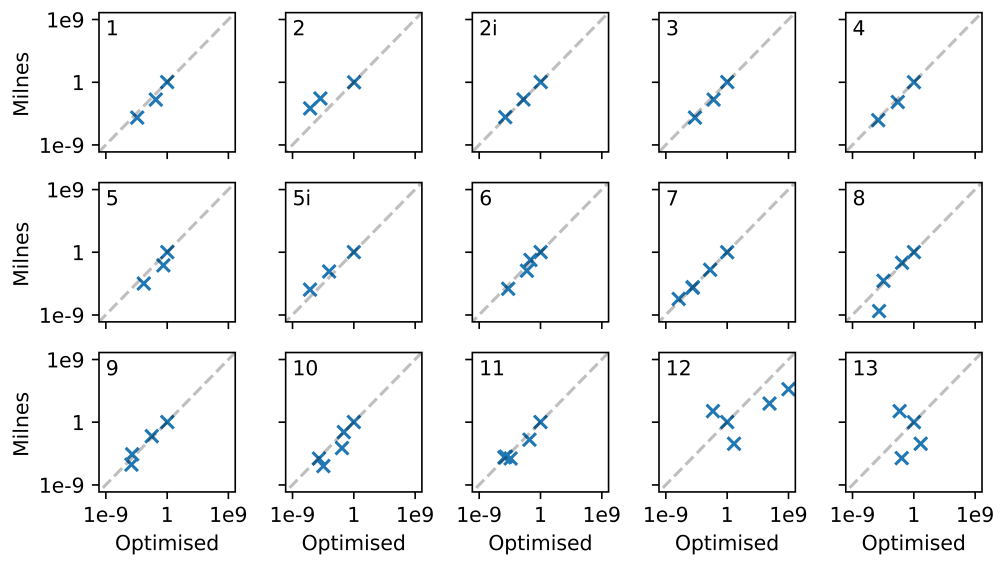


Figure B.5: Comparison plots of model parameters fitted to Milnes data and optimised protocol data. This is for the verapamil model 7 case, with discrepancy between the data-generating hERG model and the hERG model used for model fitting. The blue crosses indicate the values of the fitted model parameters for each binding model.

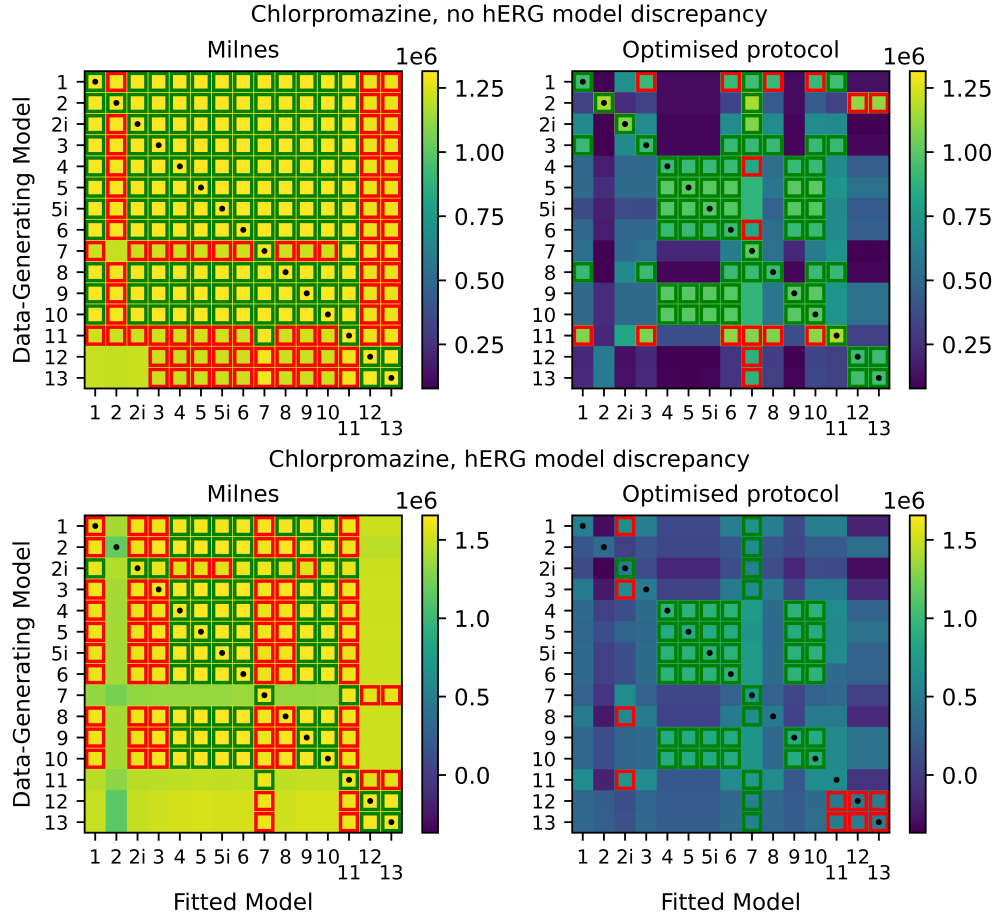


Figure B.6: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic chlorpromazine data. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data.

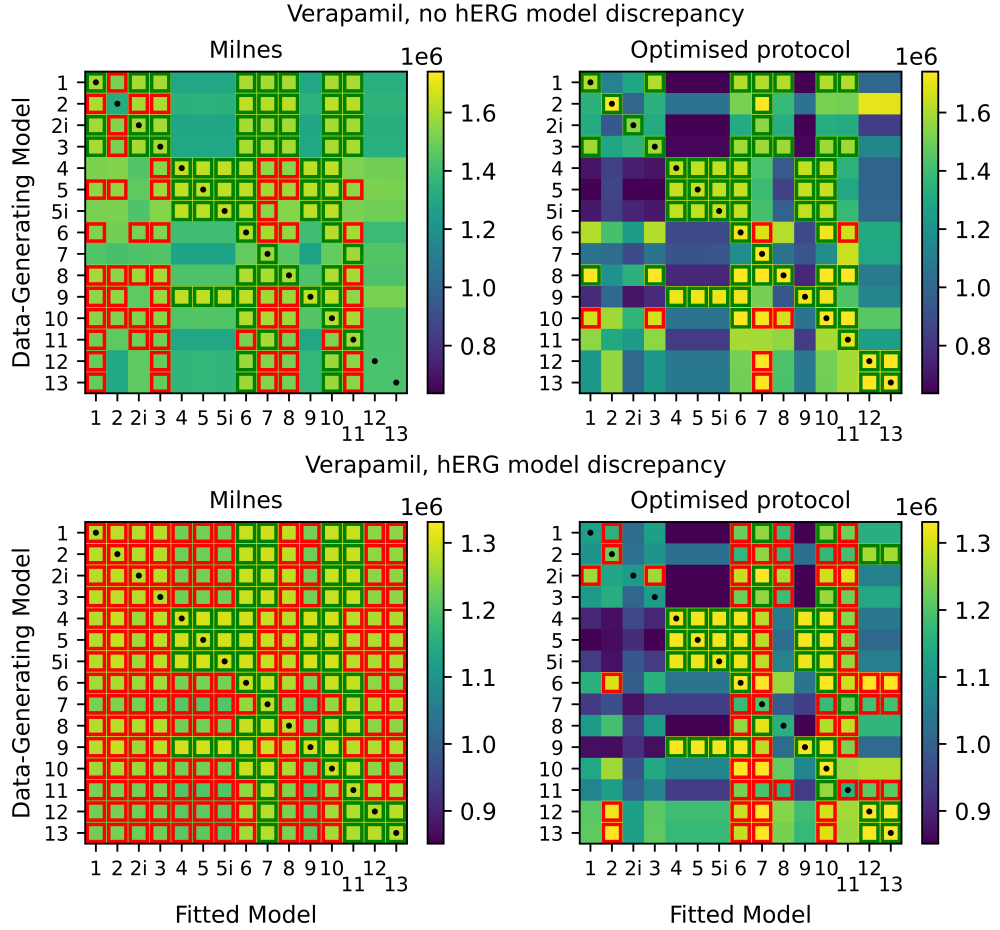


Figure B.7: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic verapamil data using the Lu hERG model. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data; the data-generating hERG model is the Beattie model, while the Lu model is used when performing model fitting.

