Optimization of Ion channel recordings through analysis of multiple fittings

by

Aaron Luchko

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

We develop an approach for optimizing Hidden Markov model representations of voltage-gated ion channels that addresses the issues of topology determination and poorly performing optimization algorithms. Developing accurate models of neurological processes is a major goal of computational neuroscience, but creating accurate models of voltage-gated ion channels is a difficult task. Noisy data, a large range of potential topologies, and large numbers of parameters make machine optimization very difficult and topology comparison techniques unreliable.

We attempt to address the unreliability of the optimization process through multiple fittings. We then analyze the sets of fitted models with a new metric designed to measure consistency in the behaviour of the hidden states. When combined with the LogLikelihood this indicates whether the model has the complexity necessary to fit the data. We then design a protocol based around the creation of multiple fitted models that utilizes this metric both as a guide for further fittings and a way to identify a selection of suitable models and topologies.

We apply the metric to five sets of simulated data and two pairs of live recordings of voltage-gated K+ channels. On the simulated data the described protocol generated a range of topologies that successfully captured the correct topology in all but one of the simulated trials where it underestimated the topology required. Applied to the live data the procedure performed well on one channel type, for the other results were impacted by the difficulty of the optimization problem. In general the procedure and metric performed well but were limited by the ability of the optimizer to deliver a range of high quality solutions.

Acknowledgements

These Markov Models, eh? Why are they always hidden?

Are they up to something suspicious hiding away back there? I don't trust 'em.

@Neuro_Skeptic

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Voltage-gated ion channels form the basis of neuronal activity, as such generating accurate computational models of them is an important problem. One of the most common approaches uses Hidden Markov models (HMMs) to reproduce recordings of ion channel activity, but creating an appropriate HMM topology and finding a suitable parameterization is a difficult optimization problem. We approach the issue by generating multiple versions of fitted models and evaluating the models through a new metric, *max deviation*. Through this we attempt to detect if a topology is overparameterized, distinguish whether a group of models are poorly fitted or underparameterized, and alleviate some of the difficulties of dealing with a highly uncertain optimization process.

Among the things that make it a challenge are the extreme complexity [21] and diversity [13] of ion channels. This complexity makes HMMs a convenient basis with which to model them. Not only can the HMM topologies be modified to suit more or less complex dynamics but the hidden states of the HMM are analogous to the hidden internal behaviour of the ion channels.

But fitting the models is difficult with a high failure rate and an ambiguous definition of success. Fitted models can range anywhere from having no relationship to the data at all to a very strong fit. But the chances of achieving the global optimum are very unlikely, and even if we did it would be impossible to verify. Not knowing the optimal parameterization creates a second problem in comparing topologies. Did a model from one topology have a superior LogLikelihood (LL) because the topology was better suited, or because the optimization was just luckier? Without an accurate estimate of the optimal LLs for a set of topologies it

is difficult to detect overparameterization, particularly through the application of penalty based metrics.

To gain a better picture of the capabilities of each model topology we perform multiple fittings of it. We then address the problem of determining a successful fit by introducing a cutoff criteria, the *peak LL* range, that takes a range of the best fits. The peak LL range can be applied at the level of a single topology, to determine the best fits for that model configuration, and globally, to determine which topologies seem to be successful at reproducing the data.

Among the models in peak LL range we then apply our new metric, *max deviation.* The metric analyses the behaviour of the hidden states to estimate the degree to which the topology is free to reproduce the data. We use this metric in combination with the model scores to estimate if the models and their behaviour are exhibiting under or overparameterization.

Using max deviation as a guide we have developed a protocol to fit a group of topologies to an ion channel recording and present the user with a range of candidate topologies potentially able to capture the data.

1.1 Contributions

We explore the issue of modeling voltage-gated ion channels using the modeling tool QUB Express [30]. Through this we have made the following contributions:

1. A technique for evaluating the effectiveness of optimization settings and a mathematical model to rank the effectiveness of the settings.

2. A metric *max deviation,* designed to measure the similarity in behaviour of the hidden states in a set of HMMs when run against a given data set.

3. An optimization procedure based around generating multiple fittings of ion channel models for each model topology. Our procedure is based on the establishment of a set of test topologies and creating a standard configuration to apply to each topology. Both the number of fittings for each topology and the selection of recommended models is based on max deviation.

4. A QUB Express plugin and scripts designed to implement the described procedure.

1.2 Outline

- Chapter 2 Background: Defines necessary terms and concepts as well as provides background information on ion channels and ion channel modeling. We also review relevant literature about optimization of ion channels and topology determination.
- Chapter 3 Protocol for model fitting and topology determination: We include an overview of the live recordings we analyze and the detail some aspects of the artifact detection and correction. We then describe our method for evaluating optimizer configurations. Finally we describe the *max deviation* metric, our selection of sample topologies, and our optimization protocol.
- Chapter 4 Experimental Design: Includes descriptions of several recordings and the experimental configurations we use to test the viability of our approach.
- Chapter 5 Results: We perform the experiments then analyze and interpret the results.
- Chapter 6 Conclusions: A high level overview of our work and results
- Chapter 7 Future Work: Describes research ideas we would like to explore.

Chapter 2

Background

In this chapter we cover background material to assist in understanding our work. In section 2.1 we provide definitions for terms. Section 2.2 provides background information on ion channels and section 2.3 describes the patch clamp technique often used to supply ion channel recordings. Section 2.4 provides a brief overview of QUB Express, the primary tool we used during out investigations. One of the original ion channel models, the Hodgkin-Huxley equations, are covered in section 2.5 and we cover the formulation of the Hidden Markov Model based models we work with in section 2.6. Section 2.7 covers some related literature on fitting ion channel models while section 2.8 describes the LogLikelihood calculations that form the basis of the fitting technique used by QUB Express and are thus indirectly used by our approach. Finally section 2.9 describes several approaches to determining model topology.

2.1 Terminology

2.1.1 Topologies, Configured Topologies, and Models

When referring to ion channel models there are three levels of abstraction we refer to. The most general characteristic of a model is its **topology**; the topology is the underlying graph comprising of open and closed states connected by edges. Open topology states correlate to conductive channel states and closed to non-conductive, edges between states allow a model to transition directly from one state to another. More specific than a topology is a **configured topology**; the configuration includes parameters such as the reversal potential

and variables that determine the total conductance, however the parameters that govern the transition rates are still left unaltered. The most specific entity we refer to is a **model**; a model is a configured topology with fitted parameters. When the topology, configuration, and parameterization of the model are correct, the model can accurately reproduce the current of an ion channel exposed to the same voltage stimulus.

2.1.2 Ancestor/Descendant, Parent/Child

Topology A is considered an **ancestor** of topology B if the graph of A is a strict subset of B. Conversely topology B is a **descendant** of topology A if topology B is a strict superset of topology A. Topology A is a **parent** of topology B, and B is a **child** of A, if A is an ancestor of B that differs by the addition of a single edge (with a new state if necessary). A topology may have multiple parents or children.

2.1.3 LL

The likelihood of a model is the probability that the model, θ, would reproduce the data x, and is generally denoted $\mathcal{L}(\theta|x)$. For applications such as ion channel modeling the chances of a model reproducing a particular trace are extremely small. As such the natural logarithm of the likelihood is generally used instead, we denote the log-likelihood as **LL**. A model that always produced the given data would have a likelihood of 1 and a LL of 0 while a model that would never produce the data would have a likelihood of 0 and a LL of negative infinity. An optimization function using LL as an objective function attempts to maximum the given function.

2.1.4 Peak LL Range and Global Peak LL Range

Given a configured topology and a recording we would ideally find the globally optimal parameterization every time. Unfortunately globally optimal solutions are very difficult for the optimizer to find. Even when optimization is relatively successful the region around the global maxima is littered with local maxima so the vast majority of solutions close to the global maxima end up in one of these local maxima instead. We define the **peak LL range** of a configured topology to be the range of solutions with a LL close to the best result we have found for that configured topology. The rationale for basing the filter on LL is that LL is the objective function used by the optimizer and LL scales linearly with the number

of observations. Interpretations based on the Likeilhood itself are difficult since the very minor change in LL translates into a huge difference in the Likelihoods.

For our data sets we have used a range that includes all LLs within 5% of the best result for that configured topology. The choice of a 5% cutoff is a compromise between comparing models of a similar fit and finding enough fits to form a comparison. As well in some simple artificial models with baseline noise similar to the live data we found that even with the same configured topology we could generate fits with a LL about 2.5% better than the generative models themselves. A figure of 5% would put the generative model at the midpoint of the peak LL range.

Specifically if ll_{best} is the LL of the best fitting for a topology then the ll_{peak} is the lower bound of our cutoff and so only models with a LL greater than ll_{peak} are included in the peak LL range:

$$ll_{peak} = ll_{best} \times 1.05 \tag{2.1}$$

The **global peak LL range** is defined the same except ll_{best} is the LL of the best fitting across all configured topologies, it is generally used to indicate which configured topologies do a roughly equivalent job of fitting the data.

Note that LL is heavily influenced by the noise in the data set. Most of our recordings had similar noise levels but for data sets with alternate noise levels the 5% figure may not be appropriate.

2.1.5 Candidate Range

While the model with the best LL typically offers the best fit it does not necessarily represent the topology most suitable for the data. More complex topologies can achieve a higher LL while being less suitable as a result of overparameterization. Detecting overparameterization can be a difficult task even when the optimum fit for each topology is known. Since we cannot identifiy the optimum fit for a topology with any certainty we introduce a more conservative concept known as the **candidate range**. The candidate range is a subset of the full set of topologies that contains the true topology. Formally, when given a group of topologies T, where the true topology is t, then the candidate group is a subset $C \subseteq T$ where $t \in C$.

2.2 Ion Channel Basics

The cell membrane forms a barrier between the intracellular and extracellular fluids. In addition to protecting the inside of the cell the membrane also regulates the intracellular concentration of ions using ion channels and ion transporters.

Unobstructed ions flow in the direction of the electrochemical gradient, ion channels embedded in the cell membrane temporarily allow this unobstructed flow by opening and becoming permeable to their associated ion. Conversely ion transporters help maintain the imbalance between intracellular and extracellular ion concentrations by pumping ions across the membrane against the electrochemical gradient. Typically sodium ions (Na+) have a much higher extracellular concentration, while potassium (K+) ions have a higher intracellular concentration though the electrochemical gradient is determined by a combination of electrical potential and ionic concentration.

The membrane voltage is the difference in electrical potential between the cell interior and exterior. For each type of ion the direction and strength of the electrochemical gradient is determined by the difference between the membrane voltage and the reversal potential, also known as the Nernst potential, calculated using the Nernst equation [18].

For a specific ion, or group of ions, given the equilibrium potential, V_{rev}, ionic conductance g, and the membrane voltage, V_m, the ionic current, I, can be calculated with the following equation

$$I = g(V_m - V_{rev}) \tag{2.2}$$

Ion channels are selective to a specific class of ion, such as Na+, K+, Ca^{2+}, protons, or negative ions such as Chloride (Cl-). They change state in response to a variety of stimuli including voltage, neurotransmitters, mechanical force, and ATP among others. There are many different types of channels, and among the different types the kinetics they show in response to stimulus are highly diverse.

Individual channels are generally either fully open, closed, or inactive. But within those broad configurations there may be many sub-configurations. For voltage-gated ion channels the predisposition to assume different configurations is based on the membrane voltage. This in turn enables the ion channels to regulate the membrane voltage and leads to complex electrochemical phenomena such as the action potential.

The classic action potential is caused when the membrane voltage rises above the thresh-

old potential, this initial rise in voltage is either caused by external stimulus or the characteristics of the channels. This causes both Na+ and K+ channels to start activating, following the electrochemical gradient Na+ enters the cell and K+ exits. Na+ channels typically activate much more quickly than K+ channels so the net effect is a sudden increase in voltage described as depolarization. As the voltage peaks the Na+ channels begin inactivating while K+ channels continue to open. This repolarization stage is characterized by a drop in the voltage, driving the membrane voltage slightly below the initial voltage in a phase known as hyper-polarization.

2.3 Patch Clamp Experiments

The voltage clamp technique is one of the primary methods for investigating voltage-gated ion channels [17, 35, 25]. It aims to maintain the membrane voltage at a given level and measure the current resulting from the passage of charged ions through the open ion channels. One variation of the voltage clamp is the whole cell patch clamp as illustrated in figure 2.1. A glass pipette makes contact with the cell membrane, and a strong suction is applied to create a seal between the membrane and the pipette and rupture the membrane patch inside the pipette. This allows the electrode access to the intracellular environment.

The resistance across this seal is in excess of a gigaohm and is commonly referred to as a "gigaohm seal". The gigahom seal ensures that a negligible amount of current will be transmitted across the barrier so the electrical properties inside the pipette are the same as inside the cell. The voltage across the cell membrane can then be calculated using the pipette electrode and a reference electrode in the extracellular environment.

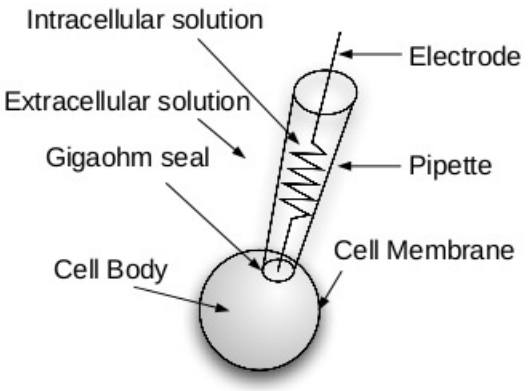


Figure 2.1: A basic patch clamp experiment, a section of cell membrane is sealed off, bathed in the experimental solution, and maintained at a prescribed voltage by the electrode.

A feedback amplifier acts through the electrode to set the membrane voltage at the command voltage specified by the experimenter. This is achieved through negative feedback; if the membrane voltage deviates from the command voltage then the feedback ampliphier injects current though the electrode

into the cell. The amount of current the amplifier inserts depends on the current being conducted through the cell membrane and the degree to which the membrance voltage must be modified. The electrode also records both the membrane voltage and membrane current.

The experimenter typically sets the command voltage to run through a voltage protocol. A typical voltage protocol consists of a series of sweeps at different voltage levels. The experiments we work with consist of a three stage protocol. The first stage is the conditioning prepulse. The prepulse is applied for an extended period and is used to put the channels into steady state so the probability of a channel being open is constant. Typically the prepulse attempts to put the channels into a fully closed configuration. The prepulse is followed by the test pulse, the test pulse changes the voltage to trigger channel activation. The voltage level used in the test pulse is the variable that changes between sweeps. Finally the deactivation pulse closes the channels and/or returns them to the steady state equillibrium of the pre-pulse.

The specific voltage protocol for the experiments we worked with is covered in more detail in section 3.2.1.

2.4 QUB Express

QUB Express [30] is a software tool designed for the Hidden Markov model simulations of ion channels. Among other functions it can optimize single [33, 34] and multiple [28] ion channels as well as generate simulated recordings of ion channel models.

The model formulation is as described in section 2.6 and the optimization is based on the LogLikelihood objective function from section 2.8, the actual optimization is performed by the global optimization algorithms described in section 2.7. We selected QUB Express as a basis for our approach for several reasons. First it was a cross-platform tool, written mostly in Python with performance intensive sections in C++. We also found it to fit models more quickly and accurately than other HMM based ion channel simulation tools we investigated such as NEURON [5]. Finally QUB Express includes well developed interfaces for scripting and plug-in support. This allowed us an environment in which there was straightforward way to perform model analysis and other complex calculations and to integrate more functionality directly into QUB Express.

2.5 Hodgkin Huxley Equations

The first significant attempt to produce a mathematical model of ion channels was performed in 1952 by Alan Lloyd Hodgkin and Andrew Huxley [19]. While other formulations have been developed their formulation is still among the most widely used in the field for modeling K+ and Na+ channels.

Their approach hypothesized independent gating variables that represented the probable configuration of the channel. Their first work was on K+ channels with a gating variable denoted n. Each gating variable representes the fraction of that variable in the open configuration. All gating variables would need to be open for the channel itself to be open and conducting.

To compute the value of n they assumed that at any given voltage the gating variable could change from closed to open or open to closed and the probability of a change was dependent only on the current voltage. They denoted the rate that n would open as $\alpha_n(V_m)$, where V_m is the membrane voltage, and the rate that n would close as $\beta_n(V_m)$. Thus the rate of change of the gating variable n is.

$$\frac{dn}{dt} = \alpha_n(V_m)(1-n) - \beta_n(V_m)n$$

If the voltage is not changed the channel will eventually reach *steady state activation.* For n the steady state value is determined:

$$n_\infty(V_m) = \frac{\alpha_n(V_m)}{\alpha_n(V_m) + \beta_n(V_m)}$$

If the channel is in steady state at the initial voltage V_0 then changed to V_m and held for time t the value of n will be

$$n(t) = n_\infty(V_m) - (n_\infty(V_m) - n_\infty(V_0))\exp(-\frac{t}{\tau_n})$$

Where the time constant τ can be calculated:

$$\tau_n = \frac{1}{\alpha_n(V_m) + \beta_n(V_m)}$$

It follows that the opening and closing rates α and β are a function of the new steady state value of n and τ.

$$\alpha_n(V_m) = \frac{n_\infty(V_m)}{\tau_n(V_m)}$$

$$\beta(V_m) = \frac{1 - n_\infty(V_m)}{\tau_n(V_m)}$$

Working from recordings they could determine the initial activation, steady state activation, and the time required to reach steady state activation to solve the system of equations. As they did this they also determined the relevant gating variables through trial and error, eventually determining that four copies of the gating variable had to be active for the channel to be open, or n^4. For $K+$ channels defining the number of channels as N_{cK} and the conductance per channel as C_K, the net conductance for potassium is:

$$g_K = N_{cK} C_K n^4$$

A similar procedure was used with Na+ channels, in addition to another activation variable m they added a second variable h to represent the inactivation that $Na+$ channels exhibited when held open. Though a similar trial and error process they determined the probability of an open unblocked channel to be m^3h:

$$g_{Na} = N_{cNa} C_{Na} m^3 h$$

Both sets of gating variables turned out to correspond to the physical structure of the channels in question. The four $K+$ variables corresponded to four proteins that all had to be in an open state for a $K+$ channel to conduct, while the $Na+$ had a three protein gate with a separate blocking protein h that would inactivate the channel at higher voltages.

The Hodgkin and Huxley equations are still useful though they do have limitations, they suggest inactivation is purely voltage dependent when evidence suggests that inactivation is triggered by the channel being activated [2][27]. Hodgkin and Huxley also models conductance as a continuous variable when conductance is in fact a stochastic system based on many independent ion channels [42].

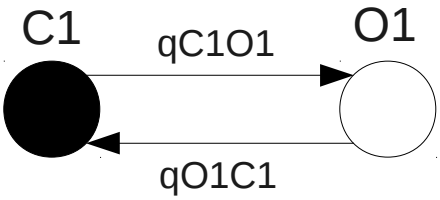


Figure 2.2: A simple two state model where **O1** represents the open state and **C1** the closed. **qC1O1** is the transition rate from **C1** to **O1** and **qO1C1** from **O1** to **C1**.

2.6 Hidden Markov Model representation

A Hidden Markov Model (HMM) is a Markov process where the system's state cannot be observed directly, and instead must be inferred by observing the system's output [3, 37]. They are well equipped to describe the complex kinetics of ion channels. Similar to the Hodgkin-Huxley equations one may attempt to map the states of an HMM to protein configurations in the channel but a physical analogue is not required.

Figure 2.2 shows a basic two state model consisting of an open (O1) and closed state (C1), note that in general HMMs do not distinguished between closed and inactive channel configurations and both are treated as closed.

Colquhoun and Hawkes used a Markov process to model single channels when they investigated the behaviour of single channels. They demonstrated that channels had clusters of short bursts of conductance [7, 8]. The short duration of the bursts meant they were often missed by the recording equipment, as such modeling methods must deal with these missed events. When calculating the Likelihood of an HMM one can include the potential for short lived events occurring between observations as done by Colquhoun [8, 9], Qin [33, 34, 35], and others [6, 20].

Qin's formulation was extended to multiple channel systems by Milescu et al. [28] and later implemented in QUB-Express [30].

Take the state probability vector, P_t, of which the individual elements $P_t[i]$ represent the probability that the channel is in state i at time t. The mean conductance of a group of channels, g, can be obtained by taking the product of P_t with the transposed vector of state conductances, C, and multiplying by the the number of channels, N_c.

$$g = N_c P_t * C' \tag{2.3}$$

Similarly the mean current conducted by an individual channel in a given state is calcu-

lated

$$\mu_c = C' * (V - V_{rev}) \tag{2.4}$$

Combining either of these formulas with the current formula in equation 2.2 and we can calculate I_t, the average current due to the channels at time t.

$$I_t = N_c * P_t * C' * (V_m - V_{rev}) \tag{2.5}$$

The ith element ofP_t is the probability that a channel will be in state i at the time t. In each interval P_t is updated by dP which is calculated using the transition matrix Q

$$P_{t+d} = P_t + dP$$

$$dP = Q_t P_t dt \tag{2.6}$$

Q is an $N_s \times N_s$ matrix where N_s is the number of states and each entry q_{ij} is the flow rate from state i to$j$. The formula used to calculate each q_{ij} varies depending on the model formulation but the formulation used by QUB-Express calculates the rates as $q_{ij} = k0_{ij} * \exp(k1_{ij} V_m)$, where V_m is the membrane voltage and $k0$ and $k1$ are model parameters. Elements along the diagonal, q_{ii} instead show the rate at which channels leave state i so Q is described as:

$$Q = \begin{cases} i \neq j & q_{ij} = k0_{ij} * \exp(k1_{ij} V_m) \\ i = j & q_{ii} = -\sum_{i \neq j} q_{ij} \end{cases} \tag{2.7}$$

The steady state probability, denoted P_s, is the value of P_t for which $dP = QP_t dt = 0$. The value of P_s can be determined by solving for $P_s Q = 0$ while ensuring that $\sum P_s = 1$. This can be done by solving the following equation where $Q1$ is Q with a row of 1's at the end and setting it equal to a vector of length $N_s + 1$ consisting of N_s 0's followed by 1.

$$P_s \times Q1 = \begin{pmatrix} \overrightarrow{0} \\ 1 \end{pmatrix}$$

To track the progression of P we do not actually use the formula $dP = QP_t dt$ since the

system responds far more quickly than typical values of dt. The individual q_{ij} elements are so large that numerical integration creates an inaccurate approximation of dP. In response to changes in voltage some models even generate elements in dP far greater than one, causing the system of equations to quickly break down. Instead we calculate the transition probability matrix A_t using the matrix exponential expm.

$$A_t = expm(Q_t dt) \tag{2.8}$$

Using Q_t directly in equation 2.6 assumed that a channel could make only a single transition during dt. A_t instead allows for the possibility of multiple transitions during dt so each element in the transition matrix a_{ij} is the conditional probability that a channel in state i at time t will be in state j at $t + dt$, assuming every state is reachable all a_{ij} will have a non-zero value. This method allows for a more accurate calculation of dP.

$$dP = A_t * P_t \tag{2.9}$$

2.7 Parameter Determination

Determining the correct parameters to allow a HMM to reproduce an ion channel recording is a widely studied problem. While experts will often determine both the model topology and individual parameters by hand, either task may be approached through an algorithmic approach.