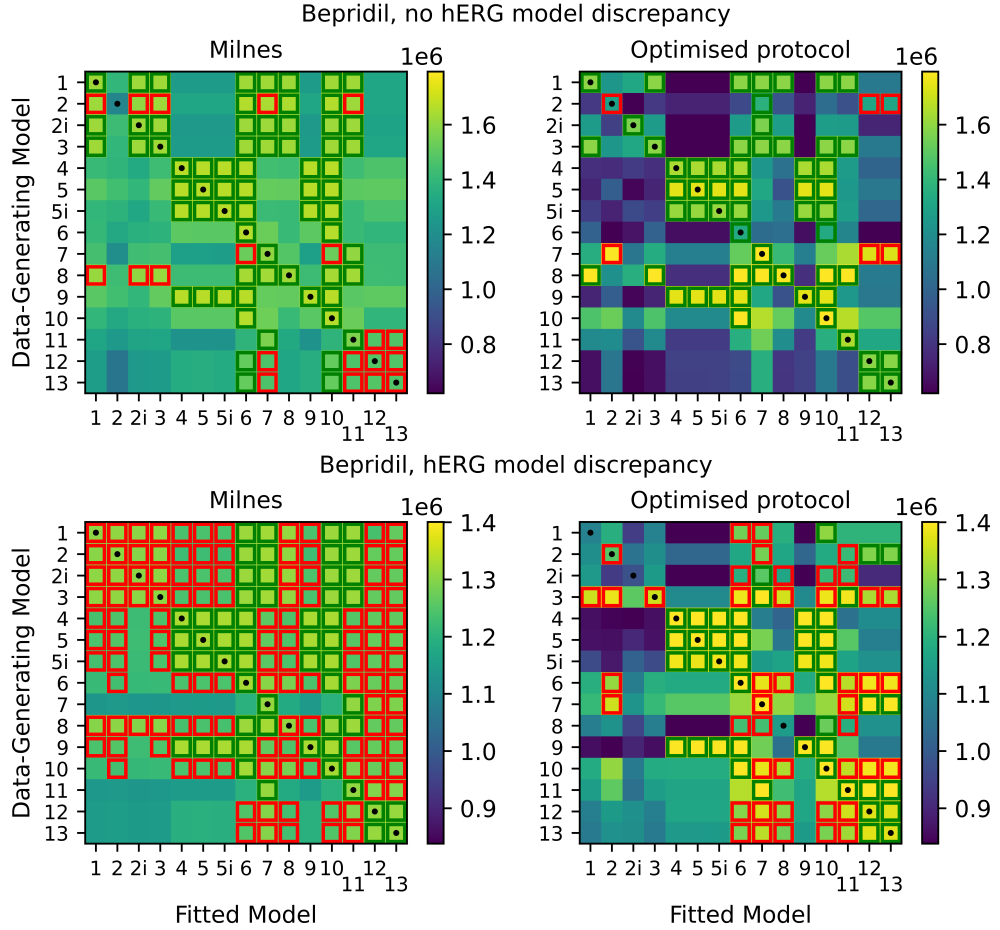


Figure B.8: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic bepridil data using the Lu hERG model. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data; the data-generating hERG model is the Beattie model, while the Lu model is used when performing model fitting.

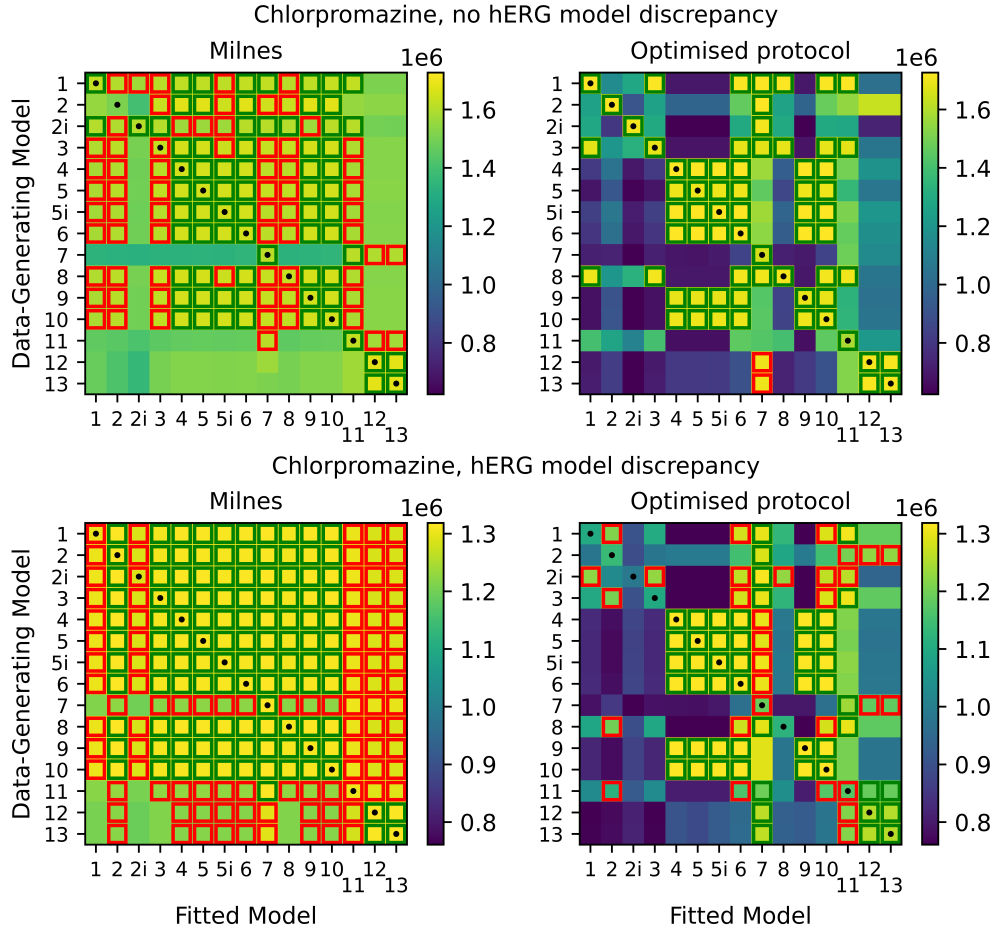


Figure B.9: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic chlorpromazine data using the Lu hERG model. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data; the data-generating hERG model is the Beattie model, while the Lu model is used when performing model fitting.

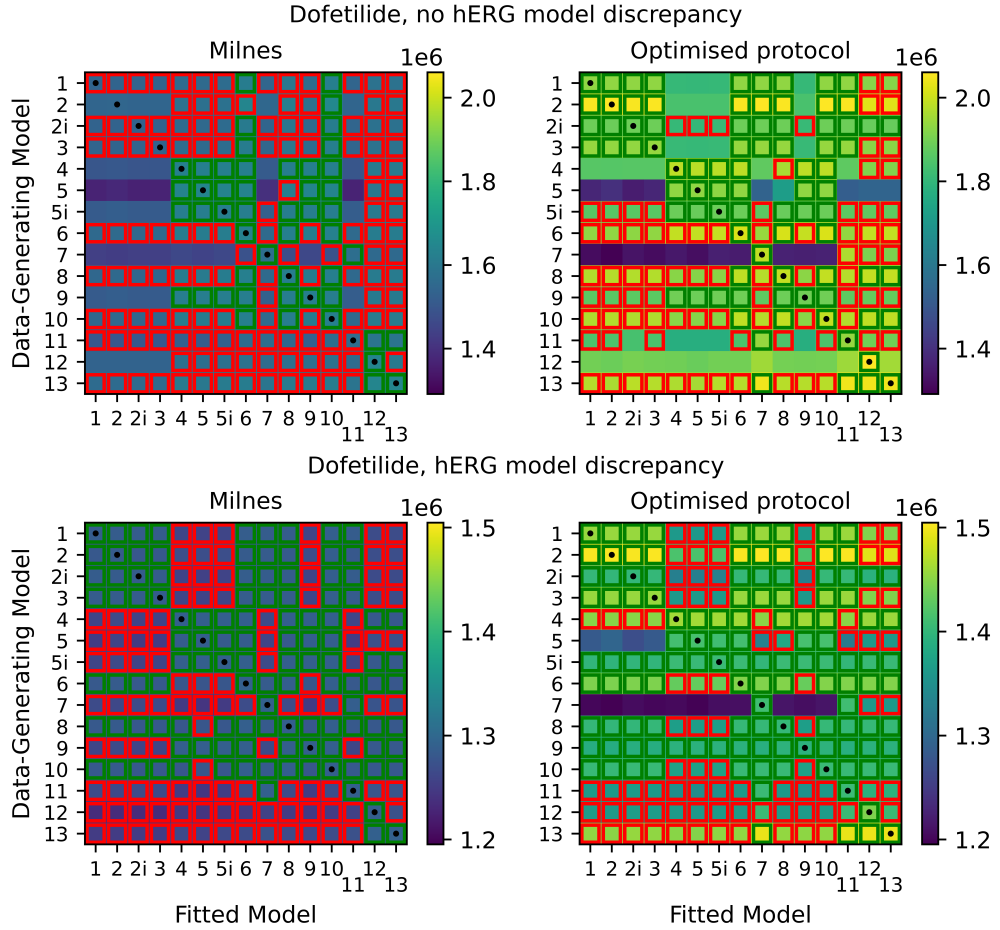


Figure B.10: Top: heatmaps of maximised log-likelihoods for fitted models to Milnes and optimised protocol synthetic dofetilide data using the Lu hERG model. Bottom: equivalent heatmaps of maximised log-likelihoods with discrepancy between the data-generating hERG model and the hERG model used to fit the data; the data-generating hERG model is the Beattie model, while the Lu model is used when performing model fitting.