Hodgkin and Huxley fit their equations to the the *Loligo* (squid) giant axon using the disjoint method [19, 46]; a set of two voltage clamp experiments designed to measure different characteristics of the model. The first stage involves a prepulse to hyper-polarize the channels followed by an extended test pulse to trigger channel activation and eventual inactivation. The time taken for activation and inactivation at the various voltages allows the formulas for the activation and inactivation time constants to be derived. As well, the steady state activation curve is derived by observing the activation at different voltages. In a second experiment the prepulse varies and causes activation while the test pulse has a uniform voltage to trigger complete inactivation; again the time constants can be derived but the varying test pulse allows for calculation of the steady state inactivation curve.

Willms et al. [46] uses an alternative approach to fitting the Hodgkin and Huxley models

by takes advantage of standard nonlinear optimization algorithms to fit the entire set of equations at once.

For HMMs expectation-maximization techniques such as Baum-Welch [23, 3] are popular for estimating the transition rates of Q for single [6] and multiple [44] channel models, though are generally less useful in determining the parameters for the equations that create the transition rates. Zhou, Pearson, and Auerbach [48] even developed an analytical solution for determining transition rates for a subclass of single channel models.

Gurkiewicz [16] and Menon [26] optimized parameterized models using HMM based formulations optimized with genetic algorithms. NEURON [5], an ion channel simulation tool, has a model optimization framework. It includes the PRincipal AXIS (PRAXIS) algorithm [14], which attempts to find derivatives numerically and perform a search orthogonal to the gradient. We also integrated Gurkiewicz's genetic algorithm and our implementation of the Cross Entropy Method (CME) [11] but had little success in fitting live data.

The method implemented in QUB-Express uses an objective function based the calculation of the Log Likelihood for single channel [33, 34] and multiple channel models [35, 36, 28, 29]. QUB-Express includes three optimization algorithms. The first algorithm is the Nelder-Mead simplex method [32]. A simplex is the generalization of a triangle to an arbitrary number of dimensions by creating a shape with $k + 1$ vertices where k is the number of dimensions, for instance a 3 vertex triangle in 2 dimensions and a tetrahedron in 3. The simplex method works by evaluating the objective at each vertex then discarding the worst vertex and evaluating a new point in its place. It is generally used as the initial optimization step because it is more resistant to valleys and the lack of gradient calculations allow it to process the large initial search space more quickly.

The next algorithm is the Davidon-Fletcher-Powell (DFP) formula [10], a quasi-Newton method that uses the gradient of the estimate to direct the search and is more aggressive in moving towards a solution. Since the original development of the LL algorithm for the single channels the standard approach has been to perform an initial pass of the Simplex algorithm to reach a favourable global region followed by an application of DFP [35].

A recent addition to QUB-Express is the Broyden–Fletcher–Goldfarb–Shanno (BFGS) [4] algorithm. It is an successor of the DFP algorithm and uses a similar gradient based method. The strategy suggested by Qin and later used by Milescu used an initial run by the Simplex algorithm followed by an additional pass by DFP.

Due to our extensive usage of QUB-Express we include a summary of the LogLikelihood

algorithm.

2.8 Calculating the LogLikelihood

The QUB-Express HMM model formulation best described by Milescu [28] contains sources of current in the form of Gaussians representing channel current and baseline current. The state conductance vector has mean C and variance V^C with the mean channel current calculated by equation 2.5. The baseline current is a Gaussian random variable which represents the current from non-channel sources, and has mean μ^B and variance V^B. The baseline is described in terms of current instead of conductance as the reversal potential of the baseline is generally unknown. Combining these gives the mean total current:

$$\mu_t = \mu^B + N_c * P_t * C' * (V_m - V_{rev}) \tag{2.10}$$

At this point a common objective function is the Sum of Squared Differences (SSD) that compares the mean simulated current to the measured current I_t^{meas}

$$SSD = \sum_{t=1}^{t=T} (I_t^{meas} - \mu_t)^2$$

However, this ignores the opportunity to extract information from variance in the recordings. By taking into account the variance in the model the noise in the recording can help inform the fit.

The model variance comes from three sources, the variance in the baseline current V^B, the amount of conductance from an individual channel in a specific state C_V, and the number of channels in each state. The formula for the total variance from individual channels is based off of equation 2.5 and is as follows:

$$V_{ct} = N_c * P_t * C'_V * (V_m - V_{rev}) \tag{2.11}$$

The variance from the number of channels in each state is $N_c V_t^x$ where V_t^x is the state covariance matrix at time t based on the probability vector P_t.

$$V_t^x = \begin{cases} i \neq j & -P_t[i]P_t[j] \\ i = j & P_t[i](1 - P_t[i]) \end{cases} \tag{2.12}$$

The corresponding variance in current due to the state fluctuations is determined by multiplying the state fluctuations by the mean squared current for each channel (equation 2.4). The total variance due to the state fluctuations is:

$$V_s = \mu_c^T N_c V_t^x \mu_c \tag{2.13}$$

Combining the variance from state fluctuations in equation 2.13 with the total per channel variance in equation 2.11 and the baseline variance V_B we get a total variance of:

$$V_t = V_B + N_c(\mu_t^T V_t^x \mu_t + P_t^T V^c) \tag{2.14}$$

Since V_B, V_t^x, and V^c are all Gaussian and the only operations are addition and multiplication the resulting total variance V_t is also Gaussian.

Having a method to calculate the mean and variance of the current generated by the model it is possible to calculate the likelihood. We denote θ as the description of the model. The likelihood is the probability that a simulation of the model reproduces the given current trace, I.

$$L = p(I|\theta)$$

This in turn is equivalent to the product of the probability that the model reproduces the current at each individual time step, $L = \prod_{t=0}^{T} p(I_t|\theta)$. This likelihood can be an extremely small number so the logarithm is used instead.

$$LL = \ln(\prod_{t=0}^{T} p(I_t|\theta)) = \sum_{t=0}^{T} \ln(p(I_t|\theta)) \tag{2.15}$$

Since the current is a Gaussian with mean $u_t$from equation 2.10 and variance V_t from equation 2.14 the probability of a specific observation, I_t, is given as follows

$$p(I_t|\theta) = \frac{1}{\sqrt{2\pi V_t}} \exp(\frac{-(I_t^{meas} - \mu)^2}{2V_t}) \tag{2.16}$$

Taking the natural logarithm yields

$$\ln(\frac{1}{\sqrt{2\pi V_t}} \exp(\frac{-(I_t^{meas} - \mu)^2}{2V_t})) = \ln(\frac{1}{\sqrt{2\pi V_t}}) + \frac{-(I_t^{meas} - \mu)^2}{2V_t}$$

$$\frac{\ln(2\pi V_t)}{2} - \frac{(I_t^{meas} - \mu)^2}{2V_t} = \frac{1}{2}\left(\ln(2\pi) + \ln(V_t) - \frac{(I_t^{meas} - \mu)^2}{V_t}\right)$$

Extending this to all time steps in the trace

$$LL = \sum_{t=0}^{T} \frac{1}{2}(\ln(2\pi) + \ln(V_t) - \frac{(I_t^{meas} - \mu)^2}{V_t})$$

$$\frac{1}{2}\sum_{t=0}^{T} \ln(2\pi) + \ln(V_t) - \frac{1}{2}\sum_{t=0}^{T} \frac{(I_t^{meas} - \mu)^2}{V_t}$$

$$\frac{1}{2}(\sum_{t=0}^{T} \ln(2\pi) + \sum_{t=0}^{T} \ln V_t - \sum_{t=0}^{T} \frac{(I_t^{meas} - \mu)^2}{V_t})$$

This leads to an estimation of the LL for a model to reproduce a given recording

$$LL = \frac{1}{2}((T+1)\ln(2\pi) + \sum_{t=0}^{T} \ln V_t - \sum_{t=0}^{T} \frac{(I_t^{meas} - \mu)^2}{V_t}) \tag{2.17}$$

2.9 Topology Determination

Determining the proper topology is a critical part of modeling. For some applications researchers may favour particular topologies, but they often have have no scientific preference to a specific topology and primarily want a topology that is implied by the data. For these situations an algorithmic approach to topology determination is ideal.

Hodgson [20] approached the problem of topology determination from single channel data. By analyzing the duration of closed sections they could discriminate between several simple topologies due to the restrictions the topologies placed on model behaviour. However this method is limited to single channel data and is difficult to extend to more complex topologies.

When choosing between topologies one common strategy is to introduce a penalty term that penalizes more complex models. Two of the more common ones are Akaike information criterion (AIC) and Bayesian information criterion (BIC) [24] though other penalty terms are common. If the number of parameters is k and data points is n then the formulas are as follows:

$$AIC = 2k - 2LL \tag{2.18}$$

$$BIC = -2LL + k\ln(n) \tag{2.19}$$

Siekmann et al. [39] develop a procedure based on Markov chain Monte Carlo (MCMC). They use MCMC to determine statistical profiles for each of the rate constants in Q by sampling and scoring different models from the rate space. In addition to creating estimates for Q from the means of the MCMC derived profiles. they are also able to detect overparameterization through parameters which show poor convergence. Unfortunately, the sampling approach of the MCMC is much slower than traditional optimization, and their approach was limited to determining only the rate constants Q rather than the parameters that generated Q.

Menon's genetic algorithm based procedure [26] mutated model architectures along with the parameters. Their procedure started with 100 models within a set complexity range that all bore random topologies, the objective function was SSD with a penalty function based on the number of edges. Each generation the worst models of the generation would be pruned and replaced with modified versions of the survivors, and the survivors would experience a mutation in either their rate constants or their topologies. Their algorithm had an additional complication not present in our data in that their data used six different experimental protocols. To deal with the six protocols they ordered the protocols and for each generation n would complete the selection and mutation procedure once for each protocol. Their procedure prescribed 3,000 generations per protocol which given the six protocols meant a total of 18,000 generations and 1,800,000 simulations. On live data they were able to converge to the same topology with a similar parameterization in 2 of 3 runs with the third run resulting in a model with an extra edge. The strength of their technique was fitting a protocol that required a complex unknown topology.

Speech recognition also makes extensive use of HMMs and those techniques could potentially be applied to ion channel model determination as well. Rowels [38] proposes that models be constrained by a spatial topology. If the HMM does in fact map to a physical representation then it could be subject to some physical constraints as well. Since the underlying system has limited degrees of physical freedom each state in the HMM should also have limited degrees of transitions. For their application of mouth movements for speech recognition Rowels proposes that each state be limited in degree as the mouth is physically limited in the number of configurations it can reach in a single step. While the topologies proposed for ion channels rarely approach the size or degree as the speech recognition models

that Rowels describes it does suggest large fully connected topologies are unlikely to have physical analogues.

Vasko et al. *[43]* describe a method that starts with a large fully connected topology and goes through pruning iterations, successively removing the transition least important to the fit, until only a single transition remains. Afterwards the candidate model is chosen by a selection criteria from among the generations of pruned models. The downside of this strategy as applied to our domain is they attempted the simpler problem of fitting transition rates as opposed to a model that generate transition rates based on an input. While their fits of complex topologies were generally straightforward, in our experience QUB-Express optimization often fails, particularly for overparameterized topologies. Obtaining a top fit from which to perform the edge subtraction is a non-trivial problem, particularly as model complexity decreases and it becomes unclear if poor fits are due to luck or an underparameterized topology.

Chapter 3

Protocol for model fitting and topology determination

We develop a protocol to aid in the development of ion channel models from a recording of channel current. The goal is to develop an automated method to generate a model, or range of models, to fit a set of ion channel recordings.

Section 3.1 lays out the stages of the protocol in broad terms, the details and motivation behind these stages is covered in the subsequent sections.

The first stage of our protocol involves preparing the recording for modeling and configuring the topologies so that they can best represent the data. In section 3.2 we detail the analysis and preparation of recordings of two ion channel types that we later fit in chapters 4 and 5. This is followed by section 3.2.2 where we provide a list of the topologies we use for fitting.

Next we investigate the reliability of the optimization process itself in section 3.3. Here we measure the effectiveness of different optimizer configurations and attempt to build a statistical model of the optimizer results. While we do not utilize this approach directly in our protocol we do base the optimization configuration we use for our experiments on the results of this investigation. More importantly this section outlines the difficulty of the optimization process as for some problems not only do less than 1% of the fits succeed, but it can be unclear if the best fits are even close to a globally optimal solution for that configured topology.

Section 3.4 covers our new metric, *maximum deviation,* that measures the behaviour of

hidden states and how we utilize it in our optimization protocol. Section 3.4.2.3 covers the usage of max deviation in determining the range of suggested topologies and section 3.4.2.2 covers the rules we use to determine whether additional rounds of optimizations are required.

Finally in section 3.4.3 we detail the implementation of the protocol in the ion channel simulation environment QUB Express [30].

3.1 Model Fitting Protocol

The full protocol involves three stages, a configuration stage, an optimization stage, and an analysis stage.

1. Configuration
    (a) Prepare the data as outlined in section 3.2. This involves looking for issues in the recording, truncating extraneous data, and correcting issues if necessary and/or possible.
    (b) Create a list of target topologies. For our experiments we used the list from section 3.2.2.
    (c) Based on the data analysis create a configuration to be applied to all topologies.
    (d) An optional stage where different optimizer settings are evaluated against the problem is covered in section 3.3.
2. Optimization
    (a) Perform an initial optimization round, creating 100 fittings of each topology as described in section 3.4.
    (b) For every topology that requires additional fittings according to table 3.4 double the number of fittings. Continue until all topologies have reached a termination criteria or hit a maximum number of fittings.
3. Analysis
    (a) Select the topologies that can reproduce the data without excessive overparameterization according to the rule laid out in section 3.4.2.1. From that subset of topologies the researcher can select a model based on their own criteria. Note that analysis can be run after every optimization round.

3.2 Data Preparation and Topology Configuration

Fitting accurate models of ion channels requires data of sufficient quality and properly configured topologies. Among the issues that can impact channel recordings are recording artifacts, data synchronization, and background noise. Failing to address these issues may lead to a recording that displays physiologically impossible data or a model that does not accurately represent the true physiology.

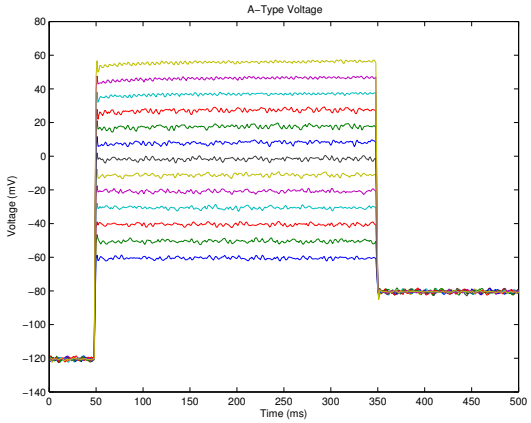
In addition to high quality data, a proper topology configuration is also necessary to produce an appropriate parameterization. Total conductance and reversal potential are clearly necessary to reproduce the data, but also important are sources of variance. Since the LL objective function does not calculate the difference between the mean trace and the data, but the probability that the model can reproduce the data, the sources of variance are a critical part of this calculation.

For both the data preparation and the topology configuration the input of an expert who understands both the data collection method and the underlying channel physiology is critical. This section will largely focus on the analysis of two pairs of recordings obtained from Patrick Stemkowski [41]. He provided valuable feedback in understanding the recording environment, experimental protocol, and channel properties. He was also involved in the discussion of how to deal with issues in the data and set proper configurations for the topologies.

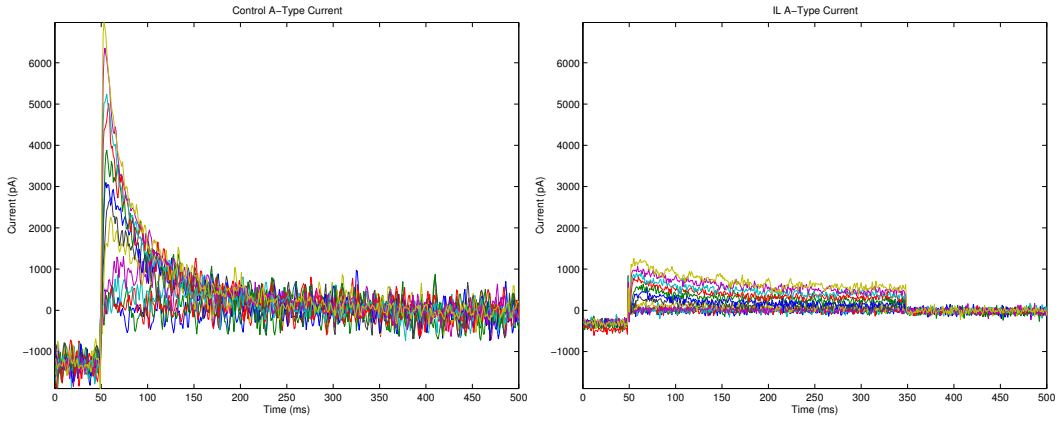
3.2.1 Recording Overviews

In the following section we give an overview of the data sets we worked with and detail a few of the issues we encountered. The experiments were performed to investigate the effects of the signaling molecule Interleukin-1 beta (IL-1β) on K+ channels in medium sized dorsal root ganglion (DRG) neurons. The cells were harvested from male Sprague-Dawley rats and subjected to 5-6 days of a control treatment of IL-1β before the patch clamp procedure was applied [40][41]. We investigated two types of K+ channel that were subjected to this treatement, an *A-Type* channel exhibiting fast inactivation and a *Delayed Rectifier* channel showing slow inactivation.

Voltage patch clamps as described in section 2.3 were used to apply a voltage protocol to the cells, the protocol consisted of three sections: a conditioning prepulse, a stimulation pulse, and a deactivation pulse. Figures 3.1 and 3.2 show measured voltage and current,



(a) The voltage protocol used in the *A-Type* experiments.

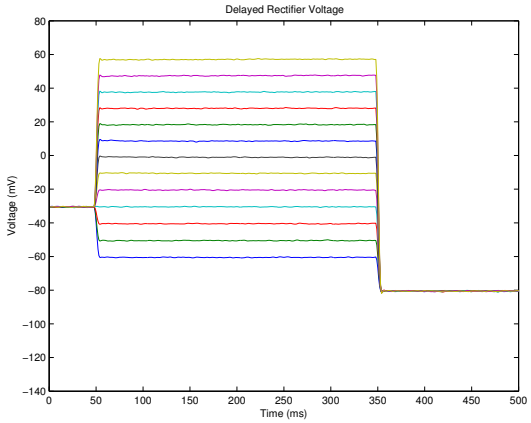


(b) The original recordings of the *A-Type* current under control conditions and IL-1β.

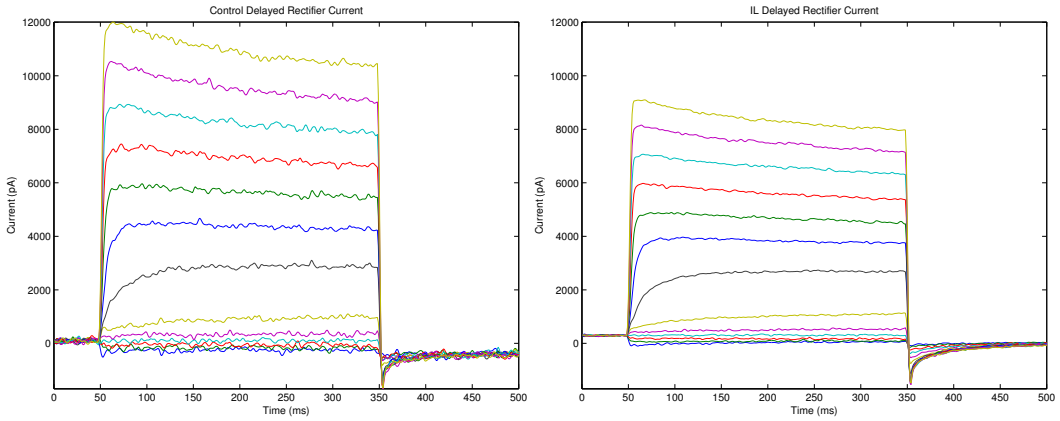
Figure 3.1: Original voltage and current traces from the *A-Type* recordings.

applied voltage was the same for both control and IL-1β but depending on the channel type a different conditioning prepulse was used.

Each recording consisted of 13 sweeps, the conditioning prepulse at the start of each sweep established equilibrium conditions prior to the stimulation pulse. The stimulation pulse was a 300 ms pulse from ranging from -60 mV to 60 mV in 10 mV increments followed by a deactivation pulse of -80 mV. The original data included an additional 500 ms of conditioning prepulse and 1500 ms of deactivation pulse which were both truncated. During data collection the extended conditioning pulse was necessary to ensure the channels had reached equilibrium. But simulations can start at equilibrium so only 50 ms of conditioning prepulse was necessary to establish the starting conditions and ensure that region of data was fitted.



(a) The voltage used in the Delayed Rectifier experiments. The conditioning pulse is the only place this protocol varies from the A-Type protocol



(b) The unaltered recordings of the Delayed Rectifier currents under control conditions and IL-1β.

Figure 3.2: Original voltage and current traces from the *Delayed Rectifier* recordings

Idealized Vs Recorded Voltage The existence of recorded voltage necessitated the decision of whether to use idealized or recorded voltage. If voltage recordings are absent or incomplete it may be necessary to make assumptions about the missing data. Idealized voltage is one standard approach that assumes the voltage acts as a step function providing an instantaneous transition between pulses. Since our voltage recordings were smooth and continuous during transitions between pulses we had the option of using recorded voltage. After analyzing both scenarios we found the idealized voltage was not consistent with the recorded current, thus we used recorded voltage for our simulations.