Appendix C

Supplementary material for Chapter 5

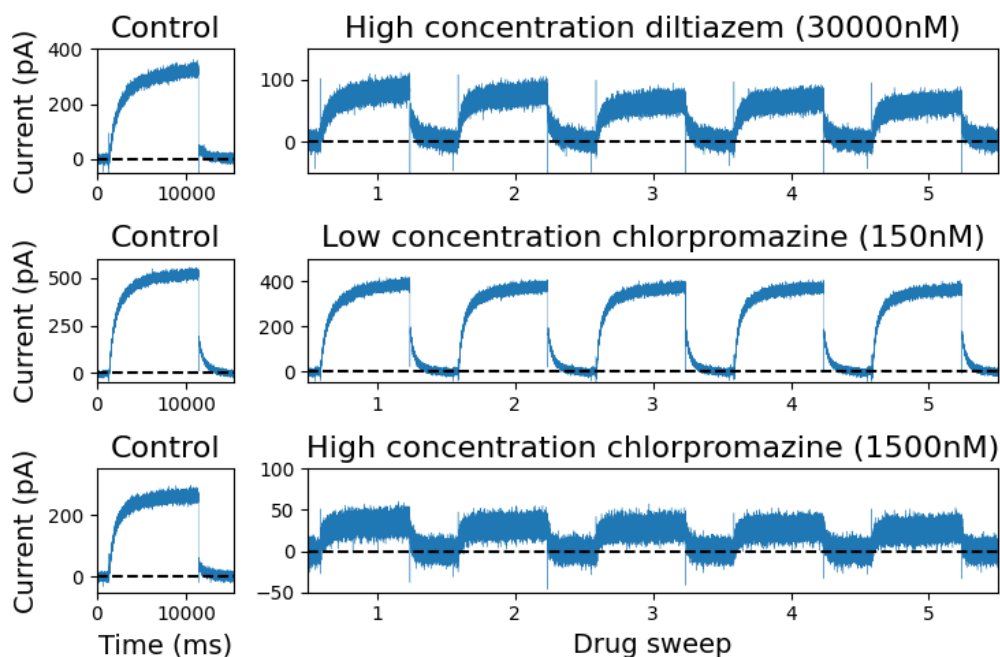


Figure C.1: Examples of post-processed currents in three wells that pass QC. In each row, we plot the final control current under the Milnes protocol (left) and the current under five sweeps of the Milnes protocol after a drug compound of a given concentration has been added (right). Currents have been leak-corrected and full-block subtracted as described in Section 5.2.5. The data was collected with 10 s at resting potential between sweeps — we do not plot the recorded currents during these 10 s intervals here, instead concatenating the 15 s recorded during each Milnes sweep.

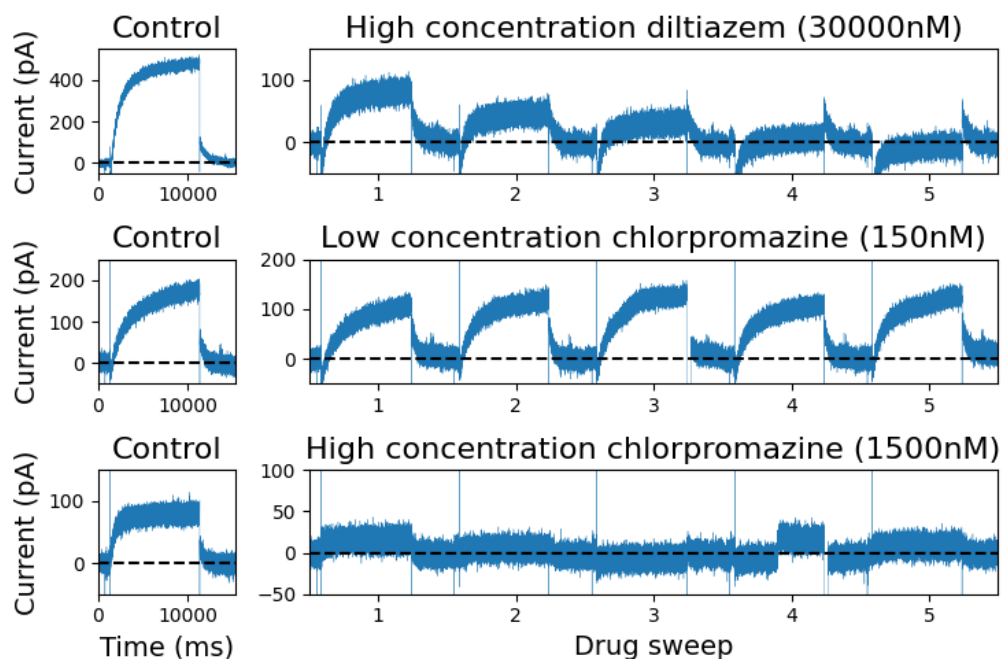


Figure C.2: Examples of post-processed currents in three wells that were manually filtered during QC. Subplots are as described in Figure C.1. These example data illustrate cases where we suspect experimental artefacts are biasing the recordings, but the data have not been filtered during the automated QC procedures detailed in Section 5.2.5. In comparison to the clean and consistent data recorded under identical experimental conditions in Figure C.1, we observe unexpected behaviour here, including negative and inconsistent currents.

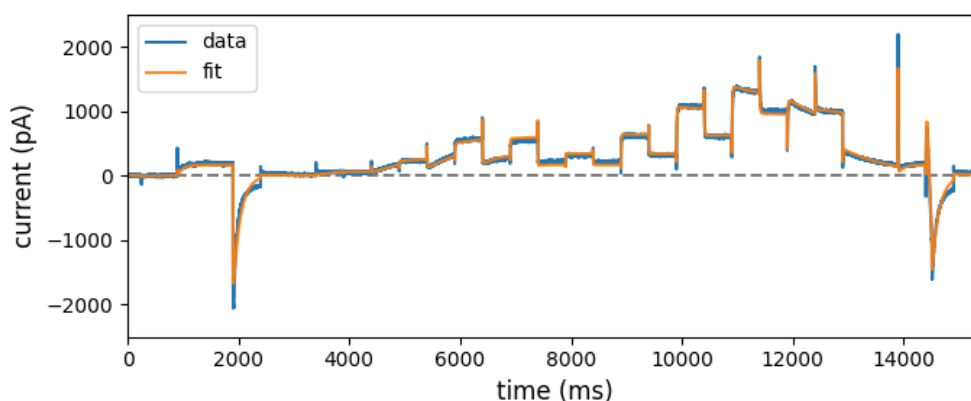


Figure C.3: The Beattie hERG model fit compared to staircase data. In blue, we plot the post-processed current under the staircase protocol in a ‘median’ representative well on the SyncroPatch plate containing slow-binding drugs (Milnes protocol data). Parameters of the Beattie hERG model were fitted to this data, and in orange we plot this fitted model.

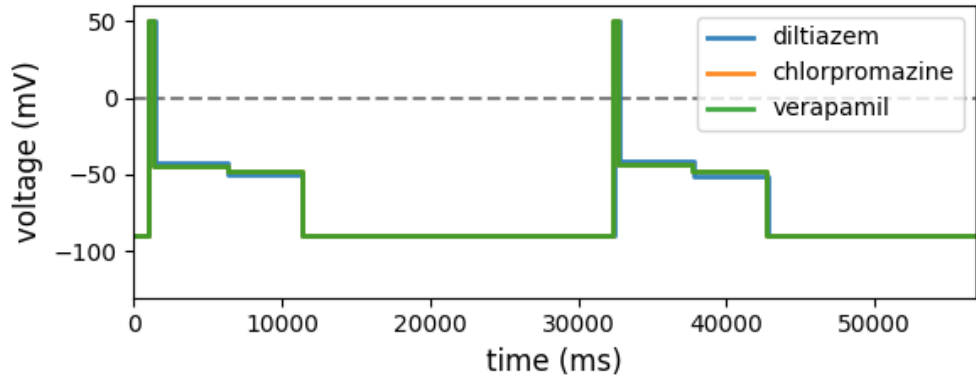


Figure C.4: Optimised protocols for a slow-binding drug, verapamil, and two fast-binding drugs, diltiazem and chlorpromazine. The optimisation methods are as detailed in Section 5.4.

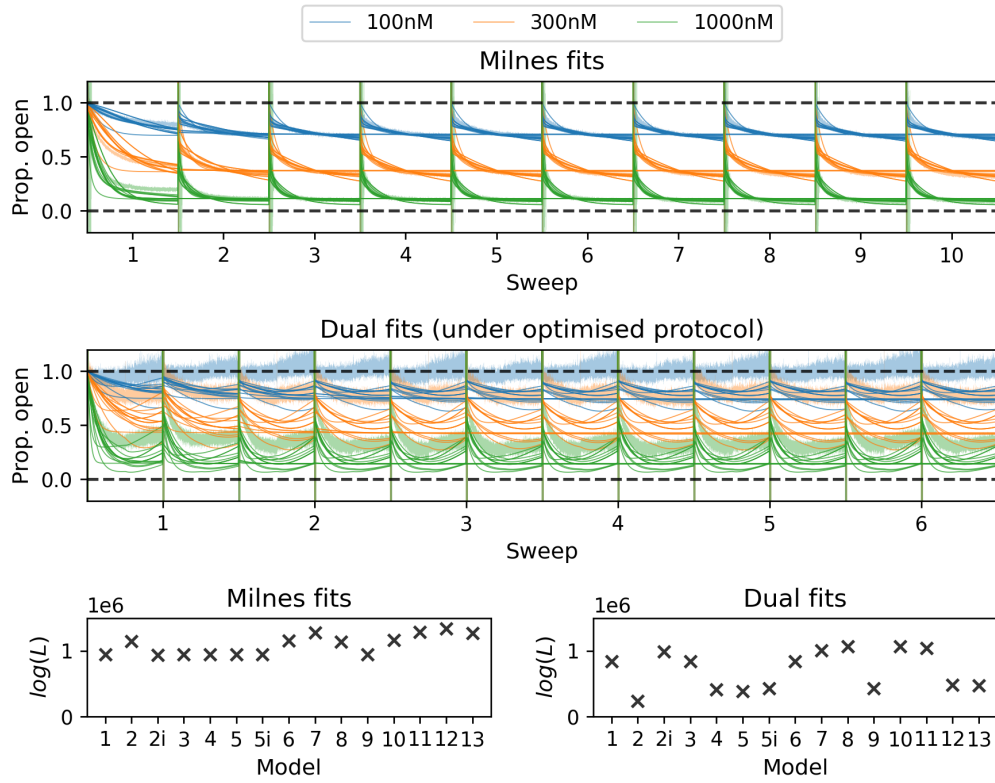


Figure C.5: Fitting drug binding models to verapamil block data collected under the Milnes protocol and an optimised protocol. Top: Milnes data vs. model predictions. Middle: Optimised protocol data vs. dual fit model predictions. Bottom left: Log-likelihoods for each binding model fit to Milnes data. Bottom right: Log-likelihoods for each binding model dual fit.

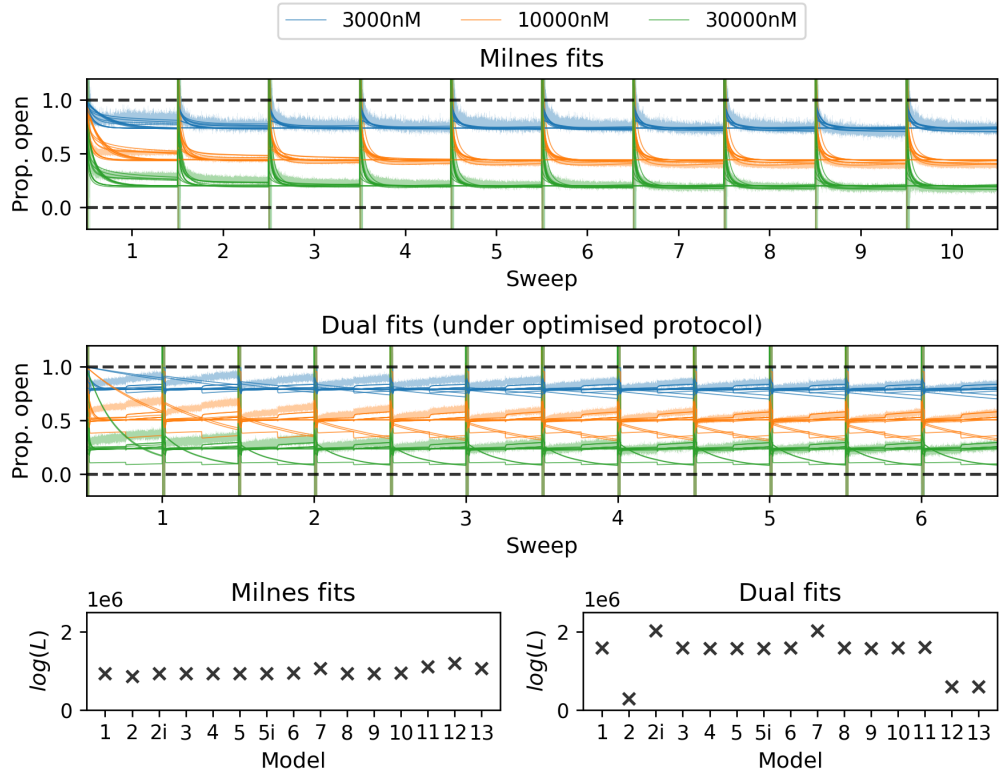


Figure C.6: Fitting drug binding models to diltiazem block data collected under the Milnes protocol and an optimised protocol. Top: Milnes data vs. model predictions. Middle: Optimised protocol data vs. dual fit model predictions. Bottom left: Log-likelihoods for each binding model fit to Milnes data. Bottom right: Log-likelihoods for each binding model dual fit.

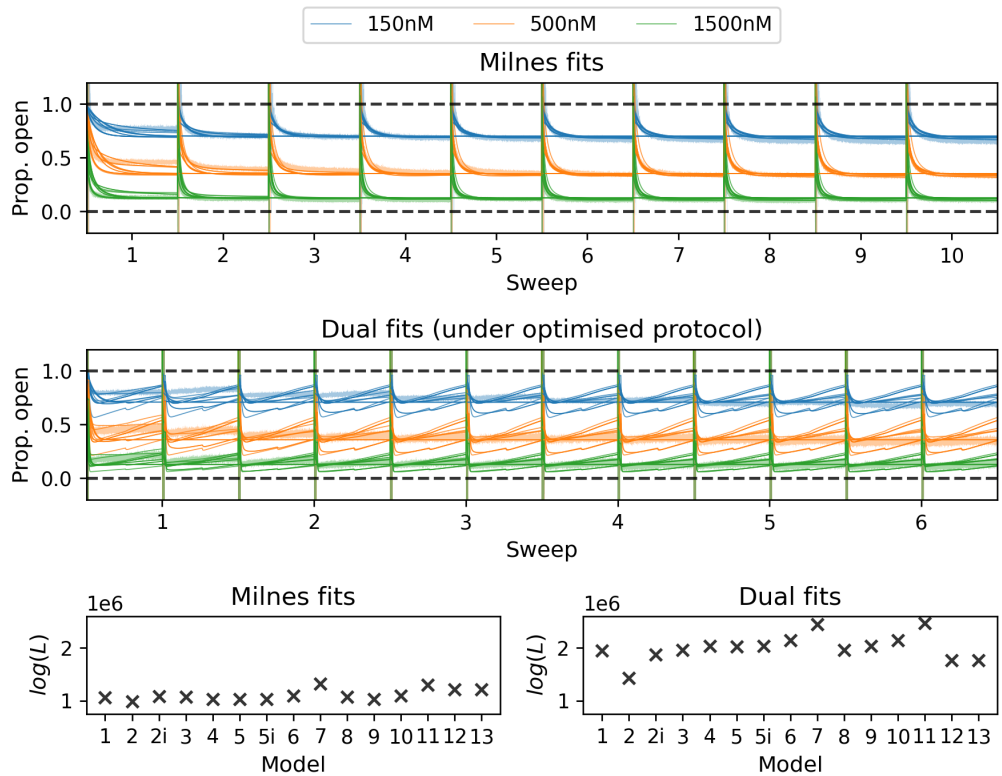


Figure C.7: Fitting drug binding models to chlorpromazine block data collected under the Milnes protocol and an optimised protocol. Top: Milnes data vs. model predictions. Middle: Optimised protocol data vs. dual fit model predictions. Bottom left: Log-likelihoods for each binding model fit to Milnes data. Bottom right: Log-likelihoods for each binding model dual fit.

Appendix D

Supplementary material for Chapter 6

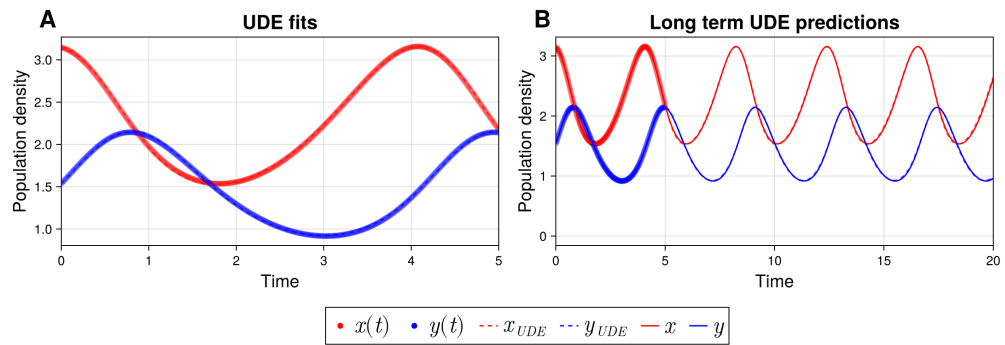


Figure D.1: Fitting a UDE to a fully observed Lotka–Volterra system with no noise. Subplots are equivalently described in Figure 6.1.