The principal difference between the recorded and idealized voltages was the transition between pulses, the idealized voltage would switch instantaneously while the recorded voltage would take 3-5 ms to rise or fall to the new voltage. This period was critical to the simulation since significant changes in current occurred over this period. The change was significant enough that models fitted under one voltage scenario would show different behaviour under the other.

After examining current and conductance traces under both scenarios we chose to use recorded voltage. Figure 3.3 demonstrates the physiological implausibility of the idealized voltage scenario. It displays the final sweep of the Control *Delayed Rectifier* recording at the transition from the stimulation pulse to the deactivation pulse. At 349 ms the recorded current starts dropping from above 10,000 pA and at 352 ms the recorded current drops below zero. Since the reversal potential is -58 mV if we were to assume an instantaneous transition to from 60 mV to -80 mV at 349 ms then the recorded current from 349 ms onward would be negative. This is clearly not the case so an instantaneous transition at 349 ms is not plausible. A later instantaneous transition is similarly implausible as it raises the question of what stimulus caused the channels to deactivate at 349 ms. This issue was typical when assuming the idealized voltage, in contrast the recorded voltage was generally consistent with the recorded current in the transition regions. After consultation with Pat Stemkowski we decided recorded voltage was more appropriate for our simulations.

Evaluating Physiological Viability Although the recorded voltage meant the pulse transitions were generally well behaved there remained issues with other sections of the recordings. The first three sweeps of the *Control Delayed Rectifier* recording had a negative current when any K+ current at the supplied reversal potential would be positive. A second issue occurred during the conditioning prepulse for the *A-Type* recordings. The prepulse

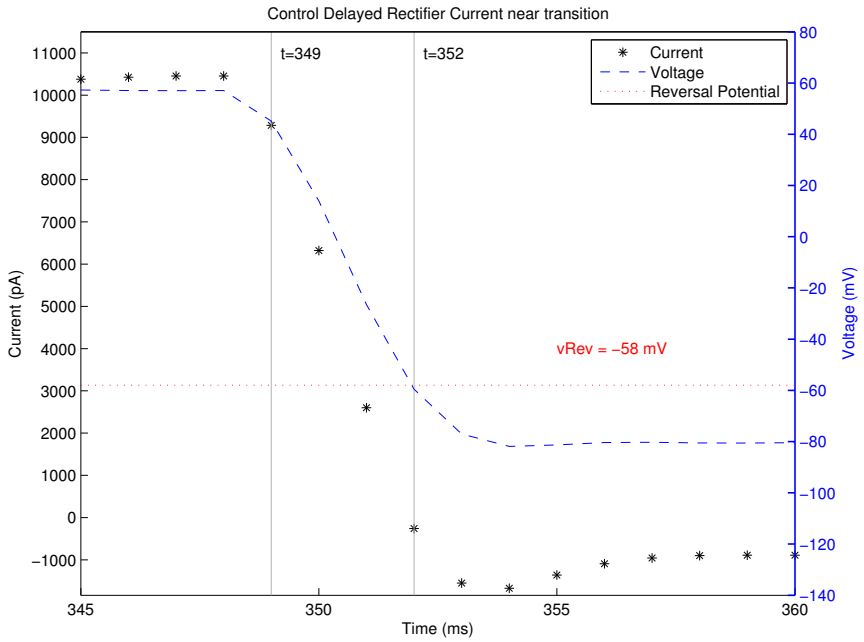


Figure 3.3: Current during the transition from the stimulus pulse to the deactivation pulse for the final sweep of the Control *Delayed Rectifier* recording.

current was consistent with the reversal potential but inconsistent with established *A-Type* channel behaviour. Further investigation revealed this to be a correctable artifact of the subtraction protocol used to isolate the currents.

Figure 3.4 shows a closer look at the current from the first three sweeps which show negative currents with means -257,-166, and -58 pA. This negative current is inconsistent with the stimulation pulses for those sweeps of -60, -50, and -40 mV respectively. Current for the initial -60 mV sweep should have been negligible and the following two should have been slightly positive. The magnitude of these currents was quite small compared to later sweeps such as the final sweep with a mean of 10,917 pA. There were several potential causes for this discrepancy including: a change in reversal potential, the presence of an additional current source, or an instrument error. The first two would be difficult to correct since a change in reversal potential would be difficult to determine precisely from the recording data alone. An additional current source would have an unknown reversal potential and conductance and would have an unpredictable effect at different voltages.

An instrument calibration error however would have a stable effect throughout the simulation and thus could be easily corrected. As well there is evidence in favour of a calibration error in the form of negative current during the deactivation pulse. This current is possible since the deactivation pulse is below the reversal potential but the magnitude of the

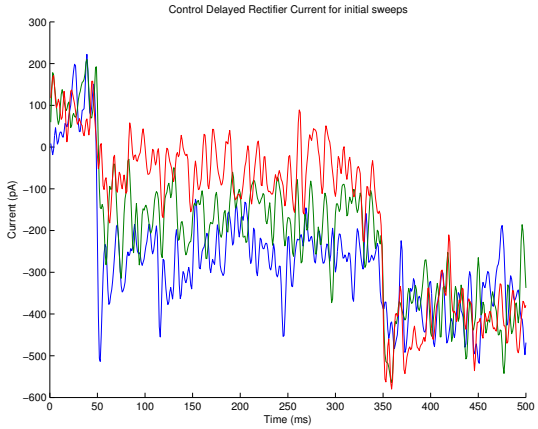


Figure 3.4: Negative current in the first three sweeps of the Control *Delayed Rectifier*

negative current suggests an unlikely level of activation for K+ channels at that voltage. A correction of 257 pA makes the initial sweeps plausible but also brings the deactivation pulse closer to zero, suggesting a more plausible level of channel activity. Note this still leaves a negative current of about 150 pA during the deactivation pulse, that suggests a level of channel activation that while consistent with the reversal potential, is unexpected.

The second issue was negative currents during the conditioning pulse for the *A-Type* recordings. These negative currents are evident in Figure 3.1. The negative current is consistent with the reversal potential but inconsistent with the known behaviour of *A-Type* channels. At -120 mV the channels should be closed and conducting no current. Investigation revealed the current to be an artifact of the subtraction protocol used to isolate the currents.

The protocol was required as the medium DRG neurons contained both *Delayed Rectifier* and *A-Type* K+ channels. To separate them the protocol created two recordings, one showing only a *Delayed Rectifier* current and the other a combined trace with both *Delayed Rectifier* and *A-Type* currents. The *A-Type* only current was then created by subtracting the *Delayed Rectifier* current from the combined trace.

The combined trace was recorded using a conditioning prepulse of -120 mV, since both channel types were closed the prepulse current was zero and both would be available to activate when the stimulation pulse arrived.

The problem was in the creation of the *Delayed Rectifier* only recording. To eliminate the *A-Type* current it used a -30 mV conditioning prepulse, this was high enough to activate and subsequently inactivate all *A-Type* channels during the prepulse. Thus when the stimulation pulse arrived only *Delayed Rectifier* channels were available to activate and provide current.

Unfortunately this was also high enough to activate some *Delayed Rectifier* channels which remained active throughout the prepulse creating a small positive current.

Subtracting this small positive *Delayed Rectifier* current from the zero current of the combined recordings created a negative current in the calculated *A-Type* recording that was not the result of *A-Type* channels. Since the issue only affected the prepulse region we could correct the recording setting the current in this region to zero.

3.2.2 List of Model Topologies

We have included a variety of topologies to use for analysis. They include examples from literature and formations we were interested in. Note that for topologies taken from literature they often use different formulas to produce the transition rates so they are not truly equivalent. In Figure 3.5 we show the complete list of topologies in tree form starting with the simplest model, v2_e1, at the root. The concept of model ancestry and descendance is important to the application of maximum deviation in determining correct model complexity.

The first set of topologies are simple linear topologies that provide a straightforward example of increasing complexity. The trivial v2_e1 model in Figure 3.6 is the most basic topology. It is also significant in that it is the only topology unable to model inactivation. The following models from Figures 3.7, 3.8, 3.9, and 3.10 are linear extensions of this model up to a depth of 5 with v6_e5. These models have some usage in literature with both v3_e2 and v4_e3 being used by Gurkiewicz et al. *[16]*.

Figure 3.6: v2_e1

Figure 3.7: v3_e2

Figure 3.8: v4_e3

Figure 3.9: v5_e4

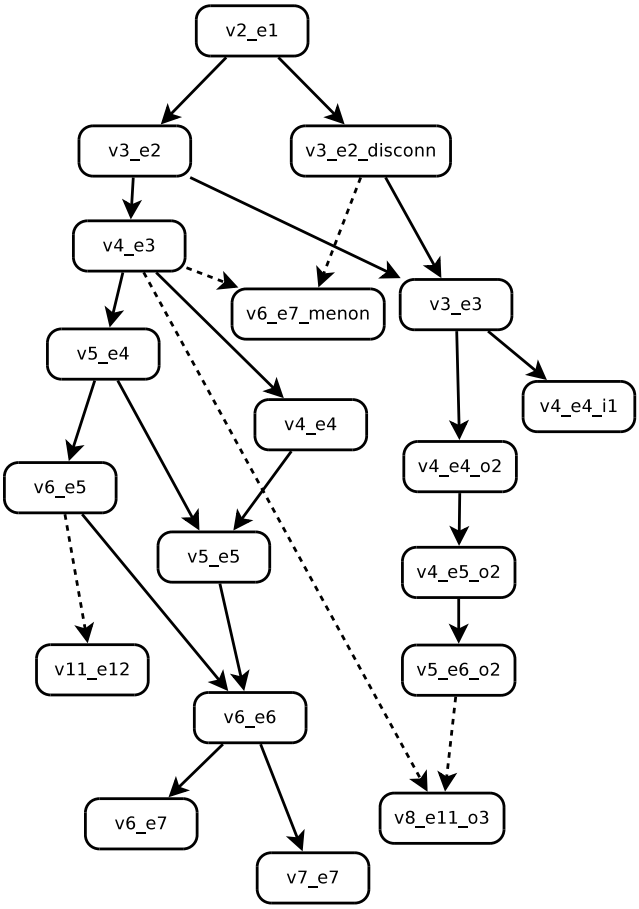


Figure 3.5: A tree of all topologies we use. For each line of descent topologies are shown connected to their closest ancestor(s), a solid line indicates the topologies are separated by the addition of a single edge or state while a dashed line indicates intermediate ancestors are missing and multiple edges or states were added. Topology nomenclature is primarily based on characteristics of the topology, v4_e3 indicates a topology with 4 **v**ertices and 3 **e**dges, 5v_e6_o2 has 5 **v**ertices, 6 **e**dges, and 2 **o**pen states. When this is insufficient to differentiate from other topologies we add a suffix based on other model features.

Figure 3.10: v6_e5

After the linear models we have a series of models that explore the simpler topologies. The topology v3_e2_disconn is another possible extension of v2_e1, the suffix 'disconn' signifies that its two closed states are disconnected and differentiates it from the other three vertex two state topology. The v3_e3 topology is a complete 3 state topology while v4_e4 and v4_e4_i1 are both extensions of the v3_e3 topology, the 'i1' on v4_e4_i1 distinguishes it from v4_e4 and simply signified that the transition from O1 to C3 is indirect as opposed to the direct transitions of its immediate ancestor v3_e3.

Figure 3.11: v3_e2_disconn

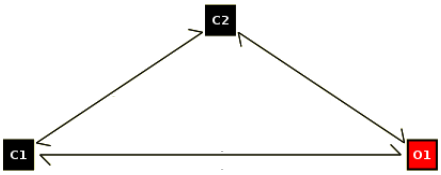


Figure 3.12: v3_e3

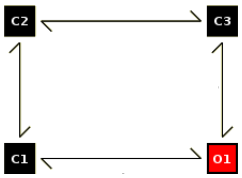


Figure 3.13: v4_e4

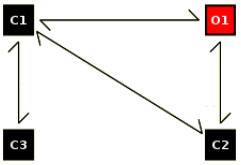


Figure 3.14: v4_e4_i1

We then have a series of topologies with multiple open states in Figures 3.15, 3.16, 3.17,and 3.18. Models with multiple open states have been used by Magleby and Pallota [25] to describe a voltage-gated K^+ channels that were sensitive to Cl^-. And both Qin et

al. [33, 34] and Milescu et al. [28] fit data to models with multiple open states. The v8_e11 architecture in Figure 3.18 was not drawn from literature but was simply an attempt to supply an additional topology with a large number of states and connections.

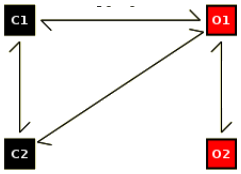


Figure 3.15: v4_e4_o2

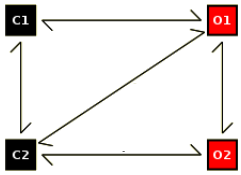


Figure 3.16: v4_e5_o2

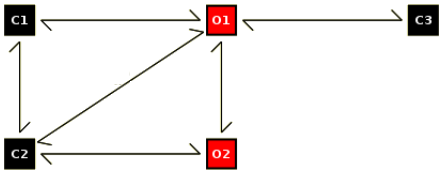


Figure 3.17: v5_e6_o2

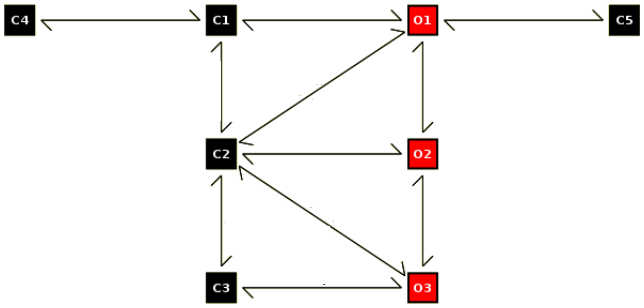


Figure 3.18: v8_e11_o3

The final series of topologies from Figure 3.19 to 3.24 extend the lower complexity topologies. The topologies v5_e5 and v6_e6 help fill the gap up to v6_e7 and v7_e7 which were both used by Gurkiewicz et al. *[16]*. The final topology v11_e12 is a high depth high complexity topology used for Kv4.3 channels [45].

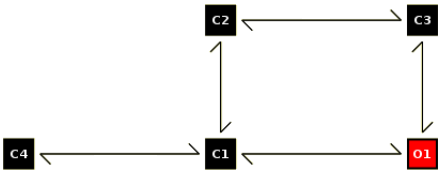


Figure 3.19: v5_e5

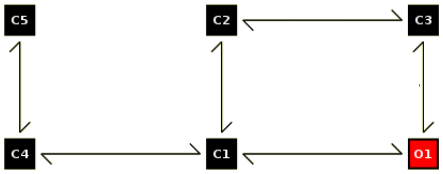


Figure 3.20: v6_e6

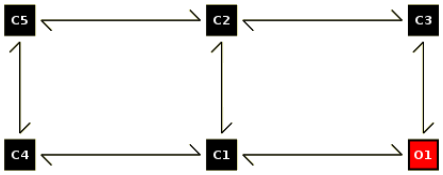


Figure 3.21: v6_e7

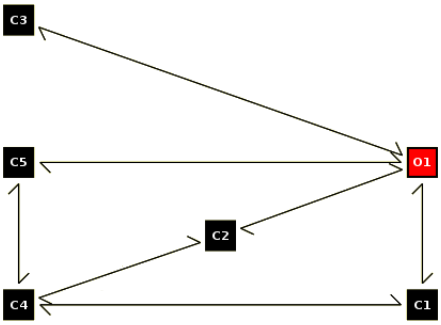


Figure 3.22: v6_e7_menon

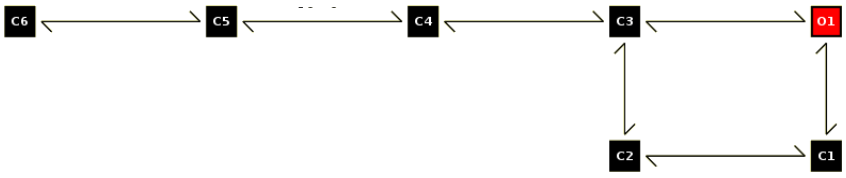


Figure 3.23: v7_e7

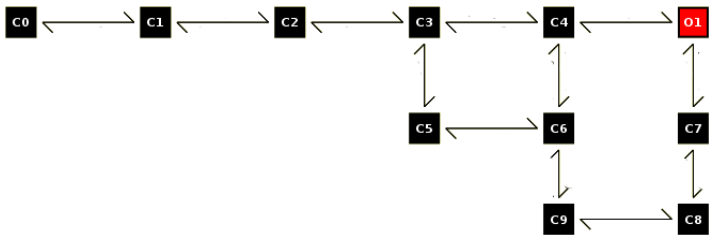


Figure 3.24: v11_e12

Several of these models have non-trivial isomorphisms including v3_e3, v3_e2_disconn, v4_e4_o2, and v6_e7_menon. The suffix menon on v6_e7_menon is because the topology is the one generated by Menon's genetic algorithm [26]

3.2.3 Topology Configuration

A model consists of three components, a topology, a configuration, and a parameterization. This section details how we combined data analysis and physiologist knowledge to create our configurations. For the model used by QUB Express there are several configuration properties including reversal potential, number of channels, baseline current and variance, individual channel conductance and variance.

The four parameters that factor directly into the calculation of the current: reversal potential, number of channels, baseline current, and conductance per channel, are all critical for the model to reproduce the data. The baseline variance and channel variance do not factor directly into the current calculation but are important to calculating the LL which is an important factor in the performance of the optimizer. Channel variance can only be accurately determined from single channel recordings, after consultation with Patrick Stemkowski we decided the QUB Express defaults of 1 pA were appropriate. The baseline variance depended largely on the characteristics of the individual experiment. Fortunately in each experiment the end of the deactivation pulse consisted of a long section of near zero current. The only source of noise in this section would be baseline noise, therefore we calculated the baseline variance as the variance of the current in this section.

Since different cells are used for the control and IL experiments the number of K+ channels will vary. Unfortunately we cannot distinguish between changes in conductance due to the total number of channels and changes due to the effects of IL. As such we calculate the number of channels from the control recording, where the relationship between peak conductance and total conductance is better understood, and use the same number for the

IL recordings.

For our data the reversal potential was calculated by Pat Stemkowski using the Nernst equation. For channel conductance and variance we decided to use the QUB Express defaults which were judged to be sufficiently close to the actual values for the channels.

To obtain the number of channels required to model each recording we used a combination of data analysis and expert knowledge. Based on feedback from the researcher who produced the data, it was assumed that during the control recordings peak activation would mean almost all channels would be in an open state, therefore we could calculate the total number of channels by finding the peak conductance.

Conductance is calculated as $g = \frac{I}{V_m - V_{rev}}$, where V_m and I are drawn directly from the recording and V_{rev} is supplied by the physiologist from knowledge of the experimental setup. Close to the reversal potential the formula becomes unreliable since the denominator approaches zero when V_m approaches V_{rev}. Thus we calculated the peak conductance using data points far from the reversal potential and pulse transitions where data irregularities may be expected. Note that while QUB Express does not enable optimization of V_{rev} it can in principal be fit like any other parameter. However, the small currents near the reversal potential make it difficult to accurately discern from the data. Physiologist derived values are considered far more accurate.

To determine the number of channels from the peak conductance we considered a pure algorithmic approach, but settled on a user guided selection as more reliable. Our approach divided the recording into ranges of 10 mV, for instance 45-55 mV and 55-65 mV. Then for each range we evaluated every data point that fell in that range and selected the point that showed the highest conductance, this generally corresponded to the point of highest current for each sweep. We specifically excluded the 10 mV region centred around the reversal potential as being too unreliable to accurately calculate conductance. For more distant points we presented them for visual inspection relying on the user to evaluate their reliability. Once the peak conductance points were selected we calculated the number of channels based on a user supplied estimation of peak activation probability and the per channel conductance. With our data we used a figure assuming 95% of the channels were open at peak conductance for the control recordings of both the *A-Type* and *Delayed Rectifier* channels.

The number of channels can be directly calculated from the noise[1], however a reliable estimate may require a hundred sweeps or more. As well the optimizer may attempt to fit

the number of channels such as any other parameter, but the results can vary by an order of magnitude or more. A researcher based estimate of the peak open probability is a more accurate means to determine the number of channels.

The final configurable property was the baseline current and variance. Baseline current is a steady current present throughout the trace from a source other than the channels in question. This assumption may be false as it assumes the baseline current remains constant throughout the recording, perhaps originating from an instrument calibration error. However, if the baseline current does vary with voltage then we know neither the reversal potential nor how conductance varies with the voltage. By using a constant baseline current of unknown origin we assume a possible source of systemic error if the current is not constant with voltage. Our control *Delayed Rectifier* recordings required a slight baseline current to accommodate a small negative current of unknown origin during the deactivation pulse.

Baseline variance was present in all recordings, it is characterized as the background noise that exists independent of the ion channels and likely comes from background currents or instrument noise. We calculated baseline variance from a section of data near the end of the trace where the channels would show little or no activation.

Note QUB Express can also include channel conductances and variances as optimization parameters as information about both is contained in the channel noise. However, similarly to the optimizations of the number of channels, this would lead to huge variances as there was not enough data to extract reliable estimates. Instead expert knowledge was a more reliable method to determine these values.

3.3 Determining Optimizer Settings

Due to the fact we intend to perform a large number of optimizations without the benefit of user feedback it is prudent to determine a well performing optimizer profile for the problem.

With this objective we performed a series of experiments to evaluate the performance of optimizer profiles against an artificial dataset.

QUB-Express includes three optimization algorithms with multiple settings for each algorithm. The standard optimization includes an initial pass by the Nelder-Mead simplex method (simplex) [32] followed by an application of the Davidon-Fletcher-Powell (DFP) formula [10] or the recently added Broyden–Fletcher–Goldfarb–Shanno (BFGS) [4] algorithm.