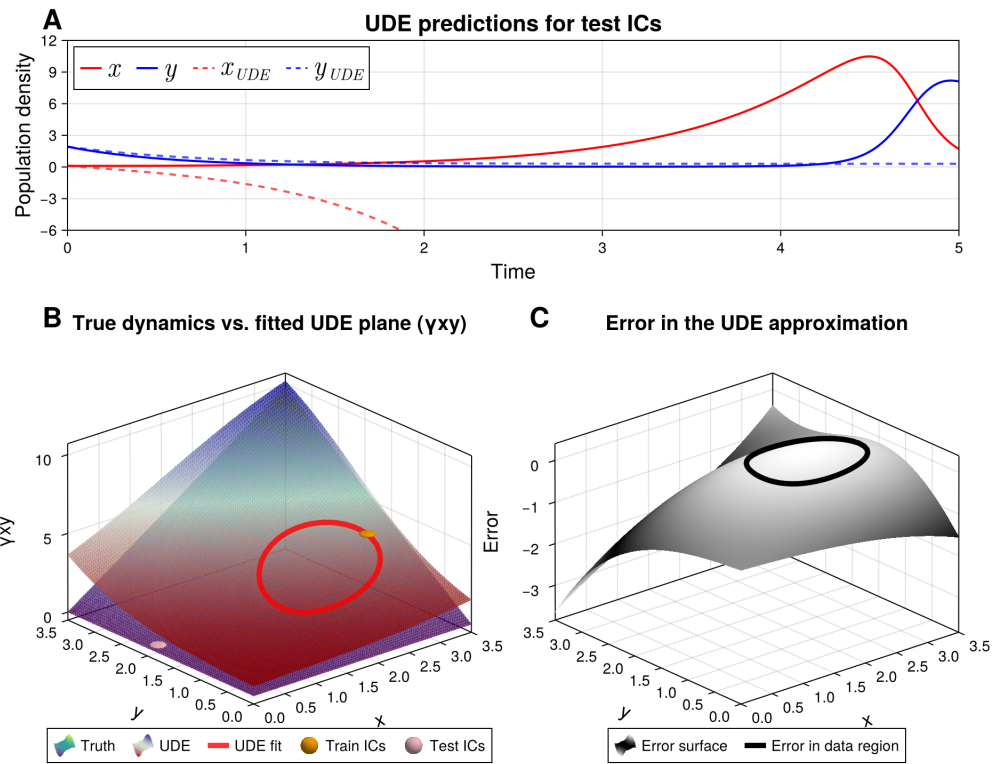


Figure D.2: Extrapolating a UDE fitted to a fully observed Lotka–Volterra system with no noise. Subplots are equivalently described in Figure 6.2.

Bibliography

- Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723.
- Anstett-Collin, F., Denis-Vidal, L., and Millérioux, G. (2020). A priori identifiability: An overview on definitions and approaches. *Annual Reviews in Control*, 50:139–149.
- Atkinson, A. C. and Fedorov, V. V. (1975). The design of experiments for discriminating between two rival models. *Biometrika*, 62(1):57–70.
- Bakris, G. L., Copley, J. B., Vicknair, N., Sadler, R., and Leurgans, S. (1996). Calcium channel blockers versus other antihypertensive therapies on progression of NIDDM associated nephropathy. *Kidney international*, 50(5):1641–1650.
- Ball, F. and Sansom, M. (1989). Ion-channel gating mechanisms: model identification and parameter estimation from single channel recordings. *Proceedings of the Royal Society of London. B. Biological Sciences*, 236(1285):385–416.
- Beattie, K. A., Hill, A. P., Bardenet, R., Cui, Y., Vandenberg, J. I., Gavaghan, D. J., de Boer, T. P., and Mirams, G. R. (2018). Sinusoidal voltage protocols for rapid characterisation of ion channel kinetics. *Biophysical Journal*, 114(3):293a–294a.

- Běhrádek, J. (1930). Temperature coefficients in biology. *Biological Reviews*, 5(1):30–58.
- Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854.
- Bernardo, J. M. and Smith, A. F. (2009). *Bayesian Theory*, volume 405. John Wiley & Sons.
- Bett, G., Zhou, Q., and Rasmusson, R. (2011). Models of hERG Gating. *Biophysical Journal*, 101(3):631–642. Publisher: Elsevier.
- Bickel, P. J. and Doksum, K. A. (2015). *Mathematical statistics: basic ideas and selected topics, volumes I-II package*. Chapman and Hall/CRC.
- Box, G. E. and Draper, N. R. (1987). *Empirical model-building and response surfaces*. John Wiley & Sons.
- Brennan, T., Fink, M., and Rodriguez, B. (2009). Multiscale modelling of drug-induced effects on cardiac electrophysiological activity. *European Journal of Pharmaceutical Sciences*, 36(1):62–77.
- Brezinski, C. and Wuytack, L. (2012). *Numerical analysis: Historical developments in the 20th century*. Elsevier.
- Brunton, S. L., Proctor, J. L., and Kutz, J. N. (2016). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, 113(15):3932–3937.
- Brynjarsdóttir, J. and O’Hagan, A. (2014). Learning about physical parameters: The importance of model discrepancy. *Inverse problems*, 30(11):114007.

- Burnham, K. and Anderson, D. (2007). *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer New York.
- Butcher, J. C. (1996). A history of Runge-Kutta methods. *Applied numerical mathematics*, 20(3):247–260.
- Butler, A., Helliwell, M. V., Zhang, Y., Hancox, J. C., and Dempsey, C. E. (2020). An Update on the Structure of hERG. *Frontiers in Pharmacology*, Volume 10 - 2019.
- Carmeliet, E. and Mubagwa, K. (1998). Antiarrhythmic drugs and cardiac ion channels: mechanisms of action. *Progress in Biophysics and Molecular Biology*, 70(1):1–72.
- Casella, G. and Berger, R. (2002). *Statistical Inference*. Duxbury advanced series in statistics and decision sciences. Thomson Learning.
- Chambers, R. L., Steel, D. G., Wang, S., and Welsh, A. (2012). *Maximum likelihood estimation for sample surveys*. CRC Press.
- Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. (2018). Neural ordinary differential equations. *Advances in neural information processing systems*, 31.
- Chetverikov, D. and Liao, Z. (2016). On cross-validated Lasso. Technical report, cemmap working paper.
- Clerx, M., Beattie, K. A., Gavaghan, D. J., and Mirams, G. R. (2019a). Four ways to fit an ion channel model. *Biophysical Journal*, 117(12):2420–2437.
- Clerx, M., Collins, P., De Lange, E., and Volders, P. G. (2016). Myokit: a simple interface to cardiac cellular electrophysiology. *Progress in biophysics and molecular biology*, 120(1-3):100–114.

- Clerx, M., Robinson, M., Lambert, B., Lei, C. L., Ghosh, S., Mirams, G. R., and Gavaghan, D. J. (2019b). Probabilistic inference on noisy time series (PINTS). *Journal of Open Research Software*, 7:23.
- Colquhoun, D., Dowsland, K. A., Beato, M., and Plested, A. J. (2004). How to impose microscopic reversibility in complex reaction mechanisms. *Biophysical journal*, 86(6):3510–3518.
- Cooper, J., Spiteri, R. J., and Mirams, G. R. (2015). Cellular cardiac electrophysiology modeling with Chaste and CellML. *Frontiers in Physiology*, 5:511.
- Csanády, L. (2006). Statistical evaluation of ion-channel gating models based on distributions of log-likelihood ratios. *Biophysical journal*, 90(10):3523–3545.
- Dhaka, A., Viswanath, V., and Patapoutian, A. (2006). Trp ion channels and temperature sensation. *Annu. Rev. Neurosci.*, 29(1):135–161.
- Di Veroli, G. Y., Davies, M. R., Zhang, H., Abi-Gerges, N., and Boyett, M. R. (2013). High-throughput screening of drug-binding dynamics to hERG improves early drug safety assessment. *American Journal of Physiology-Heart and Circulatory Physiology*, 304(1):H104–H117.
- Díaz-Francés, E. and Rubio, F. J. (2013). On the existence of a normal approximation to the distribution of the ratio of two independent normal random variables. *Statistical Papers*, 54(2):309–323.
- Dutta, S., Chang, K. C., Beattie, K. A., Sheng, J., Tran, P. N., Wu, W. W., Wu, M., Strauss, D. G., Colatsky, T., and Li, Z. (2017). Optimization of an In silico Cardiac Cell Model for Proarrhythmia Risk Assessment. *Frontiers in Physiology*, Volume 8 - 2017.

- Elmes, K., Heywood, A., Huang, Z., and Gavryushkin, A. (2022). A fast lasso-based method for inferring higher-order interactions. *PLoS computational biology*, 18(12):e1010730.
- Escobar, F., Gomis-Tena, J., Saiz, J., and Romero, L. (2022). Automatic modeling of dynamic drug-hERG channel interactions using three voltage protocols and machine learning techniques: A simulation study. *Computer Methods and Programs in Biomedicine*, 226:107148.
- Eyring, H. (1935). The activated complex in chemical reactions. *The Journal of chemical physics*, 3(2):107–115.
- Fermini, B., Hancox, J. C., Abi-Gerges, N., Bridgland-Taylor, M., Chaudhary, K. W., Colatsky, T., Correll, K., Crumb, W., Damiano, B., Erdemli, G., et al. (2016). A new perspective in the field of cardiac safety testing through the comprehensive in vitro proarrhythmia assay paradigm. *Journal of biomolecular screening*, 21(1):1–11.
- Fink, M. and Noble, D. (2009). Markov models for ion channels: versatility versus identifiability and speed. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 367(1896):2161–2179.
- Fink, M., Noble, D., Virag, L., Varro, A., and Giles, W. R. (2008). Contributions of hERG K⁺ current to repolarization of the human ventricular action potential. *Progress in biophysics and molecular biology*, 96(1-3):357–376.
- Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical transactions of the Royal Society of London. Series A, containing papers of a mathematical or physical character*, 222(594-604):309–368.

- Gintant, G., Sager, P. T., and Stockbridge, N. (2016). Evolution of strategies to improve preclinical cardiac safety testing. *Nature reviews Drug discovery*, 15(7):457–471.
- Gomis-Tena, J., Brown, B. M., Cano, J., Trenor, B., Yang, P.-C., Saiz, J., Clancy, C. E., and Romero, L. (2020). When does the IC50 accurately assess the blocking potency of a drug? *Journal of chemical information and modeling*, 60(3):1779–1790.
- Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*, volume 1. MIT press Cambridge.
- Goutelle, S., Maurin, M., Rougier, F., Barbaut, X., Bourguignon, L., Ducher, M., and Maire, P. (2008). The Hill equation: a review of its capabilities in pharmacological modelling. *Fundamental & clinical pharmacology*, 22(6):633–648.
- Grandi, E., Pasqualini, F. S., Puglisi, J. L., and Bers, D. M. (2009). A novel computational model of the human ventricular action potential and Ca transient. *Biophysical journal*, 96(3):664a–665a.
- Grant, A. O. (2009). Cardiac ion channels. *Circulation: Arrhythmia and Electrophysiology*, 2(2):185–194.
- Gupta, S., Lee, R. E., and Faeder, J. R. (2020). Parallel Tempering with Lasso for model reduction in systems biology. *PLoS computational biology*, 16(3):e1007669.
- Halliwel, J. V., Plant, T. D., Robbins, J., and Standen, N. B. (1994). Voltage clamp techniques. In *Microelectrode Techniques. The Plymouth Workshop Handbook*, pages 17–35. The Company of Biologists, Ltd. Cambridge.

- Hancox, J. C., McPate, M. J., El Harchi, A., and hong Zhang, Y. (2008). The hERG potassium channel and hERG screening for drug-induced torsades de pointes. *Pharmacology & Therapeutics*, 119(2):118–132.
- Hansen, N. (2006). The CMA evolution strategy: A comparing review. In Lozano, J. A., Larrañaga, P., Inza, I., and Bengoetxea, E., editors, *Towards a New Evolutionary Computation: Advances in the Estimation of Distribution Algorithms*, pages 75–102. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer.
- Heist, E. K. and Ruskin, J. N. (2010). Drug-induced arrhythmia. *Circulation*, 122(14):1426–1435.
- Hendry, D. F. and Nielsen, B. (2007). *Econometric modeling: a likelihood approach*. Princeton University Press.
- Higdon, D. M., Anderson, M. C., Habib, S., Klein, R., Berliner, M., Covey, C., Ghattas, O., Graziani, C., Seager, M., Sefcik, J., et al. (2009). Uncertainty quantification and error analysis. Technical report, Los Alamos National Laboratory (LANL).
- Hill, A. P., Perry, M. D., Abi-Gerges, N., Couderc, J.-P., Fermini, B., Hancox, J. C., Knollmann, B. C., Mirams, G. R., Skinner, J., Zareba, W., et al. (2016). Computational cardiology and risk stratification for sudden cardiac death: one of the grand challenges for cardiology in the 21st century. *The Journal of Physiology*, 594(23):6893–6908.
- Hill, A. V. (1910). The possible effects of the aggregation of the molecules

- of hemoglobin on its dissociation curves. *The Journal of Physiology*, 40:iv–vii.
- Hille, B. (2001). *Ion channels of excitable membranes / Bertil Hille*. Sinauer Associates, Sunderland, Mass, 3rd ed. edition.
- Hindmarsh, A. C., Brown, P. N., Grant, K. E., Lee, S. L., Serban, R., Shumaker, D. E., and Woodward, C. S. (2005). Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*, 31(3):363–396.
- Hirsh, S. M., Barajas-Solano, D. A., and Kutz, J. N. (2022). Sparsifying priors for Bayesian uncertainty quantification in model discovery. *Royal Society open science*, 9(2):211823.
- Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4):500–544.
- Hondeghem, L. (1987). Antiarrhythmic agents: modulated receptor applications. *Circulation*, 75(3):514–520.
- Howell, S. J., Stivland, T., Stein, K., Ellenbogen, K. A., and Tereshchenko, L. G. (2021). Using machine-learning for prediction of the response to cardiac resynchronization therapy: the SMART-AV study. *Clinical Electrophysiology*, 7(12):1505–1515.
- Huikuri, H. V., Castellanos, A., and Myerburg, R. J. (2001). Sudden death due to cardiac arrhythmias. *New England Journal of Medicine*, 345(20):1473–1482.
- ICH E14/S7B Implementation Working Group (2022). Clinical and Non-clinical Evaluation of QT/QTc Interval Prolongation and Proarrhythmic Potential.

- Jarvis, G. E. and Thompson, A. J. (2013). A golden approach to ion channel inhibition. *Trends in Pharmacological Sciences*, 34(9):481–488.
- Jia, S., Mou, H., Wu, Y., Lin, W., Zeng, Y., Chen, Y., Chen, Y., Zhang, Q., Wang, W., Feng, C., et al. (2022). A simple logistic regression model for Predicting the Likelihood of recurrence of Atrial Fibrillation within 1 year after initial radio-frequency catheter ablation therapy. *Frontiers in Cardiovascular Medicine*, 8:819341.
- Johnstone, R. H., Chang, E. T., Bardenet, R., De Boer, T. P., Gavaghan, D. J., Pathmanathan, P., Clayton, R. H., and Mirams, G. R. (2016). Uncertainty and variability in models of the cardiac action potential: Can we build trustworthy models? *Journal of molecular and cellular cardiology*, 96:49–62.
- Kalyaanamoorthy, S. and Barakat, K. H. (2018). Development of Safe Drugs: The hERG Challenge. *Medicinal Research Reviews*, 38(2):525–555.
- Kamerud, D. (1978). Solution to problem 6104: the random variable X/Y , X, Y normal. *American Mathematical Monthly*, 85(3):206–207.
- Kamiya, K., Niwa, R., Mitcheson, J. S., and Sanguinetti, M. C. (2006). Molecular determinants of hERG channel block. *Molecular Pharmacology*, 69(5):1709–1716.
- Karlik, B., Olgac, A. V., et al. (2011). Performance analysis of various activation functions in generalized MLP architectures of neural networks. *International journal of artificial intelligence and expert systems*, 1(4):111–122.
- Kennedy, M. C. and O’Hagan, A. (2001). Bayesian calibration of computer

- models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(3):425–464.
- Khurshid, S., Choi, S. H., Weng, L.-C., Wang, E. Y., Trinquart, L., Benjamin, E. J., Ellinor, P. T., and Lubitz, S. A. (2018). Frequency of cardiac rhythm abnormalities in a half million adults. *Circulation: Arrhythmia and Electrophysiology*, 11(7):e006273.
- Kidger, P., Morrill, J., Foster, J., and Lyons, T. (2020). Neural controlled differential equations for irregular time series. *Advances in neural information processing systems*, 33:6696–6707.
- Kiefer, J. (1959). Optimum Experimental Designs. *Journal of the Royal Statistical Society. Series B (Methodological)*, 21(2):272–319.
- Kiehn, J., Lacerda, A. E., Wible, B., and Brown, A. M. (1996). Molecular physiology and pharmacology of hERG. *Circulation*, 94(10):2572–2579.
- King, G. (1989). *Unifying Political Methodology: The Likelihood Theory of Statistical Inference*. Cambridge University Press.
- Kingma, J., Simard, C., and Drolet, B. (2023). Overview of cardiac arrhythmias and treatment strategies. *Pharmaceuticals*, 16(6):844.
- Koch, K. (1999). *Parameter Estimation and Hypothesis Testing in Linear Models*. Springer.
- Kornreich, B. G. (2007). The patch clamp technique: principles and technical considerations. *Journal of Veterinary Cardiology*, 9(1):25–37.
- Kozek, K. A., Glazer, A. M., Ng, C.-A., Blackwell, D., Egly, C. L., Vanags, L. R., Blair, M., Mitchell, D., Matreyek, K. A., Fowler, D. M., et al. (2020). High-throughput discovery of trafficking-deficient variants in

- the cardiac potassium channel KV11. 1. *Heart rhythm*, 17(12):2180–2189.
- Kreutz, C., Raue, A., Kaschek, D., and Timmer, J. (2013). Profile likelihood in systems biology. *The FEBS journal*, 280(11):2564–2571.
- Kuethé, D. O., Caprihan, A., Gach, H. M., Lowe, I. J., and Fukushima, E. (2000). Imaging obstructed ventilation with NMR using inert fluorinated gases. *Journal of applied physiology*, 88(6):2279–2286.
- Kutalik, Z., Cho, K.-H., and Wolkenhauer, O. (2004). Optimal sampling time selection for parameter estimation in dynamic pathway modeling. *Biosystems*, 75(1-3):43–55.
- Kuwahara, B. and Bauch, C. T. (2024). Predicting Covid-19 pandemic waves with biologically and behaviorally informed universal differential equations. *Heliyon*, 10(4).
- Laverty, H., Benson, C., Cartwright, E., Cross, M., Garland, C., Hammond, T., Holloway, C., McMahon, N., Milligan, J., Park, B., et al. (2011). How can we improve our understanding of cardiovascular safety liabilities to develop safer medicines? *British journal of pharmacology*, 163(4):675–693.
- Lehmann, E. L. and Casella, G. (1998). *Theory of point estimation*. Springer.
- Lei, C. L., Clark, A. P., Clerx, M., Wei, S., Bloothoof, M., de Boer, T. P., Christini, D. J., Krogh-Madsen, T., and Mirams, G. R. (2025). Resolving artifacts in voltage-clamp experiments with computational modeling: An application to fast sodium current recordings. *Advanced Science*, 12(30):e00691.