Following the prescribed procedure we tested two arrangements, Simplex followed by

Algorithm	Default	Long
Simplex	100 iterations	1000 iterations
DFP	0 restarts	4 random restarts
BFGS	100 iterations	1000 iterations

Table 3.1: Test parameter for each optimization algorithm along with the default and long setting for the parameter

Profile	Simplex	DFP	BFGS
DS+DD	D	D	-
DS+LD	D	L	-
LS+DD	L	D	-
LS+LD	L	L	-
DS+DB	D	-	D
DS+LB	D	-	L
LS+DB	L	-	D
LS+LB	L	-	L

Table 3.2: The different combinations of QUB-Express optimizer settings we tried. A **-** indicates the algorithm was disabled, **D** indicates the algorithm was enabled with default settings and L indicates the long setting.

DFP and Simplex followed by BFGS. Also for each algorithm we tested two different settings, default and long. The default setting uses the default parameters while the long setting increases a single parameter to extend the run time of that optimization.

The list of the test parameters is shown in table 3.1 while table 3.2 shows the eight resulting test profiles. For the simplex and BFGS optimizers we increased the number of iterations by an order of magnitude while for the DFP we followed a QUB Express developer recommendation to use 4 semi-randomized restarts.

The data set we used to test the profiles was the same recording we generated in section 4.1.1 using the v6_e5 model from figure 4.6a.

3.3.1 Modeling the Model Distribution

Since we are generating multiple fittings of models the most successful profile is not necessarily the profile that generates the most high quality models, but rather the profile that generates peak LL models at the highest rate. The time an optimization takes to complete is primarily determined number of function calls it makes to evaluate specific model parameterizations. We calculate the speed of the optimizer in terms of function calls so if ll_{peak} is lower bound of the peak LL range then the mean number of function calls required to generate a peak LL model is $Gen(ll_{peak})$. Note that all values depend on the optimizer

profile, configured topology, and recording. In the interest of brevity we exclude them from the notation. $Gen(ll_{peak})$ is a function of the mean number of function calls required to generated a model for the profile, N_{call}, and the probability that an individual model generated by the optimizer is better than ll_{peak}.

$$Gen(ll_{peak}) = N_{call} \times P(ll > ll_{peak}) \tag{3.1}$$

Even at 1000 trials per profile peak LL models were very infrequent. Two methods were used to expand our analysis. First in addition to our standard cutoff of 5% we included a stricter cutoff of 1% and a softer cutoff of 20%. We then tested the statistical significance by deriving the mean and variance of the generation rate with a jackknife estimator [12].

Our second method involved building a statistical model of the distribution of fittings.

Figure 3.25 shows that most LL values are above -1e+05 but a small clump persists below -1e+06, we refer to these two groups as **functional** and **degenerate** models respectively.

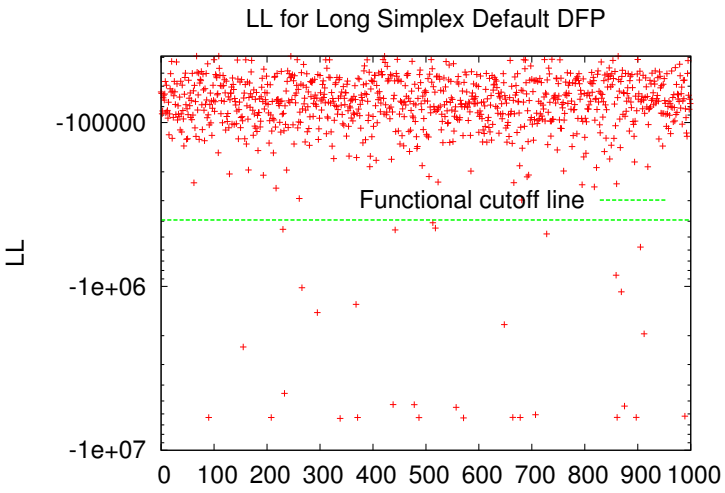
The **degenerate** models are instances where the optimization procedure has failed completely. A subset of these are numerically unstable where the equations in sections 2.6 and 2.8 break down. Many other models are stable numerically but do not display subtle dynamics, often stuck completely open or closed regardless of voltage.

The **functional** group consists of models where the simulation is numerically stable and the models are able to react to the voltage by showing partial activation, they range from the best fits to models that show very little relation to the data. There are ambiguous cases but for our calculations any model above a cutoff of 10 times ll_{best} (approximately -4e+05) is considered a functional model, though adopting any figure between -1e5 and -1e6 made very little difference to our results.

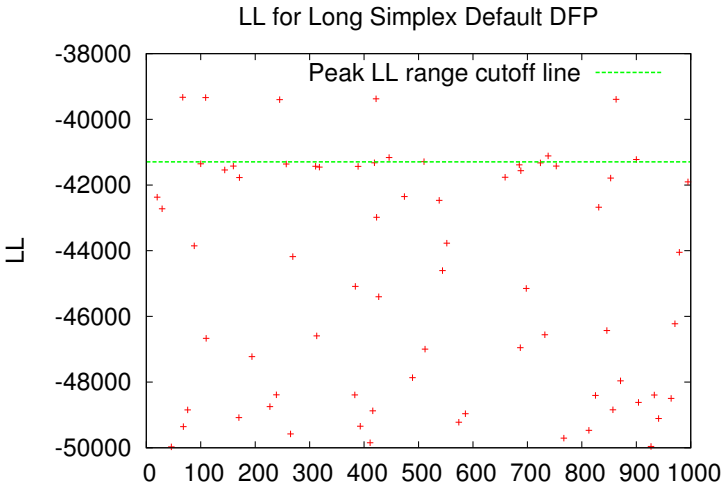
The optimization breakdown in the degenerate group meant that they followed a different distribution from the functional models. As such we separated them so we could study the distribution of the functional models separately. We denote the probability that a generated model is functional as P_{func}, and the probability that a functional model is in the peak LL range as $P(peak|func)$.

$$P(ll > ll_{peak}) = P_{func} \times P(peak|func) \tag{3.2}$$

We attempted several statistical distributions to approximate $P(peak|func)$ though none performed well. The Log-normal and gamma distributions came closest with the Log-normal



(a)



(b)

Figure 3.25: The LL of fittings from the LS+DD profile. The x-axis simply shows the number of fittings and displays them in the order in which they were generated. In 3.25a the functional models look roughly normal (under a logarithmic scale) but the degenerate models below the functional cutoff line show very little structure. In 3.25b we see the very small proportion of models that reach the peak LL range.

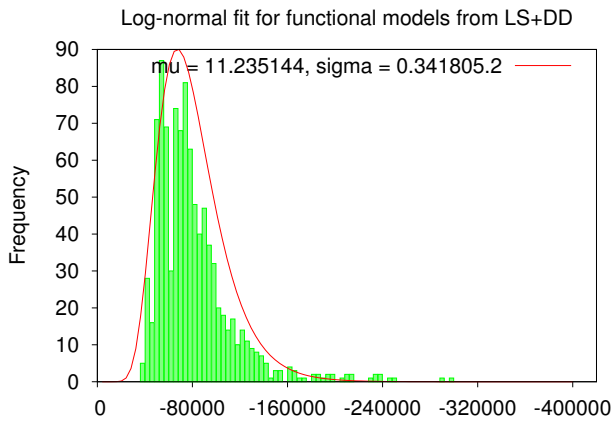


Figure 3.26: The Log-normal approximation captures the general form of the distribution but over-estimates the probability of high quality fits.

showing slightly superior performance but still significantly overestimating the probability of the peak results. Figure 3.26 shows the Log-normal approximation of the histogram from one set of fittings. The probability of a peak model can be approximated by the cumulative distribution function of a Log-normal distribution with mean μ and standard deviation σ giving the following formula:

$$P(peak|func) = F_{ll}(ll_{peak}; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \int_0^{-ll_{peak}} \frac{e^{\frac{-(\ln(t)-\mu)^2}{2\sigma^2}}}{t} dt \tag{3.3}$$

Despite the overestimation we felt it was a useful analysis since it included the LL values that did not reach the peak LL cutoff. However, due to the overestimation statistical significance comparisons [47] would be inappropriate.

3.3.2 Estimating the Efficiency of a profile

Table 3.3a shows the results of the optimization trials measured in observed probabilities and estimated probabilities using the LogNorm projection. One thing quickly apparent is the very poor performance of the BFGS optimizer, in all four profiles only two models came within 20% of the best result and none reached the 5% cutoff. The authors of QUB-Express wrote their own implementation of DFP which included slight modifications [35] but used the simplex and BFGS implementations from the SciPy library [22], so the difference may reflect adaptions made to the implemented DFP version rather than a difference in the formal description of the algorithms.

Another apparent observation is the degree to which the LogNormal projection overesti-

Profile	# Func / 1000	N_{call}	$P(peak\|func)$ Observed 1% / 5% / 20%	$P(peak\|func)$ LogNorm 1% / 5% / 20%
DS+DD	944	3196 ± 481	0.64 / 1.38 / 3.92	2.87 / 3.53 / 6.85
DS+LD	928	17084 ± 1892	**1.29** / **2.80** / 6.90	**4.59** / **5.54** / 10.03
LS+DD	967	3404 ± 301	0.52 / 0.83 / 4.76	2.92 / 3.77 / 8.26
LS+LD	963	16892 ± 1958	1.04 / **2.80** / **8.62**	3.40 / 4.47 / **10.27**
DS+DB	962	2157 ± 642	0.00 / 0.00 / 0.00	0.63 / 0.81 / 1.82
DS+LB	962	5652 ± 5958	0.00 / 0.00 / 0.10	0.80 / 1.02 / 2.16
LS+DB	905	$\mathbf{1643 \pm 971}$	0.00 / 0.00 / 0.11	0.51 / 0.69 / 1.88
LS+LB	862	3256 ± 4200	0.00 / 0.00 / 0.00	0.68 / 0.89 / 2.18

(a) Results of the optimization trials. N_{call} is the number of function calls per fitting and $P(peak|func)$ gives the probability of a functional model reaching the given cutoff

Profile	# Gen for 1%	# Gen for 5%	# Gen for 20%
DS+DD	$\mathbf{5.33e+05 \pm 1.16e+04}$	$\mathbf{2.46e+05 \pm 5.34e+03}$	$8.64e+04 \pm 1.87e+03$
DS+LD	$1.42e+06 \pm 2.27e+04$	$6.57e+05 \pm 1.05e+04$	$2.67e+05 \pm 4.26e+03$
LS+DD	$6.81e+05 \pm 8.69e+03$	$4.25e+05 \pm 5.43e+03$	$\mathbf{7.40e+04 \pm 9.44e+02}$
LS+LD	$1.69e+06 \pm 2.82e+04$	$6.26e+05 \pm 1.05e+04$	$2.04e+05 \pm 3.40e+03$
DS+DB	-	-	-
DS+LB	-	-	$5.65e+06 \pm 8.59e+05$
LS+DB	-	-	$1.64e+06 \pm 1.40e+05$
LS+LB	-	-	-

(b) Calculated generation rates using the measured probabilities

Table 3.3: Results for the different optimizer profiles. The best result for each category is indicated in **bold**.

mated the probability of a good fit. However, the LogNormal created very similar rankings to the observed probabilities, and consistently ranked the long DFP and BFGS configurations as superior to their respective default DFP and BFGS configurations. This was particularly useful for the BFGS configurations where there were not enough peak LL fittings to use a direct observation.

The long DFP configuration significantly improved the probability of peak LL results but at the cost of significantly increasing the number of function calls per model.

Table 3.3b shows the number of function calls required to generate a peak LL model under each profile as calculated using the observed probabilities. The two profiles using the default DFP setting outperformed any other profiles. As compared to the long DFP setting the default DFP setting did have a lower probability of generating peak LL models, but this was more than made up for by the much faster generation rate of the default DFP setting. Thus running both profiles for the same number of iterations one would expect more high quality models from the default DPF.

Based on these results the default simplex default DFP profile generated models in the peak LL range at the greatest rate.

The intention was to use the best optimizer profile when we started performing bulk optimizations as part of our full protocol. However, an early version of the experiment with only 100 fittings indicated the long Simplex default DFP profile was the top performer and so that was the profile we used for our experiments. The analysis scripts and results of the experiments are available for download[1].

3.4 Optimization and Analysis

Once the data is prepared, the topology configuration set, and the optimizer profile chosen, it is time to start the optimization stage as directed through maximum deviation.

The optimization stage is designed around creating large numbers of fittings for each topology, this was motivated by the fitting difficulties encountered in section 3.3. For many problems a single fitting has a low probability of achieving a fit without expert intervention. To build a system appropriate for non-experts to use we developed a strategy of generating 10s or 100s of models per configured topology. With multiple fittings we attempt to increase the odds of finding models close to the true peak LL range for that configured topology as well as gain a clearer picture of model behaviour within the peak LL range.

We create fittings in a series of optimization rounds. At the conclusion of each optimization round we survey all topologies, select the ones that require further fittings, and double the number of fittings for that topology. This continues until we reach a termination criteria or hit a maximum number of fittings.

This process is initiated with an initialization round. The size of the initialization round is typically 100 fittings. It is important not to make the initial number of fittings too small since we use the best global fittings when evaluating the termination criteria. Significantly underestimating the global maxima could lead to us prematurely terminating the optimization process.

After the initialization round analysis and further rounds of fittings are guided by the maximum deviation.

3.4.1 Maximum Deviation

We developed maximum deviation as a technique to analyze the collections of models in the peak LL range. It is a measure of the maximum divergence in model behaviour a configured

[1]Results of optimization parameter experiments: http://hdl.handle.net/10402/era.38721

topology may express when properly fitted to a recording. We define the deviation as the degree to which two models responding to a specific voltage protocol show similar behaviour within their hidden states. The maximum deviation of a group of models is then defined as the largest deviation that can be found between any pair of models in the group.

We expect that underparameterized models will be highly constrained in how they obtain an optimal fit. This constrained behaviour will be evident in the hidden state behaviour and thus show a low max deviation. Similarly overparameterized models will be underconstrained and thus able to show a large max deviation. These ideas are consistent with the concepts of model bias and variance[15].

Ideally we would measure the max deviation of several models that all share the same LL, the global maximum for that configured topology. However, obtaining two fittings with the same LL is very rare and the difficultly of fitting can make it unclear if we have achieved the global maxima for a configured topology or are stuck in a local maxima. Instead we measure the max deviation among the models in the peak LL range.

Having a set of peak LL models we consider the behaviour to evaluate. The behaviour of the models in terms of open vs closed state occupancy is already well measured by the LL. If two models have a similar LL it is very likely for them to generate similar fits. Instead we wish to evaluate the unobserved behaviour. By unobserved behaviour we mean the occupancy of the hidden states of the HMM that cannot be distinguished by observing generated current.

If there are multiple open and closed states then all states are hidden and we simply measure the total difference in state occupancy and normalize by the number of data points. But many commonly used topologies only have a single open state, meaning they are only partially hidden as the occupancy of the open state can be inferred from the current. As such for a partially hidden model we calculate the deviation among only the hidden states and normalize by the total hidden state occupancy.

The formula is described in the following. Take models A and B to be two fittings of the same configured topology. The mean occupancy of a state for A is the same as the state probability vector for that model P_{At}, so the deviation of state n at time t is simply $\mid P_{At}(n) - P_{Bt}(n) \mid$. Over the entire set of states the cumulative difference of all unobserved states is:

$$\sum_{t=0}^{T}\sum_{n=0}^{N} \mid P_{At}(n) - P_{Bt}(n) \mid$$

Now we divide by the total amount of unobserved state occupancy over the duration of the recording. This gives the formula for the deviation in Equation 3.4.

$$\frac{\sum_{t=0}^{T}\sum_{n=0}^{N} \mid P_{At}(n) - P_{Bt}(n) \mid}{\sum_{t=0}^{T}\sum_{n=0}^{N} (P_{At}(n) + P_{Bt}(n))} \tag{3.4}$$

The max deviation of a set of models is produced by the pair of models that maximizes Eq 3.4

$$\arg\max_{A,B} \frac{\sum_{t=0}^{T}\sum_{n=0}^{N} \mid P_{At}(n) - P_{Bt}(n) \mid}{\sum_{t=0}^{T}\sum_{n=0}^{N} (P_{At}(n) + P_{Bt}(n))} \tag{3.5}$$

The normalization means that max deviation values should fall in the range $[0, 1]$ though to accommodate models with non-trivial isomorphisms we calculate the deviation of each isomorphism and take the smallest deviation as the deviation for that model pair. This means that for certain classes of isomorphic topologies such as the v3_e2_disconn and v3_e3 topologies shown in figures 3.11 and 3.12 the range of max deviation is restricted to [0, 0.5] as a max deviation above 0.5 means the other isomorphism should be used. Also strictly speaking we should not apply max deviation to the v2_e1 topology in figure 3.6 since it effectively has no hidden states. However, we still apply the measure to the closed state occupancy with the knowledge that the max deviation is very unlikely to rise much above 0 since v2_e1 is the base of our topology tree.

Also note that any model from an ancestor topology can be a member of its descendant topology by adding any new states and edges with transition rates low enough not to alter the LL. Thus when given a set of models from an ancestor topology those models can be included by creating fake states with $P_{At}(n) = 0$ for any state n that does not exist in the ancestor topology[2].

Another important note is that while the theoretical range of the max deviation should be $[0, 1]$ simulation error means this is not strictly true. For reasons covered in section 7.5 the calculations to generate P are imprecise and the sum of P_t is not necessarily 1 nor are all $P_t(n)$ in the strict range $[0, 1]$. Filtering out all models that violate one of these conditions

[2]We did not have sufficient time to fully implement this aspect of the calculation, however, we did include a simplified version that took the max deviation from the set of ancestor models within the peak LL range of the descendant.

would greatly reduce the number of peak LL range models to little benefit since the errors do not significantly affect the LL calculation. Instead of demanding flawless models we cap the size of the errors by filtering out models where P contains an element P_t where the sum of all state probabilities is outside the range $[0.99, 1]$. Still the existence of these states with a slight negative probability means that calculated max deviations can occasionally be slightly greater than 1.

Note that the calculated maximum deviation is generally an estimate, and usually an underestimate, of the true max deviation of models in the peak LL range. The peak LL will generally increase as we find additional peak LL models that show a higher divergence than the current set, however, it may decrease if one of the new models has a higher LL and raises the lower bound of the peak LL range disqualifying some previous included models.

By calculating the max deviation of a large group of peak LL models we can gain information about the greatest degree of dissimilarity the configured topology can show when reproducing the data. In general a set of models showing a high degree of dissimilarity implies that they may have more complexity than is necessary to reproduce the data. However, exceptions may occur if an underparameterized topology is able to achieve a similar fit with more than one constrained solution, in which case maximum deviation would falsely indicate underparameterized models to be overparameterized.

3.4.2 Maximum Deviation applied to topology selection

Our basic assumption around max deviation is that the true max deviation of a configured topology will correlate to the ability of the configured topology to fit the data. Namely the higher the max deviation the more freedom a model will have in fitting the data. Thus a low max deviation is consistent with underparameterization and a high max deviation is consistent with overparameterization. On this basis we design a protocol that directs which topologies to explore and produces a selection of candidate topologies.

To understand the rules of our protocol it is necessary to understand the implications of our assumptions about max deviation and model complexity. The basic intuition is that an overparameterized topology has many potential behaviours that can reproduce the given data, leading to a high max deviation. By contrast an underparameterized topology is highly constrained by the data and will generally only have one behaviour that is able to reproduce results in the peak LL range, this would lead to a low max deviation. This is not

the same as saying a topology with a low max deviation is underparameterized or high max deviation overparameterized. A topology with the correct parameterization could show any level of max deviation, but we would expect its ancestors to have a low max deviation and its descendants to be high.

In general we consider a low max deviation 'Low' as [0, 0.25], 'Medium' as (0.25,0.6) and high as [0.6,1]. These values are largely best practice estimates and do not take into account factors such as model complexity and data noise that a more sophisticated measure may consider. This enables us to develop a rule to select a group of topologies that potentially include the correct parameterization. As well we create a set of rules designed to direct future optimization rounds and select topologies that would benefit from additional fittings.

3.4.2.1 Rule for selecting the candidate range

The difficulty of the optimization problem means we do not have sufficient information to select the correct topology among all the possible topologies. Even along a single branch of the topology tree from figure 3.5 it may be unclear if there is a significant gap in the global maxima between the parent topology and a child. Since we have no basis on which to select the correct topology we instead attempt to estimate the candidate range described in section 2.1.5. Specifically we select a range of topologies that attempts to include the correct topology.

The set of candidate topologies includes every topology in the global peak LL range that does not have an ancestor that is both in the global peak LL range and has a high max deviation. Alternately along any branch of the topology tree the set of candidate topologies will start with the first topology to enter the global peak LL range and end with the first topology to enter the global peak LL range and have a high max deviation.

Consider the set of linear topologies, v2_e1 through v6_e5, from figure 3.5. Assume that starting with v3_e2 all topologies are in the global peak LL range (any model descended from v3_e2 can do at least as well as v3_e2). The max deviation of v3_e2 is low, v4_e3 is medium, and v5_e4 and v6_e5 are both high.

The group of selected topologies would include v3_e2, v4_e3, and v5_e4. We would discount v2_e1 as being unable to fit the data and v6_e5 as being overparameterized.

This procedure would be repeated for every branch of the tree, including any other descendants of v3_e2. In general any of the selected topologies may be a correct topology and the researcher may employ other criteria to further distinguish.

3.4.2.2 Rules for additional optimizations

After the initialization round we continue to perform additional optimization rounds with the objective of improving the accuracy and precision of our estimate of the candidate range.

As a general rule we attempt to obtain 10 peak LL fittings to make a determination. This number is chosen as a compromise between having enough peak LL models to get a good estimate of the true max deviation and choosing a target that can be feasibly generated. In some cases where we suspect we are not seeing the true peak LL of the model we may choose a higher cutoff such as 20. For instance if we see a high or medium max deviation in a topology that has failed to reach the global peak LL we take that as an indication that the topology has not reached its true peak LL and we attempt more fittings with the objective of obtaining a better fit.

There are also some cases when fewer than 10 fittings are necessary, in particular when we have achieved the global peak LL and have a high max deviation. In this scenario there is nothing else that additional fittings could tell us.

This situation becomes quite common due to the fact that any descendant model can reach the same peak LL as its ancestor, but can also meet or exceed the max deviation of its ancestor at that LL. Therefore the moment a topology reaches the global peak LL and a high max deviation there is no need to perform additional fittings for that topology or any of its descendants. The assumption is that since we are seeing a high max deviation the model is already able to capture all the complexity in the data and any further improvements in LL would be marginal increments from capturing noise. This is also consistent with our topology selection rule as we would consider those descendant topologies to be overparameterized anyways.

In table 3.4 we go into detail for each possible scenario. It outlines the possible combinations of max deviation, number of peak LL fittings, and whether additional fittings are required for that scenario. It should be noted that there are several assumptions we make in the construction of this table.

First we assume that we have correctly identified the global peak LL range, and even if further fittings can incrementally shift the global peak LL range this will not change other topologies inclusion in the global peak LL range. Clearly this is not always the case but we cannot continue to search for better peak global LL indefinitely.

Secondly we assume that 10 peak LL models is sufficient to assume that the measured

Max Dev	Global Peak LL	# Peak LLs	More Fittings?	Reasons
Low	No	< 10	Yes	Insufficient Results
Low	No	≥ 10	No	Sufficient Results
Low	Yes	< 10	Yes	Insufficient Results Max deviation may increase
Low	Yes	≥ 10	No	Sufficient Results
Medium	No	< 20	Yes	Unlikely Scenario LL may increase
Medium	No	≥ 20	No	Unlikely Scenario
Medium	Yes	< 10	Yes	Insufficient Results Max deviation may increase
Medium	Yes	≥ 10	No	Sufficient Results
High	No	< 20	Yes	Unlikely Scenario LL may increase
High	No	≥ 20	No	Unlikely Scenario
High	Yes	< 10	No	Additional fittings can not increase max dev or LL
High	Yes	≥ 10	No	Additional fittings can not increase max dev or LL

Table 3.4: Various scenarios pertaining to the max deviation of a topology, relation of the peak LL to the global peak LL, and number of models in the peak LL range. The "Max Dev" column refers to the size of the max deviation, this includes the max deviation measured directly from the models and the max deviation implied from ancestors sharing the peak LL range. We define 'Low' as [0, 0.25], 'Medium' as (0.25,0.6) and high as [0.6,1] but these values are best practices. In the 'Global Peak LL' column a 'Poor' peak LL means the configured topology did not reach the global peak LL range, while a 'Good' peak LL did. The column '# Peak LLs' refers to number of models in the peak LL range for that topology and whether they have reached the described number. The test number in question is generally 10 but rises to 20 in the case of unlikely scenarios. The "Reasons" column give the interpretation of the given combination of characteristics. The 'More Fittings?' column advises whether additional optimizations are required. In some cases it is not necessary to calculate more results because we already have sufficient models to arrive at a likely conclusion.

peak LL range overlaps with the true peak LL range as determined by the globally optimal fit for each topology. Furthermore we assume that the 10 models in the measured peak LL range would still be included in the true peak LL range and that the models cover enough of the solution space that the measured peak LL is in the same category (low, medium, or high) as the true max deviation. The 10 model standard was chosen as a number low enough to be feasible but high enough to generate a reasonable number of fits. Choosing a slightly higher or lower number may be appropriate depending on the number of topologies, desired accuracy of results, and computational constraints.

The second assumption is the one most likely to be invalidated. We can never be certain how far we are from the globally optimal fit, and even when we do have that fit we cannot estimate how much of the possible range of solutions we are missing.

The main risk of the protocol is a scenario where the initialization round fails to reach the true global peak LL range and the sub-optimally fit models that comprise the measured peak LL range have a high max deviation. In that scenario the protocol will terminate prematurely without finding the true global peak LL range. A modification to the protocol that could help avoid this scenario is to more thoroughly validate the topology with a global peak LL and high max deviation that caused the protocol to terminate. We would take the first topology in the global peak LL range with a high max deviation (which is also the last model in the candidate topology range) and ensure that model had 10 peak LL range fittings before we terminated.

3.4.2.3 Max Deviation Results Table

To display the results of our approach we use tables such at those shown in table 3.5. The first table shows the initial optimization round, in this case comprising of 100 fittings, while the second table shows the complete results, where fittings have continued until each configured topology has reached a termination criteria. To make visual identification of different features more apparent we regularly make use of emphasis and bold font, to aid in recognition emphasized values are shown in *blue* and bolded values are shown in **red**.

The tables consist of six columns. In column one 'Topology' the names of the topologies are shown. Any configured topology that part of the candidate range is shown in **bold**. In this case in the first table v4_e3 is bolded. As a consequence of the selection rule the first topology in any lineage to enter the global peak LL range will be selected. Since v4_e3 does not have a high max deviation its descendants v5_e4 and v6_e5 are also selected and

we consider all three to be potential matches. In the table of full fittings v5_e4 has a high max deviation, as a result v6_e5 is no longer selected as a potential candidate.

A model that is selected for additional optimizations is indicated with a '*'. In the first round of fittings v5_e4 and v6_e5 are both shown with a '*' since with a medium max deviation and less than 10 peak LL models they both qualify for additional fittings.

The column second column, "# peak LL", shows the number of models in the peak LL range of the configured topology and the total number of fittings. In the initial table the simplest v2_e1 configured topology had 100 out of 100 models reach the peak LL range, but for v6_e5 only 2 out of 100 models reached the range.

The third column "Best LL" shows the LL of the best model for each configured topology. The best LL out of all the models is shown in **bold**, while all configured topologies that reach the global peak LL range are shown *emphasized*. The observed LL is the LL of the best model for that configured topology, but when an ancestor model achieves a superior LL we refer to that LL as the implied LL and we show that LL in '( )'. Note that the peak LL range for the "# peak LL" column is calculated based on the observed LL.

The fourth and fifth columns show the AIC, and BIC of the best model. These are based on the LL of the best model (not the implied LL). The best AIC and BIC overall are shown in **bold**.

The final column is the max deviation. Again the max deviation of the observed models is shown but if there is a larger implied max deviation from ancestor models the larger implied max deviation is shown in '( )'. In the full table the implied max deviation for v6_e5 is taken from its ancestor v5_e4. The table is sorted by the max deviation.

To make the max deviation categories apparent low max deviations are *emphasized*, medium are shown in normal font, and large max deviations are **bolded**.

Potential Issues The major difficulty is finding enough peak LL models, depending on the problem the optimization procedure may have a very low success rate. It may not be feasible to generate enough peak LL models to make a useful determination about the configured topology.

Another issue is in the number of peak LL models necessary to determine a low or high max deviation. While the intent of max deviation is to reduce the role of the optimizer in biasing the results we still are still restricted to the models the optimizer gives us. If the solution space allows for multiple behaviours with only slight preference then models with

Results for example at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	-15000	30008	30035	*0.001*
v3_e2	100/100	-12000	24016	24070	*0.100*
v4_e3	91/100	*-10050*	20124	20205	*0.200*
***v5_e4**	5/100	**-10000**	**20032**	**20140**	0.500
***v6_e5**	2/100	*-11000 (-10000)*	22040	22176	**0.630**
Full results for example					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	-15000	30008	30035	*0.001*
v3_e2	100/100	-12000	24016	24070	*0.100*
v4_e3	91/100	*-10050*	20124	20205	*0.200*
v5_e4	10/200	*-10000*	**20032**	**20140**	**0.700**
v6_e5	6/200	**-9999**	20038	20174	**0.650 (0.700)**

Table 3.5: Example table of fittings subjected to max deviation analysis

high max deviation should show that readily. If however the various behaviours exist in distinct valleys in the solution space than the optimizer may be unable to find them, in this case not only do we not know if a low max deviation is the true result, but we have no way of estimating the probability that a future iteration will generate a contrary result.

One way of investigating this is also calculating the mean deviation between all models in the peak LL range. If the mean deviation is correlated to the max deviation it suggests that the different solutions exist along a continuum and a low max deviation may be interpreted with low confidence. If they are uncorrelated then it suggests the solution space is disconnected and solutions creating a high max deviation may be found unpredictably.

Another issue is as models grow more complex there will be a natural tendency for model deviations to grow regardless of parameterization. This tendency is difficult to measure or correct since the absolute range of max deviation will still be [0,1] for non-isomorphic models.

3.4.3 Implementation of Protocol

We implemented the described protocol in QUB Express using a combination of a plugin, scripts, and modifications to existing QUB Express plugins.

The modification in question took the form of a patch to the MacRates plugin which is responsible for running the Hidden Markov Models and calculating the LogLikelihood as well as providing much of the optimization infrastructure. The plugin design was such that it could not record and store the state occupancy information without modification. Next was our QUB States plugin which added two panel interfaces to QUB Express. The first

was a data configuration panel which allowed the user to select the recording to fit and create a topology configuration as described in section 2.9. The panel had functionality to automatically scan the data to determine the background noise as well as a functionality to scan the recording for the proper number of channels.

To scan for the number of channels we required two pieces of user input, the reversal potential and the peak activation level (ratio of channels open at the maximum activation level). We then perform a scan of the recording in 10 mV increments and for each increment determine the point of channel with the highest conductance and calculate the number of channels necessary to deliver that conductance. We automatically exclude the 10 mV range centered around the reversal potential due to the unreliability of the conductance calculation in this region. For each 10 mV increment we show the highest conductance point to the user and allow them to select one from which to calculate the number of channels, or they can set the number of channels directly. This technique was effective for the recordings described in section 3.2.1 but other recordings with different artifacts may require other methods.

The actual optimization panel includes the list of topologies from section 3.2.2, configured using the settings from the previous panel, as well as information on the number of models fitted and the LL of the best model. It also includes the ability to select topologies and perform fittings until one reaches a predetermined number of fitted models. As well it can generate a number of different reports on the generated models.

The optimization process itself is run through the modified MacRates plugin, a standard plugin for macroscopic modeling and optimization shipped with QUB Express. Each configured topology is initialized with all k0 parameters set to 10 and k1 parameters set to 0. Immediately prior to optimization a randomization procedure is applied to the k0 values modifying them by a random number from [0,1]. During simulation each individual sweep is initialized with the state probability vector at steady state.

The plugin was designed before we developed the max deviation protocol and does not fully implement the protocol as described. Instead the protocol is implemented through a series of scripts we wrote to integrate with the plugin. They calculate the statistics for each model and implement the rules from table 3.4 to generate tables such as table 3.5 that provide guidance on suggested topologies and further fittings.

The plug-in, QUB Express patch, and max deviation scripts are available for download[3]. We have future plans to integrate the max deviation functionality into the plugin and then

[3]http://hdl.handle.net/10402/era.38722

submit the combined work into QUB Express.

Chapter 4

Experimental Design

We designed a series of experiments to analyze how well our predictions about max deviation held and to evaluate the utility of max deviation as a metric. These experiments consisted of five simulated recordings and four cell recordings.

The five simulated recordings were generated from linear models ranging in complexity from two to six states. These generative models were constructed to ensure occupancy of the terminal state furthest from the open state and given baseline noise levels at the same approximate level as the recorded data.

The cell recordings were obtained from Patrick Stemkowski[41] and are described in more detail in section 3.2.1.

Each experiment consists of two stages, first is a configuration stage where the recordings are analyzed and properly parameterized. The cell recordings all had a physiologist supplied reversal potential of -58 mV, this same value was used for all simulated data as well. For all recordings baseline noise levels were estimated from the recordings. For the live data the number of channels was estimated from the recordings based on physiological assumptions. For the simulated data we used the number of channels from the generative models (i.e. the simulated models that generated the data).

After configuring the models and using the optimizer configurations suggested in section 3.3 we performed a two stage process. Step one involved an initial fitting of either 50 or 100 models for each configured topology. Step two involved additional fittings to increase the number of peak LL models according to the rules laid out in table 3.4.

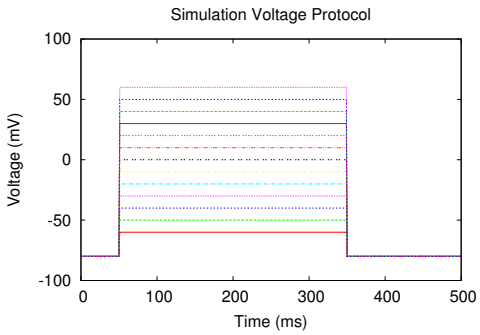


Figure 4.1: Voltage protocol used for the simulated data. The prepulse and deactivation pulse were both set to -80 mV and the main pulse ranged -60 to +60 mV in 10 mV increments.

4.1 Test Data

4.1.1 Simulated Data

The five models simulated are based on the five linear topologies v2_e1 through v6_e5 from Section 3.2.2. In addition to the noise from channel variance each model also had baseline noise with a standard deviation of 100 pA. Each model was subjected to the 13 sweep voltage protocol shown in Figure 4.1 which was again based on the protocols used in our experimental data. Simulations were done at a granularity of 0.1 ms matching that of the recorded data. One important difference between the simulated protocol and the experimental protocols is that the simulated protocol was a step function with instantaneous switches between pulses while the transitions in the experimental protocol were more gradual as show in section 3.2.1.

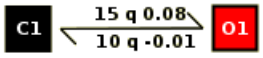
None of the simulated models were developed to exhibit inactivation.

The first simulated model is the trivial model using the v2_e1 topology in figure 4.2. The model rates are shown in the illustration. The k0 and k1 rates from C1 to O1 are 15 and 0.08 respectively while the k0 and k1 rates from O1 to C1 are 10 and -0.01.

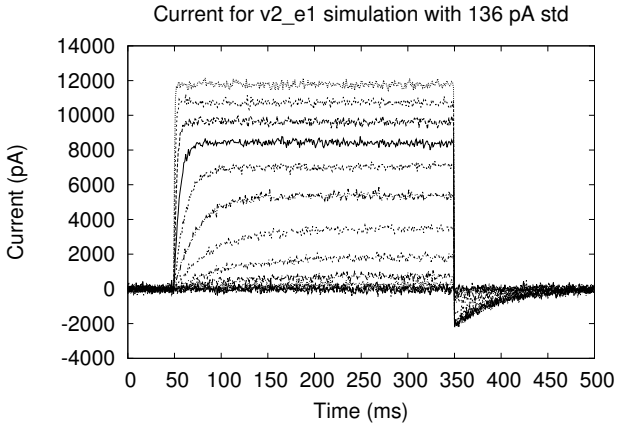
Next was **v3_e2** shown in Figure 4.3. This was the simplest model which displayed multiple levels of state occupancy.

We constructed the simulated models to ensure that all states in the models were used. However, we did not ensure that the models we generated used the full expressiveness of their topology. For instance, a v5_e4 model could possibly reproduce the recording produced by the v6_e5 model in figure 4.6.

In table 4.1 we show the configurations used for the simulated experiments. Baseline noise was measured from the generated data but the channel count was taken directly from



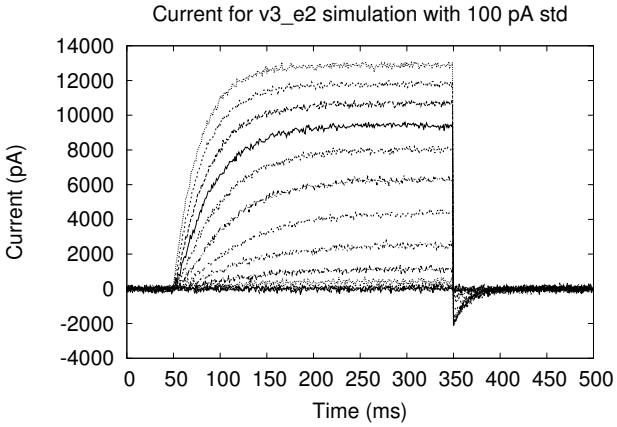
(a) Rate parameters between states are shown as 'k0 q k1'



(b)

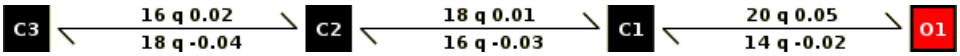
Figure 4.2: Simulated v2_e1 model and recording.

(a) Rate parameters between states are shown as 'k0 q k1'

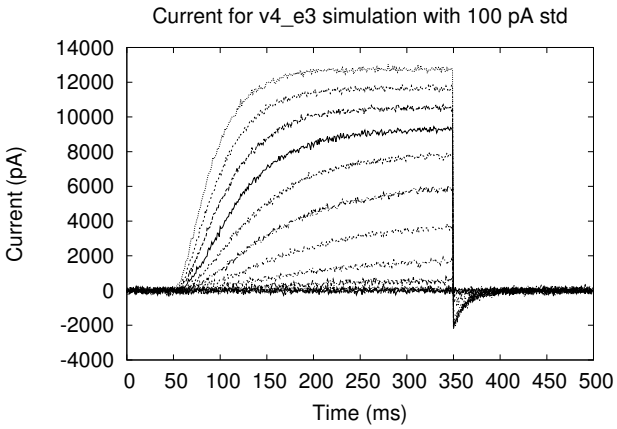


(b)

Figure 4.3: Simulated v3_e2 model and recording.



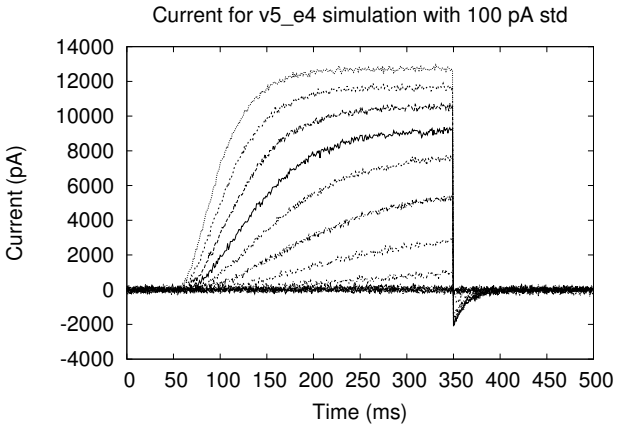
(a) Rate parameters between states are shown as 'k0 q k1'



(b)

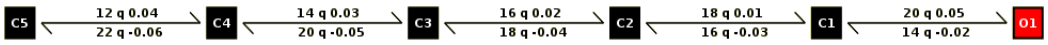
Figure 4.4: Simulated v4_e3 model and recording.

(a) Rate parameters between states are shown as 'k0 q k1'

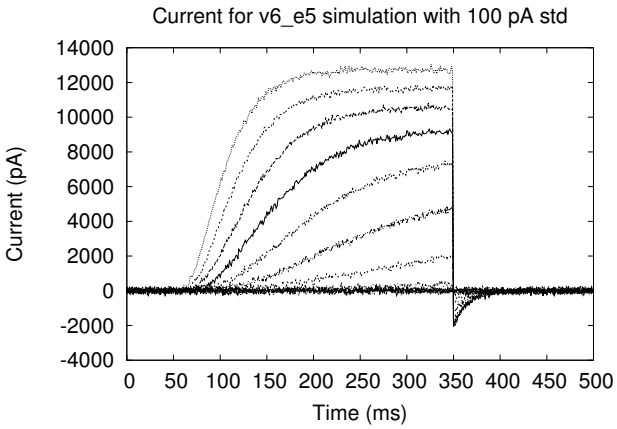


(b)

Figure 4.5: Simulated v5_e4 model and recording.



(a) Rate parameters between states are shown as 'k0 q k1'



(b)

Figure 4.6: Simulated v6_e5 model and recording.

Recording	vRev (mV)	# Channel	Baseline current (pA)	Channel Conductance (pS)
v2_e1	-58	10000	0 ± 92	10 ± 1
v3_e2	-58	10000	0 ± 99	10 ± 1
v4_e3	-58	10000	0 ± 94	10 ± 1
v5_e4	-58	10000	0 ± 97	10 ± 1
v6_e5	-58	10000	0 ± 92	10 ± 1

Table 4.1: Topology configuration for the five simulated recording experiments

the generative models.

4.1.2 Experimental Data

We had a set of four recordings of K+ channels made by Patrick Stemkowski, a control and Interleukin-1 beta (IL-1β) treated group of *A-Type* channels and a control and IL-1β treated group of *Delayed Rectifier* channels. A lengthier discussion of methods used to prepare the recordings and create the configurations is found in section 3.2. The voltage protocol was the original truncated voltage data shown in figure 3.1a, the current was unmodified except for the zeroed out conditioning sections as shown in figure 4.7. The *Delayed Rectifier* voltage is shown in figure 3.2a and the current traces are shown in figure 4.8. As detailed in the full analysis we included a baseline current to offset the negative current shown by the initial sweep near the reversal potential for the *Delayed Rectifier* control recording.

The configuration used for the four experiments is shown in Table 4.2.

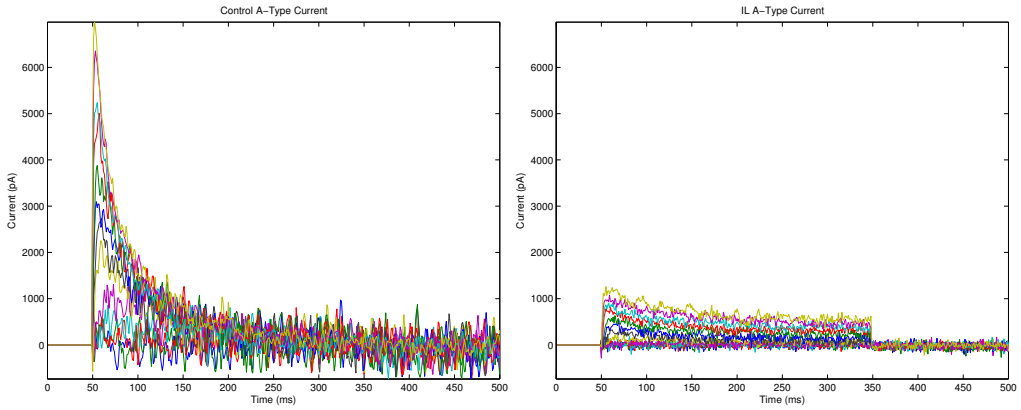


Figure 4.7: Current sweeps for *Control* and *IL A-Type* recordings used for fitting.