- Lei, C. L., Clerx, M., Beattie, K. A., Melgari, D., Hancox, J. C., Gavaghan, D. J., Polonchuk, L., Wang, K., and Mirams, G. R. (2019a). Rapid characterization of hERG channel kinetics II: Temperature dependence. *Biophysical Journal*, 117(12):2455–2470.
- Lei, C. L., Clerx, M., Gavaghan, D. J., and Mirams, G. R. (2023). Model-driven optimal experimental design for calibrating cardiac electrophysiology models. *Computer Methods and Programs in Biomedicine*, 240:107690.
- Lei, C. L., Clerx, M., Gavaghan, D. J., Polonchuk, L., Mirams, G. R., and Wang, K. (2019b). Rapid characterization of hERG channel kinetics I: Using an automated high-throughput system. *Biophysical Journal*, 117(12):2438–2454.
- Lei, C. L., Clerx, M., Whittaker, D. G., Gavaghan, D. J., de Boer, T. P., and Mirams, G. R. (2020). Accounting for variability in ion current recordings using a mathematical model of artefacts in voltage-clamp experiments. *Philosophical Transactions of the Royal Society A*, 378(2173):20190348.
- Lei, C. L., Fabbri, A., Whittaker, D. G., Clerx, M., Windley, M. J., Hill, A. P., Mirams, G. R., and de Boer, T. P. (2021). A nonlinear and time-dependent leak current in the presence of calcium fluoride patch-clamp seal enhancer. *Wellcome Open Research*, 5:152.
- Lei, C. L. and Mirams, G. R. (2021). Neural network differential equations for ion channel modelling. *Frontiers in Physiology*, 12:708944.
- Lei, C. L., Whittaker, D. G., and Mirams, G. R. (2024). The impact of uncertainty in hERG binding mechanism on in silico predictions of drug-induced proarrhythmic risk. *British Journal of Pharmacology*, 181(7):987–1004.

- Li, Z., Dutta, S., Sheng, J., Tran, P. N., Wu, W., Chang, K., Mdluli, T., Strauss, D. G., and Colatsky, T. (2017). Improving the In Silico Assessment of Proarrhythmia Risk by Combining hERG (Human Ether-à-go-go-Related Gene) Channel–Drug Binding Kinetics and Multichannel Pharmacology. *Circulation: Arrhythmia and Electrophysiology*, 10(2):e004628. Publisher: American Heart Association.
- Li, Z., Ridder, B. J., Han, X., Wu, W. W., Sheng, J., Tran, P. N., Wu, M., Randolph, A., Johnstone, R. H., Mirams, G. R., Kuryshv, Y., Kramer, J., Wu, C., Jr., W. J. C., and Strauss, D. G. (2019). Assessment of an in silico mechanistic model for proarrhythmia risk prediction under the CiPA initiative. *Clinical Pharmacology & Therapeutics*, 105(2):466–475.
- Lindley, D. V. (1956). On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005.
- Lipton, S. A. (2004). Failures and successes of NMDA receptor antagonists: molecular basis for the use of open-channel blockers like memantine in the treatment of acute and chronic neurologic insults. *NeuroRx*, 1(1):101–110.
- Liu, Y., Maini, P. K., and Baker, R. E. (2025). Optimal experiment design for practical parameter identifiability and model discrimination. *arXiv preprint arXiv:2506.11311*.
- Loman, T. E. and Baker, R. E. (2025). Functional and parametric identifiability for universal differential equations applied to chemical reaction networks. *arXiv preprint arXiv:2510.14140*.
- Lu, Y., Mahaut-Smith, M., Varghese, A., Huang, C.-H., Kemp, P., and Vandenberg, J. (2001). Effects of premature stimulation on hERG K⁺ channels. *The Journal of Physiology*, 537(3):843–851.

- Lukacs, P., Pesti, K., Földi, M. C., Zboray, K., Toth, A. V., Papp, G., and Mike, A. (2021). An advanced automated patch clamp protocol design to investigate drug—ion channel binding dynamics. *Frontiers in Pharmacology*, 12:738260.
- Maclaren, O. J. and Nicholson, R. (2019). What can be estimated? identifiability, estimability, causal inference and ill-posed inverse problems. *arXiv preprint arXiv:1904.02826*.
- Mangan, N. M., Brunton, S. L., Proctor, J. L., and Kutz, J. N. (2017). Inferring biological networks by sparse identification of nonlinear dynamics. *IEEE Transactions on Molecular, Biological, and Multi-Scale Communications*, 2(1):52–63.
- Marty, A. and Neher, E. (1995). Tight-seal whole-cell recording. In *Single-Channel Recording*, pages 31–52. Springer.
- Michalik, C., Stuckert, M., and Marquardt, W. (2010). Optimal experimental design for discriminating numerous model candidates: the AWDC criterion. *Industrial & engineering chemistry research*, 49(2):913–919.
- Milnes, J. T., Witchel, H. J., Leaney, J. L., Leishman, D. J., and Hancox, J. C. (2010). Investigating dynamic protocol-dependence of hERG potassium channel inhibition at 37°C: Cisapride versus dofetilide. *Journal of Pharmacological and Toxicological Methods*, 61(2):178–191.
- Mirams, G. R. (2023). Computational cardiac safety testing. In Hock, F., Gralinski, M., and Pugsley, M., editors, *Drug Discovery and Evaluation: Safety and Pharmacokinetic Assays*, pages 1–33. Springer International Publishing.
- Mirams, G. R., Clerx, M., Whittaker, D. G., and Lei, C. L. (2024). Optimal experimental designs for characterising ion channel gating by filling the

- phase-voltage space of model dynamics. *Mathematics in Medical and Life Sciences*, 1(1):2375494.
- Mirams, G. R., Davies, M. R., Cui, Y., Kohl, P., and Noble, D. (2012). Application of cardiac electrophysiology simulations to pro-arrhythmic safety testing. *British Journal of Pharmacology*, 167(5):932–945.
- Mitcheson, J. S. (2008). hERG Potassium Channels and the Structural Basis of Drug-Induced Arrhythmias. *Chemical Research in Toxicology*, 21(5):1005–1010. Publisher: American Chemical Society.
- Mitra, E. D. and Hlavacek, W. S. (2019). Parameter estimation and uncertainty quantification for systems biology models. *Current opinion in systems biology*, 18:9–18.
- Moreno, J. D., Lewis, T. J., and Clancy, C. E. (2016). Parameterization for in-silico modeling of ion channel interactions with drugs. *PloS one*, 11(3):e0150761.
- Myung, I. J. (2003). Tutorial on maximum likelihood estimation. *Journal of mathematical Psychology*, 47(1):90–100.
- Myung, J. I. and Pitt, M. A. (2009). Optimal experimental design for model discrimination. *Psychological review*, 116(3):499.
- Neher, E. and Sakmann, B. (1976). Single-channel currents recorded from membrane of denervated frog muscle fibres. *Nature*, 260(5554):799–802.
- Noble, D. (1962). A modification of the Hodgkin—Huxley equations applicable to Purkinje fibre action and pacemaker potentials. *The Journal of Physiology*, 160(2):317–352.

- North, J. S., Wikle, C. K., and Schliep, E. M. (2023). A review of data-driven discovery for dynamic systems. *International Statistical Review*, 91(3):464–492.
- Panoulas, V. F., Toms, T. E., Douglas, K. M., Sandoo, A., Metsios, G. S., Stavropoulos-Kalinoglou, A., and Kitas, G. D. (2014). Prolonged QTc interval predicts all-cause mortality in patients with rheumatoid arthritis: an association driven by high inflammatory burden. *Rheumatology*, 53(1):131–137.
- Papadimitriou, C., Haralampidis, Y., and Sobczyk, K. (2005). Optimal experimental design in stochastic structural dynamics. *Probabilistic engineering mechanics*, 20(1):67–78.
- Patten-Elliott, F., Lei, C. L., Preston, S. P., Wilkinson, R. D., and Mirams, G. R. (2025). Optimizing experimental designs for model selection of ion channel drug-binding mechanisms. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2292):20240227.
- Perry, M., Sanguinetti, M., and Mitcheson, J. (2010). Symposium review: revealing the structural basis of action of hERG potassium channel activators and blockers. *The Journal of Physiology*, 588(17):3157–3167.
- Philipps, M., Schmid, N., and Hasenauer, J. (2025). Current state and open problems in universal differential equations for systems biology. *npj Systems Biology and Applications*, 11(1):101.
- Pollock, J. D. and Makaryus, A. N. (2022). Physiology, cardiac cycle. In *StatPearls [Internet]*. StatPearls Publishing.
- Priest, B. T. and McDermott, J. S. (2015). Cardiac ion channels. *Channels*, 9(6):352–359.