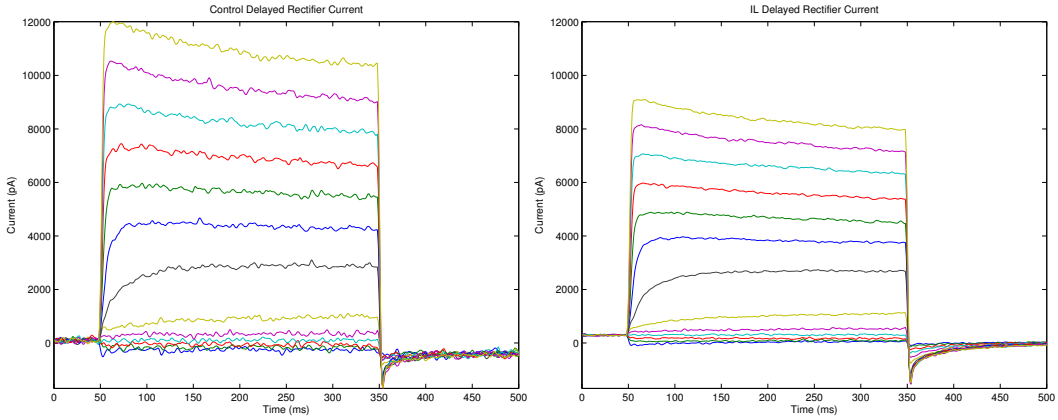


Figure 4.8: The unaltered recordings of the Delayed Rectifier currents under control conditions and IL-1β. Originally shown in figure 3.2b

Recording	vRev (mV)	# Channel	Baseline current (pA)	Channel Conductance (pS)
Control A-Type	-58	6753	0 ± 137	10 ± 1
IL A-Type	-58	6753	0 ± 52	10 ± 1
Control Delayed Rectifier	-58	11234	-257 ± 70	10 ± 1
IL Delayed Rectifier	-58	11090	0 ± 20	10 ± 1

Table 4.2: Topology configuration for the five experiments

4.2 Objectives

We have several questions we wish to investigate with our experiments. These relate both to the optimization protocol and the max deviation metric.

First we wish to evaluate the performance of our protocol as a whole.

R1: In the simulated experiments can we find fits with a LL as good as the original model?

R2: Will the candidate range for each simulated experiment contain the topology of the original model?

R3: How will the parameterization of the original model compare to the best fitted model of the original topology?

R4: Will LL/AIC/BIC identify the correct topology in the simulated data?

R5: Will we find good fits and a reasonable set of candidate topologies on the experimental data?

R6: For each channel type we would expect the Control and IL conditions to show similar behaviour (subject to the effects of IL). Therefore, will the channel type show a similar candidate topology ranges for each experimental condition?

Second we wish to investigate the interpretations in table 3.4.

R7: Is it true that topologies outside the global peak LL range will rarely have a high max deviation?

R8: Can we consistently find a high max deviation in topologies known to be overparameterized?

R9: And do the deviations in a group of peak LL models follow a predictable statistical distribution?

Chapter 5

Results

For each experiment we highlight several methods of model determination, we show the best model according to AIC, and BIC. As well we indicate the range of configured topologies considered to be potential minimal matches to the data according to our application of the max deviation.

In section 5.1 we examine the simulated data experiments and address questions R1 through R4, and R7. In section 5.2 we examine the experimental data and address questions R5 and R6. Finally in section 5.2.8 we address questions R8 and R9. We make the full set of results available for download[1].

5.1 Simulated Data

For each experiment involving the simulated data, in addition to our standard evaluation, we include a comparison of the parameters of the generative model with the parameters of the top model from the same configured topology.

The purpose of these experiments is to evaluate questions R1 through R4 as well as provide additional data for R7 through R9.

5.1.1 Simulated v2_e1 data

The first pass over the simulated data from the v2_e1 model is shown in table 5.1. On this simple data set the results are consistent with the predictions from table 3.4. Not only do all configured topologies reach the peak LL range but all except two generate a model with

[1] http://hdl.handle.net/10402/era.38736

a LL above -39576, the score of the model from figure 4.2 used to generate the data itself. The ability of even the v2_e1 topology to exceed the LL of the generative model is likely due to the effect of various noise sources in the simulation.

Three topologies have fewer than 10 peak LL models. However for two of these topologies a high max deviation, either measure directly or implied from ancestor topologies, precludes further fittings. Only the v11_e12 topology requires further fittings. The medium range max deviation of the linear models v3_e2 through v6_e5 is surprising as one would expect a high max deviation for these clearly overparameterized models. One possibility is that they are showing the true max deviation and the later states past C1 cannot be heavily utilized while achieving a peak LL range fit. Another is that a much higher max deviation is possible, but for such a simple recording any models near the peak LL range fall into the same general solution of a heavy C1 occupancy.

5.1.1.1 Comparison to generative model

Both AIC and BIC misidentify the correct model as coming from the v5_e6_o2 configured topology. According to the max deviation table the model is correctly shown to include the v2_e1 topology, however 12 models in the candidate range is quite imprecise for such a simple example.

Table 5.2 shows the percentage difference between the parameters of the original and fitted model and figures 5.2 and 5.3 show the top fitted v2_e1 model and the resulting fit.

K0	
C1→O1	O1→C1
2.3	3.4
K1	
C1→O1	O1→C1
2.1	-1.9

Table 5.2: Percentage difference in parameters between the best performing v2_e1 model and the generative model

5.1.2 Simulated v3_e2 model

Next we move on to the simulated recordings of the model from Figure 4.3. Table 5.3 shows the results of the fittings from both the initial and full run. This being a relatively simple data set the optimizer again generally performed well.

Again additional results are not required for a single configured topology. However, we

Results for v2_e1 simulated recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	*-39555*	79119	79146	0.0
v4_e3	71/100	*-39522 (-39520)*	79069	79150	0.37 (0.392)
v3_e2	92/100	*-39520*	79056	79110	0.392
v5_e4	60/100	*-39538 (-39520)*	79109	79217	0.295 (0.392)
v6_e5	57/100	*-39548 (-39520)*	79136	79271	0.317 (0.392)
***v11_e12**	1/100	*-43655 (-39520)*	87406	87731	- (0.392)
v3_e2_disconn	91/100	*-39514*	79044	79098	0.497
v3_e3	86/100	*-39480*	78984	79065	0.5
v4_e4_o2	98/100	*-39507 (-39480)*	79047	79156	0.636
v6_e7_menon	7/100	*-39534 (-39514)*	79124	79313	0.819
v4_e5_o2	93/100	*-39488 (-39480)*	79016	79151	0.869
v4_e4_i1	54/100	*-39506 (-39480)*	79045	79154	0.967
v6_e7	13/100	*-39641 (-39500)*	79338	79527	0.71 (0.969)
v4_e4	63/100	*-39505*	79043	79151	0.969
v5_e5	21/100	*-39500*	79040	79176	0.96 (0.969)
v6_e6	16/100	*-39537 (-39500)*	79122	79285	0.837 (0.969)
v7_e7	6/100	*-39432*	78920	79110	0.454 (0.969)
v8_e11	75/100	*-39451 (-39375)*	78990	79288	0.87 (0.993)
v5_e6_o2	80/100	***-39375***	**78798**	**78960**	0.993
Full results for v2_e1 simulated recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	*-39555*	79119	79146	0.0
v4_e3	71/100	*-39522 (-39520)*	79069	79150	0.37 (0.392)
v3_e2	92/100	*-39520*	79056	79110	0.392
v5_e4	60/100	*-39538 (-39520)*	79109	79217	0.295 (0.392)
v6_e5	57/100	*-39548 (-39520)*	79136	79271	0.317 (0.392)
v3_e2_disconn	91/100	*-39514*	79044	79098	0.497
v3_e3	86/100	*-39480*	78984	79065	0.5
v11_e12	5/400	*-40064 (-39520)*	80224	80549	0.615
v4_e4_o2	98/100	*-39507 (-39480)*	79047	79156	0.636
v6_e7_menon	7/100	*-39534 (-39514)*	79124	79313	0.819
v4_e5_o2	93/100	*-39488 (-39480)*	79016	79151	0.869
v4_e4_i1	54/100	*-39506 (-39480)*	79045	79154	0.967
v6_e7	13/100	*-39641 (-39500)*	79338	79527	0.71 (0.969)
v4_e4	63/100	*-39505*	79043	79151	0.969
v5_e5	21/100	*-39500*	79040	79176	0.96 (0.969)
v6_e6	16/100	*-39537 (-39500)*	79122	79285	0.837 (0.969)
v7_e7	6/100	*-39432*	78920	79110	0.454 (0.969)
v8_e11	75/100	*-39451 (-39375)*	78990	79288	0.87 (0.993)
v5_e6_o2	80/100	***-39375***	**78798**	**78960**	0.993

Table 5.1: Initial and full fittings for the v2_e1 simulated recordings.

did perform additional runs of several configured topologies that we suspected could show a higher max deviation than we had measured. Both v4_e5_o2 and v4_e4_i1 were slightly overparameterized and had significant peak LL models. We were curious to see whether additional fittings would cause the max deviation to grow past the medium range. We also selected v6_e5 and v6_e7_menon for further optimizations, two models with high max deviations implied from their ancestor, but only low or medium results themselves.

After the additional runs the max deviations of v4_e5_o2 and v4_e4_i1 grew slightly larger though not to the extent we may have anticipated. It seems likely that the configured topologies could not show complete deviations approaching 1 but we cannot exclude the possibility of undetected solutions. For the two models with large implied max deviations but low observed max deviations the m6_e7_menon did reach a high max deviation while v6_e5 grew only slightly despite reaching the 10 model threshhold.

5.1.2.1 Comparison to generative model

In Table 5.4 we see the that the top v3_e2 model is within 15% of the true value for every parameter. Of particular interest is the LL of the best fit of -39323 is significantly better than the LL of the generative model of -40639, in fact every qualifying topology had a result under -40000. AIC and BIC both identify a v3_e2 as the correct model. The ability of models to outpuerform the generative model can again be explained as a combination of baseline and channel noise. According to our max deviation table 3.4 the correct model belongs to one of six configured topologies. Table 5.4 shows the percentage difference between the parameters of the original and fitted model. Figures 5.4 and 5.5 show the top fitted v3_e2 model and the resulting fit.

5.1.3 Simulated v4_e3 model

Table 5.5 shows results from the initial and full run of fittings for the simulated data generated by the v4_e3 model shown in figure 4.4.

For the first time we see a model selected for additional fittings, the v3_e2_disconn configured topology that achieved only a single fitting (far below the peak LL). The poor performance of the model in achieving peak LL results is an interesting result considering its simplicity. It suggests that factors other than complexity are important to the ability of the optimizer to reproduce the data. The other topology with poor success is v6_e7_menon,

Results for v3_e2 simulated recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	-52399	104807	104834	0.003
v3_e2	100/100	*-39323*	**78663**	**78717**	0.106
v3_e2_disconn	95/100	-49528	99073	99127	0.205
v3_e3	82/100	*-39334 (-39323)*	78693	78775	0.239
v4_e4_i1	86/100	*-39327 (-39323)*	78686	78794	0.423
v4_e4_o2	84/100	*-39353 (-39323)*	78739	78847	0.433
v4_e5_o2	94/100	*-39321*	78682	78818	0.436
v5_e6_o2	94/100	*-39322 (-39321)*	78692	78855	0.688
v4_e3	91/100	*-39321*	78666	78747	0.911
v4_e4	80/100	*-39329 (-39321)*	78691	78800	0.864 (0.911)
v8_e11	82/100	***-39315***	78718	79017	0.856 (0.911)
v5_e4	20/100	*-39327 (-39321)*	78686	78795	0.893 (0.911)
v5_e5	84/100	*-39322 (-39321)*	78684	78820	0.874 (0.911)
v6_e7_menon	3/100	*-39415 (-39321)*	78887	79077	0.436 (0.911)
v6_e5	3/100	*-39722 (-39321)*	79485	79620	0.23 (0.911)
v11_e12	4/100	*-39845 (-39321)*	79787	80113	0.678 (0.911)
v6_e6	49/100	*-39328 (-39321)*	78704	78867	0.674 (0.911)
v7_e7	12/100	*-39329 (-39321)*	78715	78905	0.858 (0.911)
v6_e7	47/100	*-39324 (-39321)*	78704	78894	0.92
Full results for v3_e2 simulated recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	100/100	-52399	104807	104834	0.003
v3_e2	100/100	*-39323*	**78663**	**78717**	0.106
v3_e2_disconn	95/100	-49528	99073	99127	0.205
v3_e3	82/100	*-39334 (-39323)*	78693	78775	0.239
v4_e4_o2	84/100	*-39353 (-39323)*	78739	78847	0.433
v4_e5_o2	359/400	*-39321*	78682	78818	0.436
v4_e4_i1	344/400	*-39327 (-39323)*	78686	78794	0.603
v5_e6_o2	94/100	*-39322 (-39321)*	78692	78855	0.688
v4_e3	91/100	*-39321*	78666	78747	0.911
v4_e4	80/100	*-39329 (-39321)*	78691	78800	0.864 (0.911)
v8_e11	82/100	***-39315***	78718	79017	0.856 (0.911)
v5_e4	20/100	*-39327 (-39321)*	78686	78795	0.893 (0.911)
v5_e5	84/100	*-39322 (-39321)*	78684	78820	0.874 (0.911)
v6_e7_menon	5/200	*-39415 (-39321)*	78887	79077	0.863 (0.911)
v6_e5	10/400	*-39370 (-39321)*	78780	78916	0.264 (0.911)
v11_e12	4/100	*-39845 (-39321)*	79787	80113	0.678 (0.911)
v6_e6	49/100	*-39328 (-39321)*	78704	78867	0.674 (0.911)
v7_e7	12/100	*-39329 (-39321)*	78715	78905	0.858 (0.911)
v6_e7	47/100	*-39324 (-39321)*	78704	78894	0.92

Table 5.3: Initial fittings of the v3_e2 simulated recordings

K0			
C1→O1	O1→C1	C2→C1	C1→C2
-11.9	-8.8	3.8	-6.2
K1			
C1→O1	O1→C1	C2→C1	C1→C2
0.1	8.0	-3.3	-14.4

Table 5.4: Percentage difference in parameters between the best performing v3_e2 model and the generative model

however this model does not require additional fittings as its ancestor, v4_e3, has reached the peak LL range of v6_e7_menon and has a high max deviation.

For an additional round of optimizations, in addition to the recommended topology, v3_e2_disconn, we also selected v6_e7_menon, v6_e5, and v8_e11 to investigate how the observed max deviation would change with additional fittings. In particular we were interested in v8_e11 since it was clearly very overparametarized with a high implied max deviation, however despite a very large number of fittings it still had a medium max deviation.

After 400 fittings the v3_e2_disconn model reaches the 10 model cutoff with a low max deviation, giving us more confidence it is unable to reproduce the data. Surprisingly the measured max deviation among the 192 v8_e11 models is still in the medium range. This carries a worrying implication about the ability to detect large max deviations. Although we know that peak LL range v8_e11 models that show a large max deviation do exist (due to its ancestor v4_e3) these models are apparently quite uncommon for the optimizer to generate.

5.1.3.1 Comparison to generative model

The LL of the generative model was -39265 which is slightly worse than that of the best model overall and slightly better than that of the best fitted v4_e3 model. Again BIC is able to correctly identify a model from the generative configured topology but AIC misidentifies v6_e7. Using the max deviation approach we narrow the best model down to a range of seven configured topologies that includes the generative topology. In table 5.6 we see the percentage difference in each parameter between the best v4_e3 model and the generative model. Despite a slight difference in the LL of the two models the parameters themselves are far less constrained. Figures 5.6 and 5.7 show the top fitted v4_e3 model and the resulting fit.

Results for v4_e3 simulated recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v3_e2_disconn	1/100	-65378	130772	130826	-
v2_e1	75/100	-77037	154083	154110	0.005
v3_e2	99/100	*-40062*	80141	80195	0.09
v3_e3	16/100	*-40087 (-40062)*	80198	80279	0.099
v4_e4_o2	29/100	*-39665*	79362	79471	0.276
v4_e5_o2	15/100	*-39790 (-39665)*	79621	79757	0.447
v5_e6_o2	26/100	*-39460*	78968	79130	0.442 (0.447)
v4_e4_i1	65/100	*-39287*	78607	78716	0.542
v4_e3	97/100	*-39283*	78590	**78671**	0.893
v4_e4	23/100	*-39483 (-39283)*	78998	79106	0.841 (0.893)
v8_e11	94/100	*-39277*	78642	78940	0.409 (0.893)
v6_e7_menon	1/100	*-39552 (-39283)*	79161	79351	- (0.893)
v5_e4	52/100	*-39305 (-39283)*	78642	78751	0.941
v5_e5	67/100	*-39328 (-39283)*	78696	78832	0.565 (0.941)
v6_e5	3/100	*-39616 (-39283)*	79273	79409	0.314 (0.941)
v11_e12	12/100	*-39275*	78646	78971	0.919 (0.941)
v6_e7	74/100	***-39257***	**78571**	78761	0.859 (0.97)
v6_e6	29/100	*-39313 (-39283)*	78674	78837	0.97
v7_e7	8/100	*-39345 (-39283)*	78747	78937	0.639 (0.97)
Full results for v4_e3 simulated recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	75/100	-77037	154083	154110	0.005
v3_e2_disconn	10/400	-65378	130772	130826	0.084
v3_e2	99/100	*-40062*	80141	80195	0.09
v3_e3	16/100	*-40087 (-40062)*	80198	80279	0.099
v4_e4_o2	29/100	*-39665*	79362	79471	0.276
v4_e5_o2	15/100	*-39790 (-39665)*	79621	79757	0.447
v5_e6_o2	26/100	*-39460*	78968	79130	0.442 (0.447)
v4_e4_i1	65/100	*-39287*	78607	78716	0.542
v4_e3	97/100	*-39283*	78590	**78671**	0.893
v4_e4	23/100	*-39483 (-39283)*	78998	79106	0.841 (0.893)
v8_e11	189/200	*-39277*	78642	78940	0.468 (0.893)
v6_e7_menon	10/400	*-39552 (-39283)*	79161	79351	0.883 (0.893)
v5_e4	52/100	*-39305 (-39283)*	78642	78751	0.941
v5_e5	67/100	*-39328 (-39283)*	78696	78832	0.565 (0.941)
v6_e5	6/400	*-39397 (-39283)*	78835	78970	0.38 (0.941)
v11_e12	12/100	*-39275*	78646	78971	0.919 (0.941)
v6_e7	74/100	***-39257***	**78571**	78761	0.859 (0.97)
v6_e6	29/100	*-39313 (-39283)*	78674	78837	0.97
v7_e7	8/100	*-39345 (-39283)*	78747	78937	0.639 (0.97)

Table 5.5: Initial and full fittings for the v4_e3 simulated recordings.

K0					
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3
-7.7	-1.1	-25.6	-28.2	34.2	0.5
K1					
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3
8.8	2.5	88.9	-33.1	-47.9	-1.3

Table 5.6: Percentage difference in parameters between the best performing v4_e3 model and the generative model

5.1.4 Simulated v5_e4 model

Table 5.7 shows results from the initial and full run of fittings for the simulated data generated by the v5_e4 model shown in figure 4.5a.

After the initial fittings eight models are selected for additional optimizations due to insufficient results and 7 of the 19 models are selected as members of the candidate range. After additional fittings the candidate topology range is unchanged. In addition we find three of the configured topologies were unable to reach a termination criteria even after 1000 fittings, this included two models that were likely underparameterized and one that hit the global peak LL. After 1000 fittings we abandon further fittings for reasons of feasibility.

5.1.4.1 Comparison to generative model

The LL for the generative v5_e4 model was -39348, slightly worse than the best v5_e4 fit and better than the v4_e3 model, we cannot say for certain that v4_e3 was insufficient to generate the data but v5_e4 had the lowest LL (and accordingly best AIC and BIC). The max deviation indicates the correct topology is one of 10 models. Table 5.8 shows the percentage difference between the parameters of the original and fitted model, again as complexity increases so does the difference between the original and fitted model. Figures 5.8 and 5.9 show the top fitted v5_e4 model and the resulting fit.

5.1.5 Simulated v6_e5 model

The v6_e5 simulated data was the largest simulated data experiment, the initial and full results are shown in table 5.9. The initial run suggests 10 configured topologies in the candidate range that may contain the correct model, notably the generative topology is not among them. The v6_e5 configured topology is excluded as being overparameterized, this is not necessarily false as the model could potentially be completely reproduced by a simpler

Results for v5_e4 simulated recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v3_e2_disconn	1/100	-79304	158624	158679	-
v2_e1	83/100	-104930	209869	209896	0.006
v3_e2	26/100	-45150	90316	90370	0.048
*v4_e5_o2	3/100	-45259 (-44308)	90559	90694	0.033 (0.051)
*v3_e3	4/100	-45194 (-45150)	90412	90493	0.051
*v4_e4_o2	1/100	-44308	88648	88756	- (0.051)
v4_e4_i1	11/100	*-39501*	79034	79142	0.273
***v5_e6_o2**	9/100	*-39627*	79303	79465	0.275
v4_e3	87/100	*-39472*	78968	79049	0.303
***v4_e4**	1/100	*-39573 (-39472)*	79179	79287	- (0.303)
***v6_e7_menon**	1/100	*-41365 (-39472)*	82786	82976	- (0.303)
v8_e11	35/100	*-39389*	78866	79164	0.523
v6_e7	23/100	*-39479 (-39317)*	79014	79203	0.373 (0.857)
v5_e4	61/100	***-39317***	**78666**	**78774**	0.857
v5_e5	8/100	*-39340 (-39317)*	78720	78856	0.269 (0.857)
v6_e5	2/100	*-39361 (-39317)*	78762	78897	0.728 (0.857)
v11_e12	3/100	*-40886 (-39317)*	81868	82193	0.176 (0.857)
v6_e6	38/100	*-39327 (-39317)*	78702	78865	0.338 (0.857)
v7_e7	11/100	*-39324 (-39317)*	78704	78894	0.775 (0.857)

Full results for v5_e4 simulated recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	83/100	-104930	209869	209896	0.006
*v3_e2_disconn	4/1000	-79304	158624	158679	0.032
v3_e2	26/100	-45150	90316	90370	0.048
*v3_e3	4/1000	-45194 (-45150)	90412	90493	0.051
v4_e5_o2	16/400	-45129 (-44079)	90298	90434	0.074 (0.184)
v4_e4_o2	16/800	-44079	88191	88299	0.184
v5_e6_o2	17/200	*-39627*	79303	79465	0.275
v4_e3	87/100	*-39472*	78968	79049	0.303
v4_e4_i1	27/200	*-39466*	78964	79073	0.34
***v4_e4**	5/1000	*-39529 (-39472)*	79091	79200	0.385
v8_e11	35/100	*-39389*	78866	79164	0.523
v6_e7_menon	8/800	*-39581 (-39472)*	79218	79408	0.827
v6_e7	23/100	*-39479 (-39317)*	79014	79203	0.373 (0.857)
v5_e4	61/100	***-39317***	**78666**	**78774**	0.857
v5_e5	19/200	*-39340 (-39317)*	78720	78856	0.431 (0.857)
v6_e5	2/100	*-39361 (-39317)*	78762	78897	0.728 (0.857)
v11_e12	3/100	*-40886 (-39317)*	81868	82193	0.176 (0.857)
v6_e6	38/100	*-39327 (-39317)*	78702	78865	0.338 (0.857)
v7_e7	11/100	*-39324 (-39317)*	78704	78894	0.775 (0.857)

Table 5.7: Initial and complete fittings for the v5_e4 simulated recordings

K0							
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3	C4→C3	C3→C4
-33.1	-6.1	-21.2	-48.1	29.7	118.0	26.2	-59.2
K1							
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3	C4→C3	C3→C4
-0.7	6.2	109.3	58.3	11.7	38.5	-63.3	-42.9

Table 5.8: Percentage difference in rate constants between the best performing v5_e4 model and the generative model

model.

Five configured topologies are selected for additional fittings based on the protocol.

In the full results even after running 1000 fittings several configured topologies were unable to reach a stopping criteria. The 10 topologies in the candidate range. AIC and BIC both select the v6_e6 topology which contains one additional edge from v6_e5, the best model of this topology slightly exceeded the LL of the generative model (-39271) on the same recording. The max deviation protocol presents a range of 10 potential configured topologies. Significantly v6_e5 is not among the topologies selected by the max deviation topology selection rule. It is precluded by v5_e4 which is well in the global peak LL range and has a max deviation of 0.904. While it is possible the original v6_e5 model was overparameterized (for instance if a v5_e4 model could have perfectly reproduced it) the more likely interpretation is our method falsely detected v6_e5 as overparameterized. This may be a limitation of the protocol where it may underestimate the necessary topology complexity if the effects of that missing complexity are small, such as a model with two missing states still being able to reach the global peak LL range.

5.1.5.1 Comparison to generative model

Table 5.10 shows the percentage difference between the parameters of the original model and the top v6_e5 model. Figures 5.10 and 5.11 show the top fitted v6_e5 model and the resulting fit.

5.1.6 Simulated Data Summary

On the simulated data we answered R1 in the sense that we produced a model with a LL close to, or in excess of, the original model, for each topology.

On R2 we asked whether the candidate topologies would contain the original topology, this too was answered affirmatively except for the v6_e5 simulated recording where it

Results for v6_e5 simulated recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v4_e4_o2	1/100	-51957	103946	104054	-
*v3_e2_disconn	1/100	-105237	210490	210544	-
v2_e1	89/100	-139934	279876	279903	0.008
v3_e2	24/100	-54888	109793	109847	0.022
*v4_e5_o2	3/100	-52563 (-51957)	105167	105303	0.023
*v3_e3	5/100	-54914 (-54888)	109853	109935	0.033
v5_e6_o2	14/100	*-40550*	81149	81312	0.151
v4_e4_i1	21/100	*-40988*	82009	82117	0.191
v4_e3	99/100	*-41046*	82116	82197	0.202
***v4_e4**	3/100	*-41380 (-41046)*	82793	82901	0.199 (0.202)
v8_e11	25/100	*-39290*	78668	78967	0.401
v6_e7_menon	7/100	*-50069 (-41046)*	100195	100385	0.895
v6_e7	7/100	*-39552 (-39220)*	79160	79350	0.899 (0.904)
v5_e4	64/100	*-39305*	78642	78750	0.904
v5_e5	16/100	*-39262*	78564	78700	0.407 (0.904)
v6_e5	4/100	*-39448 (-39305)*	78937	79073	0.851 (0.904)
v11_e12	1/100	*-39492 (-39305)*	79080	79406	- (0.904)
v6_e6	43/100	***-39220***	**78489**	**78652**	0.472 (0.904)
v7_e7	20/100	*-39253 (-39220)*	78562	78751	0.512 (0.904)
Full results for v6_e5 simulated recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v4_e4_o2	1/1000	-51957	103946	104054	-
v2_e1	89/100	-139934	279876	279903	0.008
v3_e2	24/100	-54888	109793	109847	0.022
*v3_e3	6/1000	-54914 (-54888)	109853	109935	0.033
*v4_e5_o2	5/1000	-52156 (-51957)	104353	104489	0.04
*v3_e2_disconn	8/1000	-104386	208788	208843	0.096
v5_e6_o2	17/200	*-40550*	81149	81312	0.189
v4_e4_i1	21/100	*-40988*	82009	82117	0.191
v4_e3	99/100	*-41046*	82116	82197	0.202
***v4_e4**	6/1000	*-41298 (-41046)*	82628	82736	0.301
v8_e11	25/100	*-39290*	78668	78967	0.401
v6_e7_menon	7/100	*-50069 (-41046)*	100195	100385	0.895
v6_e7	7/100	*-39552 (-39220)*	79160	79350	0.899 (0.904)
v5_e4	64/100	*-39305*	78642	78750	0.904
v5_e5	16/100	*-39262*	78564	78700	0.407 (0.904)
v6_e5	4/100	*-39448 (-39305)*	78937	79073	0.851 (0.904)
v11_e12	1/100	*-39492 (-39305)*	79080	79406	- (0.904)
v6_e6	43/100	***-39220***	**78489**	**78652**	0.472 (0.904)
v7_e7	20/100	*-39253 (-39220)*	78562	78751	0.512 (0.904)

Table 5.9: Initial and complete fittings for the v6_e5 simulated recordings

K0									
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3	C4→C3	C3→C4	C4→C5	C5→C4
130.5	-63.8	-72.5	12.6	-55.3	-76.6	0.6	-51.4	20.0	-96.8
K1									
C1→O1	O1→C1	C2→C1	C1→C2	C3→C2	C2→C3	C4→C3	C3→C4	C4→C5	C5→C4
-95.8	76.6	458.6	-3.1	32.9	-26.8	-18.0	92.8	96.9	-711.5

Table 5.10: Percentage difference in rate constants between the best performing v5_e4 model and the generative model

identified the original topology as overparameterized.

In question R3 we asked the degree to which we could find parameters close to the original model. What we found was that as the number of states increased the degree to which the original parameters could be recovered diminished significantly. As shown in table 5.10 both k0 and k1 parameters could vary from the original values by well over 100%.

One important question was whether a topology that fell short of the global peak LL range would show a high max deviation. Across all five experiments no topology satisfied this condition.

Another question was R8, whether we would detect a high max deviation in topologies known to be overparameterized. This held true for all experiments except for the simplest v2_e1 simulated recording. In all cases of a topology known to be overparameterized we were able to detect a high max deviation. However, in several instances we only did so by detecting the high max deviation though an ancestor topology. This is particularly evident in the case of the v8_e11 topology for the v4_e3 simulated data shown in table 5.5. Even 189 models were insufficient to generate a max deviation of more than 0.468 despite the fact that a max deviation of at least 0.893 was possible as shown by an ancestor topology.

Regarding question R4 other measures such as LL, AIC, and BIC were inconsistent at detecting the proper topology. In only one of five experiments did LL select the proper topology, AIC succeeded in two of five and BIC three of five.

5.2 Experimental Data

The optimization experiments involving the K+ channel recordings investigate question R5 about whether we will be able to find high quality fits for the experimental data. In addition we also investigate R6, whether the candidate topologies will be similar for each channel type regardless of Control or IL conditions. We also gather data for R7 through R9.

5.2.1 Control A-Type

We performed an initial set of 100 fittings on the recordings of *Control A-type* channels. The results of the initial fittings are shown in table 5.11. Thirteen topologies are below the 10 model threshold and two are selected for additional optimizations. As well the initial fittings show thirteen configured topologies in the candidate range.

The additional rounds of optimizations reduced the number topologies in the candidate range to seven. AIC and BIC both indicate the model with the best LL, the top v5_e6_o2 model. However, our max deviation metric indicates that this model is unnecessarily complex.

The simplest model in the selected topologies belongs to the v3_e2 and v3_e2_disconn topologies. If we inspect either of those topologies in detail they appear to be underparameterized. More complex topologies in the candidate range such as v4_e4_o2 appear to fit the data quite well. However, we intend to compare the selected model to another model from the *IL A-Type* topology, and the only two topologies selected by each recording are v3_e2 and v3_e2_disconn. So we plot the best v3_e2_disconn model against the recorded data as shown in figure 5.13.

5.2.2 IL A-Type

Our initial run of the *IL A-Type* recordings consisted of 100 models and already showed conclusive results. Specifically the v3_e2 configured topology both supplied a large number of peak LL models and a large max deviation. Since v3_e2 also reached the global peak LL it suggested that additional fittings for v3_e2, or any descendant models, were unnecessary.

Interestingly due to the high max deviation of the v3_e2 topology and the large number of topologies descended from it only v3_e2_disconn required additional fittings which were sufficient to push the max deviation into the medium range. The only topologies in the candidate range are the v3_e2_disconn and v3_e2 topologies. This is consistent with the Control A-Type results though that recording fostered a larger range of topologies.

Again AIC and BIC simply select the model with the best LL, a v6_e6 model which is inconsistent with the v5_e6_o2 model from the Control recording. To showcase the quality of the fits we choose the v3_e2_disconn model with the fit shown in figure 5.15.

Results for Control A-Type recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	73/100	-97157	194323	194350	0.02
v3_e2_disconn	21/100	*-47155*	94326	94380	0.356
v3_e2	34/100	*-47897*	95811	95866	0.364
***v4_e5_o2**	2/100	*-46587*	93215	93351	0.172 (0.408)
***v4_e4_o2**	5/100	*-46784*	93601	93710	0.284 (0.408)
v3_e3	24/100	*-47083*	94190	94271	0.41
v4_e4_i1	18/100	*-46715*	93462	93570	0.681
v4_e3	10/100	*-47839*	95703	95784	0.686
v5_e4	11/100	*-47861 (-47839)*	95754	95863	0.587 (0.686)
v11_e12	2/100	*-49738 (-47839)*	99573	99898	0.501 (0.686)
v4_e4	9/100	*-47003*	94039	94148	0.677 (0.686)
v6_e7_menon	4/100	*-46669*	93394	93584	0.547 (0.686)
v6_e5	4/100	*-47864 (-47839)*	95768	95903	0.575 (0.686)
v8_e11	2/100	*-48060 (-46489)*	96209	96507	0.286 (0.915)
v5_e6_o2	6/100	***-46489***	**93026**	**93188**	0.915
v6_e7	7/100	*-47756 (-46692)*	95569	95759	0.941 (0.948)
v5_e5	8/100	*-46692*	93425	93561	0.948
v6_e6	4/100	*-48274 (-46692)*	96597	96760	0.937 (0.948)
v7_e7	6/100	*-49131 (-46692)*	98318	98508	0.904 (0.948)
Full results for Control A-Type recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	73/100	-97157	194323	194350	0.02
v3_e2_disconn	21/100	*-47155*	94326	94380	0.356
v3_e2	34/100	*-47897*	95811	95866	0.364
v4_e4_o2	13/1000	*-46675*	93383	93491	0.317 (0.408)
v3_e3	24/100	*-47083*	94190	94271	0.41
v4_e4_i1	18/100	*-46715*	93462	93570	0.681
v4_e3	10/100	*-47839*	95703	95784	0.686
v5_e4	11/100	*-47861 (-47839)*	95754	95863	0.587 (0.686)
v11_e12	2/100	*-49738 (-47839)*	99573	99898	0.501 (0.686)
v4_e4	9/100	*-47003*	94039	94148	0.677 (0.686)
v6_e7_menon	4/100	*-46669*	93394	93584	0.547 (0.686)
v6_e5	4/100	*-47864 (-47839)*	95768	95903	0.575 (0.686)
v4_e5_o2	14/1000	*-46587*	93215	93351	0.885
v8_e11	2/100	*-48060 (-46489)*	96209	96507	0.286 (0.915)
v5_e6_o2	6/100	***-46489***	**93026**	**93188**	0.915
v6_e7	7/100	*-47756 (-46692)*	95569	95759	0.941 (0.948)
v5_e5	8/100	*-46692*	93425	93561	0.948
v6_e6	4/100	*-48274 (-46692)*	96597	96760	0.937 (0.948)
v7_e7	6/100	*-49131 (-46692)*	98318	98508	0.904 (0.948)

Table 5.11: Initial and complete fittings of the Control A-Type recordings

Results for IL A-Type recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	67/100	-40182	80372	80399	0.009
***v3_e2_disconn**	3/100	*-36176*	72368	72422	0.226
v4_e4	17/100	*-35744*	71521	71629	0.681 (0.731)
v4_e3	11/100	*-35897*	71819	71901	0.731 (0.745)
v3_e2	15/100	*-35951*	71919	71974	0.745
v4_e5_o2	2/100	*-38076 (-35951)*	76193	76329	0.673 (0.745)
v3_e3	1/100	*-35975 (-35951)*	71975	72056	- (0.745)
v8_e11	1/100	*-36010 (-35854)*	72108	72406	- (0.745)
v4_e4_o2	2/100	*-40812 (-35951)*	81656	81764	0.458 (0.745)
v5_e6_o2	3/100	*-35854*	71756	71919	0.543 (0.745)
v5_e4	7/100	*-36560 (-35897)*	73153	73261	0.801
v11_e12	28/100	*-36039 (-35897)*	72175	72501	0.79 (0.801)
v6_e5	13/100	*-37379 (-35897)*	74798	74933	0.755 (0.801)
v6_e7_menon	16/100	*-35612*	71280	71470	0.836
v4_e4_i1	18/100	*-35621*	71274	**71383**	0.838
v5_e5	34/100	*-35630*	71300	71436	0.853
v6_e6	27/100	***-35607***	**71263**	71426	0.853
v7_e7	7/100	*-36178 (-35607)*	72413	72603	0.728 (0.853)
v6_e7	38/100	*-35672 (-35607)*	71401	71591	0.917

Full results for IL A-Type recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
v2_e1	67/100	-40182	80372	80399	0.009
v3_e2_disconn	10/200	*-36176*	72368	72422	0.346
v4_e4	17/100	*-35744*	71521	71629	0.681 (0.731)
v4_e3	11/100	*-35897*	71819	71901	0.731 (0.745)
v3_e2	15/100	*-35951*	71919	71974	0.745
v4_e5_o2	2/100	*-38076 (-35951)*	76193	76329	0.673 (0.745)
v3_e3	1/100	*-35975 (-35951)*	71975	72056	- (0.745)
v8_e11	1/100	*-36010 (-35854)*	72108	72406	- (0.745)
v4_e4_o2	2/100	*-40812 (-35951)*	81656	81764	0.458 (0.745)
v5_e6_o2	3/100	*-35854*	71756	71919	0.543 (0.745)
v5_e4	7/100	*-36560 (-35897)*	73153	73261	0.801
v11_e12	28/100	*-36039 (-35897)*	72175	72501	0.79 (0.801)
v6_e5	13/100	*-37379 (-35897)*	74798	74933	0.755 (0.801)
v6_e7_menon	16/100	*-35612*	71280	71470	0.836
v4_e4_i1	18/100	*-35621*	71274	**71383**	0.838
v5_e5	34/100	*-35630*	71300	71436	0.853
v6_e6	27/100	***-35607***	**71263**	71426	0.853
v7_e7	7/100	*-36178 (-35607)*	72413	72603	0.728 (0.853)
v6_e7	38/100	*-35672 (-35607)*	71401	71591	0.917

Table 5.12: Initial and complete fittings of the IL A-Type recordings

5.2.3 A-Type Overview

To examine the effects of the Interleukin on the *A-Type* recordings we selected one model from each recording in order to compare their behaviour. The only topologies in the candidate range for both recordings were v3_e2_disconn and v3_e2. Note that being selected only means they are potentially the correct topology, the investigator should exercise judgment in determining if they are proper fits.

In the case of the IL recording the v3_e2 and v3_e2_disconn topologies were the only ones indicated, and accordingly the top model from both seem to properly reproduce the data. The high level of noise in the IL recording may have have obscured more subtle effects. On the control data the top models from the v3_e2 and v3_e2_disconn topologies, despite reaching the global peak LL range, did not seem to adequately reproduce the data when we inspected the fits in detail. To properly reproduce the data a model belonging to a more complex topology within the selected range, such as v4_e4_o2, was required.

Unfortunately even if we generated high quality v4_e4_o2 models for the IL recording (the few that were generated were of poor quality) such a comparison would be undercut by the large max deviations that would be present in the fittings for the models fit to the IL recording.

Instead we compare the state occupancy profiles of the best v3_e2_disconn model for each recording. While the control model will be underparameterized from the perspective of the control recording v3_e2_disconn is the only topology that reaches the global peak LL range with a low or medium max deviation for both recordings. Even then the max deviations of 0.356 and 0.209 should be interpreted in light of the fact that v3_e2_disconn can not exhibit a max deviation over 0.5 due to the isomorphism in the closed states.

Figure 5.16 shows the two state occupancy plots, for visual clarity we confine the plots to the final voltage sweep. The important difference is in the open states. The control *A-Type* open state experiences strong activation followed by almost complete inactivation (any remaining activation is indistinguishable from noise). In the IL recording activation is quite mild and only partial inactivation occurs. The apparent reversal of the roles of C1 and C2 is misleading due to the isomorphic nature of the topology.

For this channel type R5 seems to be validated as we found many strong fits for each channel type. We cannot confirm whether the correct topology was inside the candidate range. However, the differences between the best fits inside the range and the best fits

among topologies thought to be overparameterized were very mild. R6 is also confirmed but with reservations. The candidate topology ranges were consistent but the topologies that were in the section of overlap were likely underparameterized for the control topology. The v3_e2 and v3_e2_disconn topologies may have been underparameterized for the IL recording as well but could be distinguished due to the high level of noise. R7, regarding whether topologies outside the global peak LL range would not have a high max deviation, was validated for the one topology (v2_e1) that satisfied this criteria.

5.2.4 Control Delayed Rectifier

The dynamics of the Control *Delayed Rectifier* records were much more difficult than any previous recordings. After an initial run of 100 optimizations only three configured topologies did not require additional fittings. We performed additional fittings though the difficulty of the problem meant that in addition to needing many more fittings than earlier problems, each fitting also took much longer as the optimization algorithms had trouble reaching their termination criteria. Despite more than a month of CPU time we were unable to reach the termination criteria described in table 3.4 for any additional configured topologies. Seeing an exhaustive investigation was infeasible we decided to concentrate our computational resources on the most interesting configured topologies. After bringing all selected topologies up to 400 fittings we only continued for ones that either reached the global maxima, or we suspected were complex enough to reach the global maxima, and configured topologies that showed an interesting condition such as a poor peak LL but a high max deviation.

The results are sparse enough that the max deviation metric simply indicates the four configured topologies that managed to reach a fitting in the peak LL range. We also note that v6_e6, despite not reaching the global peak LL, has a max deviation of 0.661. This is an interesting result because it contradicts our expectation that we should not see a high max deviation and a result that does not reach the global peak LL. However, this is based on only 6 results when our rules suggest 20 is necessary in such a situation. We may be seeing a high max deviation simply because the models failed to reach the optimal peak LL range for that configured topology.

But if the result is accurate and those six models do represent the optimal peak LL range for that configured topology then they may represent another phenomena. Most likely as the number of states increase we would expect the max deviation to increase since each

state has less impact on the behaviour. For a 6 state model a max deviation of 0.661 may properly represent a medium or even low max deviation. This is unfortunately a question we were unable to adequately explore due to the difficulty of finding fits for the configured topology in question.

5.2.5 IL Delayed Rectifier

The dynamics for the IL *Delayed Rectifier* recording may be even more difficult than the Control recording. After an initial run of 100 fittings shown in table 5.14 only a single model in a single configured topology is in the global peak LL range and only the v2_e1 topology is exempt from further fittings. As a result we again embraced the strategy of targeted additional fittings after 400 fittings.

During the extended fittings one interesting situation arose at 400 fittings with v4_e4_i1[2]. It had 8 qualifying models, a high max deviation of 0.673, and a LL of -56483 that did not reach the global peak LL range. This allowed us another opportunity to test our prediction that the true peak LL for that topology would be higher. After increasing the fittings to 800 we found the LL of the best model increased to -54624, this was sufficient to shift the peak LL range and reduce the max deviation into the medium range.

In the full results v6_e7 also has a high max deviation and a result that does not reach the global peak LL. However, we know those results do not reach the topology's true peak LL as known from its ancestor, so they do not violate our prediction that a topology below the global peak LL will not show a high max deviation. Out of several thousand fittings across our topologies the candidate range includes only two models from the v6_e7_menon configured topology. We select the better of these two models to show in figure 5.20.