- Rackauckas, C., Ma, Y., Martensen, J., Warner, C., Zubov, K., Supekar, R., Skinner, D., Ramadhan, A., and Edelman, A. (2020). Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*.
- Raue, A., Kreutz, C., Maiwald, T., Bachmann, J., Schilling, M., Klingmüller, U., and Timmer, J. (2009). Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929.
- Redfern, W., Carlsson, L., Davis, A., Lynch, W., MacKenzie, I., Palethorpe, S., Siegl, P., Strang, I., Sullivan, A., Wallis, R., et al. (2003). Relationships between preclinical cardiac electrophysiology, clinical QT interval prolongation and torsade de pointes for a broad range of drugs: evidence for a provisional safety margin in drug development. *Cardiovascular Research*, 58(1):32–45.
- Roden, D. M. (2004). Drug-induced prolongation of the QT interval. *New England Journal of Medicine*, 350(10):1013–1022.
- Rossi, R. (2018). *Mathematical Statistics: An Introduction to Likelihood Based Inference*. Wiley.
- Rudy, Y. and Silva, J. R. (2006). Computational biology in the study of cardiac ion channels and cell electrophysiology. *Quarterly Reviews of Biophysics*, 39(1):57–116. Place: Cambridge Publisher: Cambridge University Press.
- Sager, P. T., Gintant, G., Turner, J. R., Pettit, S., and Stockbridge, N. (2014). Rechanneling the cardiac proarrhythmia safety paradigm: a meeting report from the cardiac safety research consortium. *American heart journal*, 167(3):292–300.

- Sakmann, B. and Neher, E. (1984). Patch clamp techniques for studying ionic channels in excitable membranes. *Annual review of physiology*, 46(1):455–472.
- Sánchez-Chapula, J. A., Ferrer, T., Navarro-Polanco, R. A., and Sanguinetti, M. C. (2003). Voltage-dependent profile of human ether-a-go-go-related gene channel block is influenced by a single residue in the S6 transmembrane domain. *Molecular pharmacology*, 63(5):1051–1058.
- Sanguinetti, M. C., Jiang, C., Curran, M. E., and Keating, M. T. (1995). A mechanistic link between an inherited and an acquired cardiac arrhythmia: hERG encodes the IKr potassium channel. *Cell*, 81(2):299–307.
- Sanguinetti, M. C. and Jurkiewicz, N. K. (1990). Two components of cardiac delayed rectifier K⁺ current. Differential sensitivity to block by class III antiarrhythmic agents. *The Journal of general physiology*, 96(1):195–215.
- Sanguinetti, M. C. and Tristani-Firouzi, M. (2006). hERG potassium channels and cardiac arrhythmia. *Nature*, 440(7083):463–469.
- Shuttleworth, J. G., Lei, C. L., Whittaker, D. G., Windley, M. J., Hill, A. P., Preston, S. P., and Mirams, G. R. (2024). Empirical quantification of predictive uncertainty due to model discrepancy by training with an ensemble of experimental designs: An application to ion channel kinetics. *Bulletin of Mathematical Biology*, 86:2.
- Shuttleworth, J. G., Lei, C. L., Windley, M. J., Hill, A. P., Preston, S. P., and Mirams, G. R. (2025). Evaluating the predictive accuracy of ion-channel models using data from multiple experimental designs. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2292).

- Smucker, B., Krzywinski, M., and Altman, N. (2018). Optimal experimental design. *Nature Methods*, 15(8):559–560.
- Starmer, C., Nesterenko, V., Gilliam, F., and Grant, A. (1990). Use of ionic currents to identify and estimate parameters in models of channel blockade. *American Journal of Physiology-Heart and Circulatory Physiology*, 259(2):H626–H634.
- Starmer, C. F. and Courtney, K. R. (1986). Modeling ion channel blockade at guarded binding sites: application to tertiary drugs. *American Journal of Physiology-Heart and Circulatory Physiology*, 251(4):H848–H856. PMID: 2429564.
- Stoica, P. and Selen, Y. (2004). Model-order selection: a review of information criterion rules. *IEEE Signal Processing Magazine*, 21(4):36–47.
- Stork, D., Timin, E., Berjukow, S., Huber, C., Hohaus, A., Auer, M., and Hering, S. (2007). State dependent dissociation of hERG channel inhibitors. *British Journal of Pharmacology*, 151(8):1368–1376.
- Su, W.-H., Chou, C.-S., and Xiu, D. (2021). Deep learning of biological models from data: applications to ODE models. *Bulletin of mathematical biology*, 83(3):19.
- TeBay, C., McArthur, J. R., Mangala, M., Kerr, N., Heitmann, S., Perry, M. D., Windley, M. J., Vandenberg, J. I., and Hill, A. P. (2022). Pathophysiological metabolic changes associated with disease modify the proarrhythmic risk profile of drugs with potential to prolong repolarisation. *British Journal of Pharmacology*, 179(11):2631–2646.
- Thouta, S., Lo, G., Grajauskas, L., and Claydon, T. (2018). Investigating the state dependence of drug binding in hERG channels using a trapped-open channel phenotype. *Scientific Reports*, 8(1):4962.

- Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.
- Valentin, J.-P. (2010). Reducing QT liability and proarrhythmic risk in drug discovery and development. *British journal of pharmacology*, 159(1):5–11.
- Vandenberg, J. I., Perry, M. D., Perrin, M. J., Mann, S. A., Ke, Y., and Hill, A. P. (2012). hERG K⁺ channels: Structure, function, and clinical significance. *Physiological Reviews*, 92(3):1393–1478. PMID: 22988594.
- Vandenberg, J. I., Walker, B. D., and Campbell, T. J. (2001). hERG K⁺ channels: friend and foe. *Trends in Pharmacological Sciences*, 22(5):240–246.
- Villaverde, A. F. and Massonis, G. (2021). On testing structural identifiability by a simple scaling method: relying on scaling symmetries can be misleading. *PLoS computational biology*, 17(10):e1009032.
- Villaverde, A. F., Raimúndez, E., Hasenauer, J., and Banga, J. R. (2022). Assessment of prediction uncertainty quantification methods in systems biology. *IEEE/ACM transactions on computational biology and bioinformatics*, 20(3):1725–1736.
- Ward, M. D. and Ahlquist, J. S. (2018). *Maximum likelihood for social science: Strategies for analysis*. Cambridge University Press.
- Welch, C. S. and Cherry, E. M. (2025). Discovering cardiac action potential model equations using sparse identification of nonlinear dynamics. *Computing in Cardiology*, 52.
- Windisch, A., Timin, E., Schwarz, T., Stork-Riedler, D., Erker, T., Ecker, G., and Hering, S. (2011). Trapping and dissociation of propafenone

- derivatives in hERG channels. *British Journal of Pharmacology*, 162(7):1542–1552.
- Windley, M. J., Abi-Gerges, N., Fermini, B., Hancox, J. C., Vandenberg, J. I., and Hill, A. P. (2017). Measuring kinetics and potency of hERG block for CiPA. *Journal of Pharmacological and Toxicological Methods*, 87:99–107.
- Windley, M. J., Farr, J., TeBay, C., Vandenberg, J. I., and Hill, A. P. (2022). High throughput measurement of hERG drug block kinetics using the CiPA dynamic protocol. *Journal of Pharmacological and Toxicological Methods*, 117:107192.
- Windley, M. J., Lee, W., Vandenberg, J. I., and Hill, A. P. (2018). The temperature dependence of kinetics associated with drug block of hERG channels is compound-specific and an important factor for proarrhythmic risk prediction. *Molecular Pharmacology*, 94(1):760–769.
- Wu, X., Mcdermott, M., and Maclean, A. L. (2025). Data-driven model discovery and model selection for noisy biological systems. *PLOS Computational Biology*, 21(1):e1012762.
- Yang, F. and Zheng, J. (2014). High temperature sensitivity is intrinsic to voltage-gated potassium channels. *Elife*, 3:e03255.
- Yao, J.-A., Du, X., Lu, D., Baker, R. L., Daharsh, E., and Atterson, P. (2005). Estimation of potency of hERG channel blockers: impact of voltage protocol and temperature. *Journal of Pharmacological and Toxicological Methods*, 52(1):146–153.
- Zhang, S., Zhou, Z., Gong, Q., Makielski, J. C., and January, C. T. (1999). Mechanism of block and identification of the verapamil binding domain to hERG potassium channels. *Circulation Research*, 84(9):989–998.

Zhao, G., Qiu, Y., Zhang, H. M., and Yang, D. (2019). Intercalated discs: cellular adhesion and signaling in heart health and diseases. *Heart Failure Reviews*, 24(1):115–132.