5.2.6 Delayed Rectifier Overview

The *Delayed Rectifier* recordings had the opposite problem from the *A-Type*. Instead of multiple topologies and models to choose from there was only one topology that entered the global peak LL range for both recordings, v6_e7_menon. Accordingly we compare the best fitting from each recording in figure 5.21. The closed state occupancies are of little interest since even if the topologies did have a physiological interpretation we do not have sufficient results to know the size of the max deviations for those topologies, or even if we did enter

[2]This is not shown in table 5.14 as the table shows the full results where the topology was run to 800 models

Results for Delayed Rectifier control recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
***v6_e7_menon**	1/100	*-41854*	83765	83955	-
*v4_e4_o2	1/100	-47771	95574	95682	-
*v7_e7	1/100	-44453	88963	89152	-
*v4_e4_i1	1/100	-46967	93967	94075	-
v2_e1	99/100	-137192	274393	274420	0.009
v3_e2	70/100	-72217	144451	144505	0.037
*v4_e3	2/100	-48223	96470	96551	0.039
*v4_e4	1/100	-46842	93717	93826	- (0.039)
*v5_e5	1/100	-47814 (-46842)	95669	95805	- (0.039)
*v11_e12	1/100	-79856 (-48223)	159809	160134	- (0.039)
*v6_e5	3/100	-73440 (-48223)	146921	147056	0.063
*v3_e3	5/100	-50464	100953	101034	0.133
*v3_e2_disconn	3/100	-67700	135416	135471	0.216
***v5_e6_o2**	5/100	***-40637***	**81323**	**81486**	0.365
*v5_e4	2/100	-64296 (-48223)	128625	128733	0.377
***v8_e11**	4/100	*-41452 (-40637)*	82993	83291	0.463
***v4_e5_o2**	2/100	*-42550*	85141	85276	0.551
*v6_e7	1/100	-50640 (-46603)	101336	101526	- (0.673)
*v6_e6	3/100	-46603	93255	93418	0.673
Full results for Delayed Rectifier control recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
***v6_e7_menon**	1/800	*-41854*	83765	83955	-
v2_e1	99/100	-137192	274393	274420	0.009
v3_e2	70/100	-72217	144451	144505	0.037
*v4_e3	2/400	-48223	96470	96551	0.039
*v6_e5	1/400	-64594 (-48223)	129228	129363	- (0.039)
*v11_e12	1/400	-66669 (-48223)	133435	133760	- (0.039)
*v4_e4	2/400	-46842	93717	93826	0.089
*v4_e4_o2	3/400	-47524	95080	95188	0.105
*v3_e3	8/400	-50464	100953	101034	0.133
*v3_e2_disconn	3/100	-67700	135416	135471	0.216
v5_e6_o2	10/800	***-40637***	**81323**	**81486**	0.365
*v5_e4	7/400	-64296 (-48223)	128625	128733	0.38
*v4_e4_i1	5/400	-46100	92232	92341	0.542
***v8_e11**	9/800	*-41452 (-40637)*	82993	83291	0.545
***v4_e5_o2**	3/800	*-42550*	85141	85276	0.551
*v7_e7	1/800	-44453	88963	89152	- (0.643)
*v6_e7	1/400	-47761 (-45080)	95578	95768	- (0.667)
*v5_e5	7/1000	-45080	90201	90336	0.667
*v6_e6	6/1000	-45545 (-45080)	91139	91301	0.661 (0.667)

Table 5.13: Initial and complete fittings of the Delayed Rectifier Control recordings

Results for Delayed Rectifier IL recordings at 100 generations					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v5_e5	1/100	-49188	98416	98552	-
***v6_e7_menon**	1/100	***-45114***	**90284**	**90474**	-
v2_e1	98/100	-397406	794821	794848	0.002
*v3_e2_disconn	3/100	-75976	151969	152023	0.004
*v5_e6_o2	1/100	-50676	101401	101564	- (0.023)
*v3_e2	3/100	-93088	186192	186247	0.13
*v3_e3	2/100	-61675	123375	123456	0.167
*v7_e7	2/100	-56214 (-49188)	112484	112674	0.21
*v4_e3	3/100	-58637	117298	117379	0.265
*v8_e11	2/100	-52809 (-50676)	105707	106006	0.27
*v4_e5_o2	1/100	-60555 (-52601)	121150	121285	- (0.283)
*v4_e4_o2	4/100	-52601	105235	105344	0.283
*v6_e5	4/100	-106507 (-58637)	213055	213191	0.296
*v5_e4	2/100	-69706 (-58637)	139444	139553	0.361
*v6_e6	2/100	-61651 (-49188)	123350	123513	0.403
*v4_e4	5/100	-61342 (-58637)	122716	122824	0.571
*v4_e4_i1	5/100	-56483	112999	113107	0.619
*v6_e7	4/100	-60163 (-49188)	120383	120573	0.646
*v11_e12	3/100	-72271 (-58637)	144639	144965	0.808
Full results for Delayed Rectifier IL recordings					
Topology	# peak LL	Best LL	AIC	BIC	Max Deviation
*v4_e4	1/400	-56227	112486	112595	-
*v11_e12	1/400	-55240	110576	110902	-
*v5_e6_o2	1/800	-48101	96251	96414	-
v2_e1	98/100	-397406	794821	794848	0.002
*v3_e2_disconn	4/800	-75976	151969	152023	0.019
*v3_e3	3/800	-57814	115652	115734	0.066
*v3_e2	3/800	-93088	186192	186247	0.13
*v4_e5_o2	2/1000	-52694 (-52601)	105428	105564	0.016 (0.283)
*v4_e4_o2	4/400	-52601	105235	105344	0.283
*v4_e3	8/1000	-58637	117298	117379	0.288
*v6_e5	2/400	-98474 (-58637)	196988	197123	0.316
*v4_e4_i1	2/800	-54624	109280	109388	0.326
*v5_e4	2/400	-69706 (-58637)	139444	139553	0.361
***v6_e7_menon**	2/1000	***-45114***	**90284**	**90474**	0.423
*v8_e11	11/1000	-50523 (-48101)	101134	101433	0.558
*v5_e5	3/1000	-47818	95677	95813	0.609
*v6_e6	1/800	-52143 (-47818)	104334	104497	- (0.609)
*v7_e7	3/400	-54688 (-47818)	109433	109623	0.47 (0.609)
*v6_e7	8/1000	-54908 (-47818)	109873	110063	0.795

Table 5.14: Initial and complete fittings of the IL Delayed Rectifier recordings

the true global peak LL range. The open state does bear more interest as it is based on the data.

In both recordings the shape of the open state is quite similar throughout the stimulation pulse except that the activation in the IL trace if far more subdued. The start of the deactivation pulse from 350-400 ms is interesting because of how the open state deactivates. Deactivation occurs at a faster rate in the control model than the IL model. The model error in this section is not particularly high so it seems to be a real effect. However the implications to the physiology should still be limited. The combination of model error and recording noise obscure the magnitude of the effect. The control recording also has a slight negative current that persists to the end of the deactivation section. This current is absent in the IL recording. Whether this is an actual characteristic of the ion channels or a recording artifact is unknown.

For this channel type R4 is only partially validated. Although we found adequate fits there were only a handful. It is possible that a larger compliment of slightly more complex topologies would have found a better set of results. R5 is again confirmed but with reservations. The candidate topology ranges were consistent but the overall lack of peak LL fits gives us low confidence in this conclusion. For R7 we never reached a termination criteria for any topology, thus we lack sufficient data to make a determination.

5.2.7 Experimental Data Summary

With both A-Type recordings we were able to achieve a global peak LL using the same simple topologies. The range of configured topologies suggested for the Control A-Type recordings were larger than the IL A-Type but both results were consistent in the sense that the IL estimate was a subset of the Control estimate.

Similarly the Delayed Rectifier recordings were also consistent, the single model identified for the IL *Delayed Rectifier* recordings was one of the four models identified for the Control *Delayed Rectifier*. However, the *Delayed Rectifier* recordings were characterized by the difficulty of fitting the data. No configured topologies outside of the simplest achieved a termination criteria in either experiment. For the handful of peak LL fittings we did achieve with the Control *Delayed Rectifier* in table 5.11 the max deviations were in the medium range. We do not know if that was the true max deviation as we did not even achieve the 10 model threshhold. However, we may have had more success if we included more descendants

of those models.

One thing evident during the simulated data experiments was as we increased the complexity of models used to generate the recordings the proportion of fitted models in the peak LL range decreased. If the process generating the Delayed Rectifier recordings were sufficiently complex than any configured topology would have a low probability of reaching a global optima. Similarly in general, as the complexity of the configured topology we try to fit increased the proportion of peak LL models generally decreased. For particularly complex dynamics like the *Delayed Rectifier* recordings, fitting a configured topology complex enough to reproduce them might be a very difficult task.

We suspect several strategies could help improve the fittings of difficult recordings such as the *Delayed Rectifier* recordings. As we saw with the simulated data not all configured topologies that achieve the global peak LL did so with the same success rate. If we could find topologies that were better suited to the *Delayed Rectifier* dynamics than we may have more luck in generating sufficient peak LL models to evaluate them. Another strategy may involve a more thorough investigation of the optimizer profile. We used a profile determined in section 3.3, but that profile was based on a single configured topology on a single recording, a different set of parameters may have proven more effective for the *Delayed Rectifier* recordings. On question R5 on the protocols applicability to the experimental data, we either need a more appropriate set of topologies, or an optimization setup that generated peak LL models at a higher rate.

5.2.8 Max Deviation Distributions

Regarding question R9, on how the deviations are distributed among a group of peak LL, we consider model sets from across the full set of experiments.

More specifically, define max deviation $MD(S)$ as a function that takes a set of models S, finds a subset $s \in S$ that represents the peak LL range of S, then returns the max deviation of the set s.

Then if M is the set of all possible models for the configured topology then $MD(M)$ is the true max deviation for the configured topology.

Now if m is a subset of M generated by the optimizer then $MD(m)$ is the measured max deviation of that subset. The question is how well $MD(m)$ functions as an estimator of $MD(M)$ and whether we can judge the accuracy of a particular estimate.

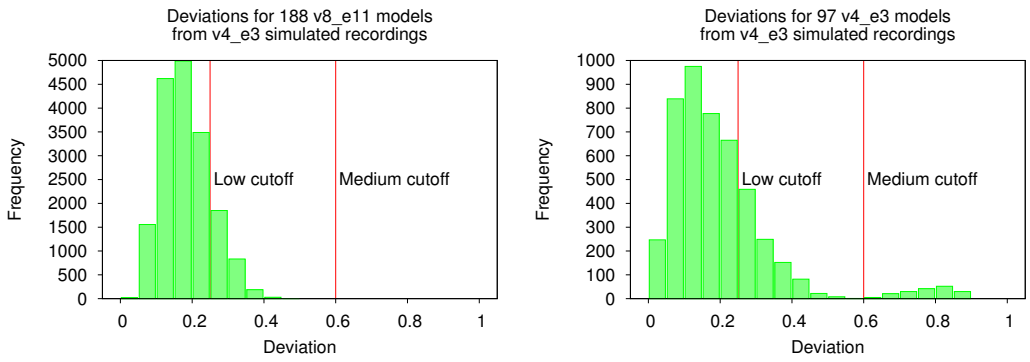


Figure 5.1: The distribution of deviations between models for the two sets of peak LL models. Since a deviation exists between every pair of models the total number of deviations for a set of N models is $\sum_{n=0}^{N-1} n$, meaning 17578 deviations for the 189 v8_e11 models and 4560 for the 97 v4_e3 models.

Unfortunately our results do not suggest a straightforward way to make this estimate. The most dramatic example is from the simulated v4_e3 recordings in section 5.1.3. The histograms of the deviations for the v8_e11 model and its v4_e3 ancestor are shown in figure 5.1. The v4_e3 topology is a subset of v8_e11 in three different ways, and there exists solutions of high deviation v4_e3 models that are in the peak LL range of the v8_e11 configured topology. Yet despite this not one of the 189 v8_e11 models shows this alternative solution. In fact viewing the histogram of the v4_e3 models we see very few high deviations between [.7, .9), further analysis revealed these results to be caused by just two models. If these models were not generated the measured max deviation would have been in the medium range.

One way to measure the degree to which this is an issue is to look at configured topologies where we have a change to observe an underestimated max deviation. Across all experiments we look at the configured topologies where an ancestor shares the descendants peak LL range. These represent all the instances where we could potentially detect an underestimated max deviation.

It should be noted that the ancestor models would have underestimated the max deviation to some extent as well, and not necessarily to a similar degree since the ancestor topologies are both simpler in a topological sense, but also less complex with respect to the recording.

Out of 26 configured topologies that had a *low* or *medium* max deviation, an ancestor which reached its peak LL range, and a minimum of 10 fittings, a total of 14 configured topologies had an ancestor with a higher max deviation. Of these 14 models in 11 cases

it was sufficient to push the max deviation into a higher category. In all 11 cases it was a *medium* max deviation with a parent having a *high* max deviation. Among 31 models with a minimum of 10 fittings where an ancestor in the peak LL range had a larger max deviation the average gap was 0.21.

For configured topologies that did not reach the 10 model threshhold but had a *low* or *medium* max deviation and an ancestor which reached its peak LL range there are 21 candidates. Of these 8 were underestimated and 7 were underestimated significantly enough to change the category, one from *low* to *medium* and six from *medium* to *high*. Among 35 models with fewer than 10 fittings where an ancestor in the peak LL range had a larger max deviation the average gap was 0.18.

The similarity between group above the 10 model threshhold and the one below seems to suggest the number of fittings was not very influential, and to some extent that is correct. On models we ran for more results the max deviation only increased by an average of 0.1. But another important factor is that the average number of peak LL fittings per configured topology varied significantly by recording. So when the child topology hit 10 models the parent generally did as well, and when the child failed to hit 10 the parent had fewer fittings as well. So while fewer fittings made the max deviation numbers less accurate the effect on the child-parent difference generally canceled out.

It is difficult to do a general analysis of the max deviations because it is so dependent on the particular recording and topology selection. But we do see evidence that this approach will underestimate max deviations by at least a category for a significant portion of configured topologies.

5.2.9 Plots from model fittings

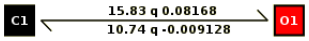


Figure 5.2: Selected model for v2_e1 simulated recordings. Rate parameters between states are shown as 'k0 q k1'

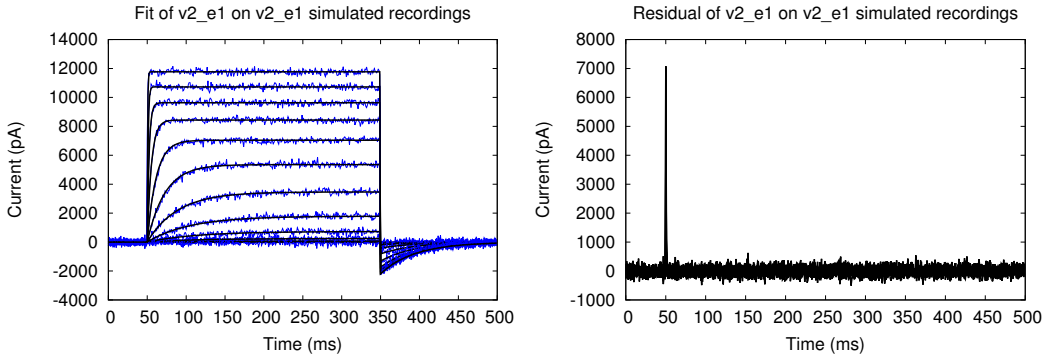


Figure 5.3: Selected fitting for v2_e1 simulated recordings

Figure 5.4: Selected model for v3_e2 simulated recordings. Rate parameters between states are shown as 'k0 q k1'

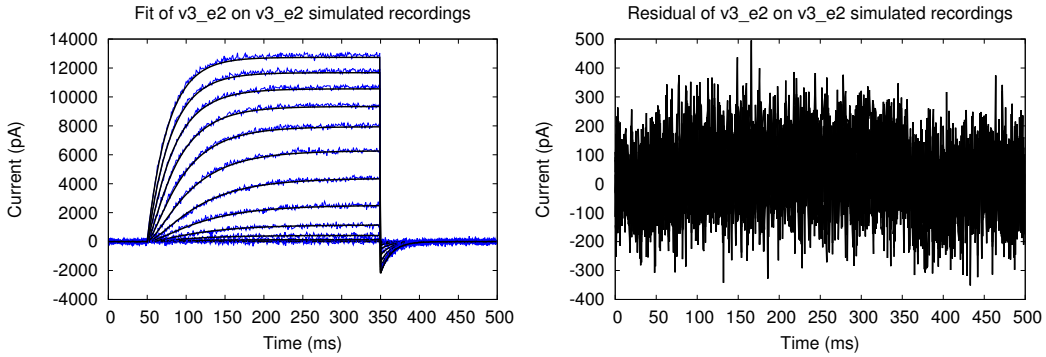


Figure 5.5: Selected fitting for v3_e2 simulated recordings

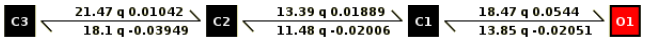


Figure 5.6: Selected model for v4_e3 simulated recordings. Rate parameters between states are shown as 'k0 q k1'

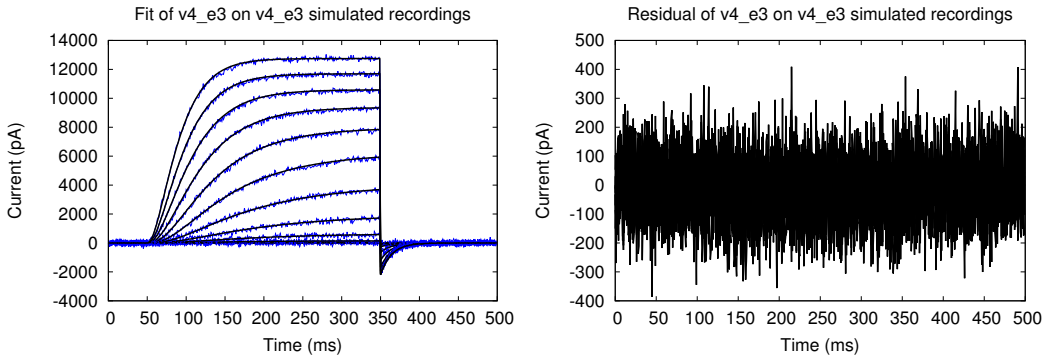


Figure 5.7: Selected fitting for v4_e3 simulated recordings

Figure 5.8: Selected model for v5_e4 simulated recordings. Rate parameters between states are shown as 'k0 q k1'

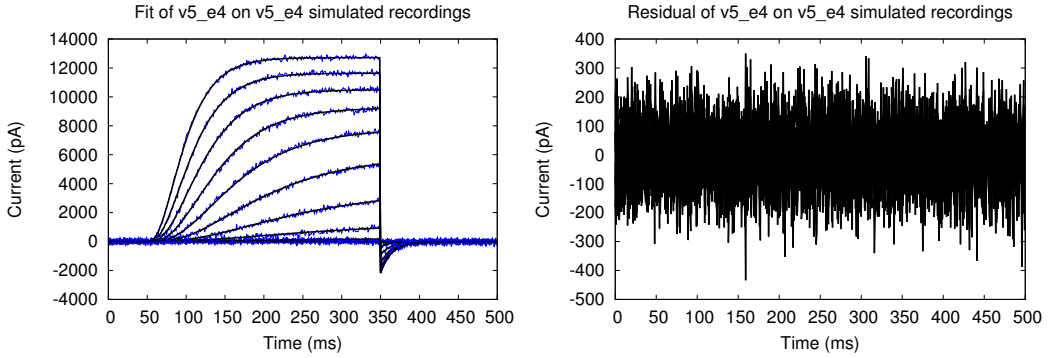


Figure 5.9: Selected fitting for v5_e4 simulated recordings

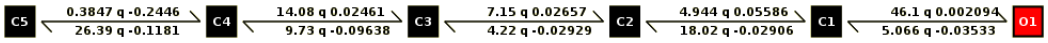


Figure 5.10: Selected model for v6_e5 simulated recordings. Rate parameters between states are shown as 'k0 q k1'

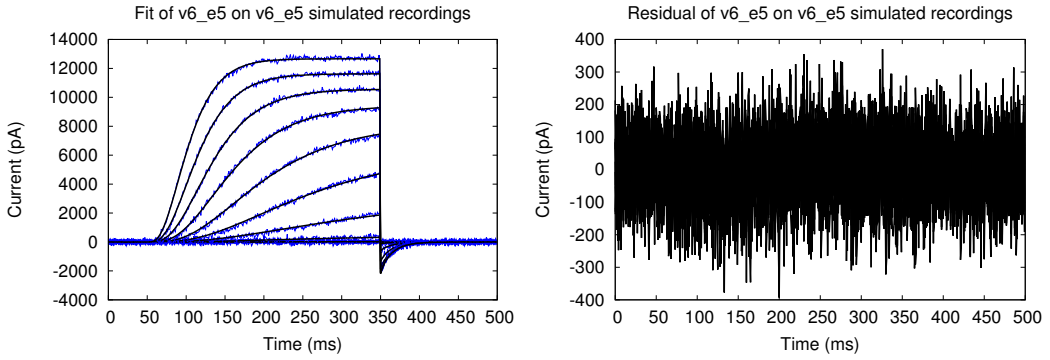


Figure 5.11: Selected fitting for v6_e5 simulated recordings

Figure 5.12: Selected model for *Control A-Type* recording. Rate parameters between states are shown as 'k0 q k1'

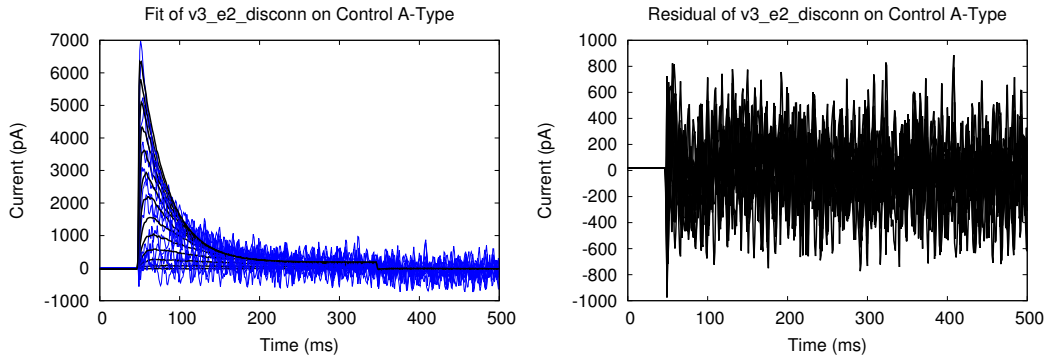


Figure 5.13: Selected fitting for *Control A-Type* recording

Figure 5.14: Selected model for *IL A-Type* recording. Rate parameters between states are shown as 'k0 q k1'

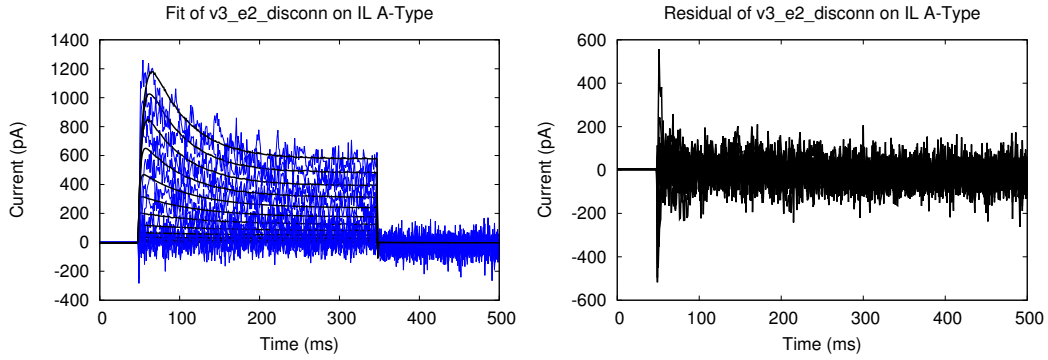


Figure 5.15: Selected fitting for *IL A-Type* recording

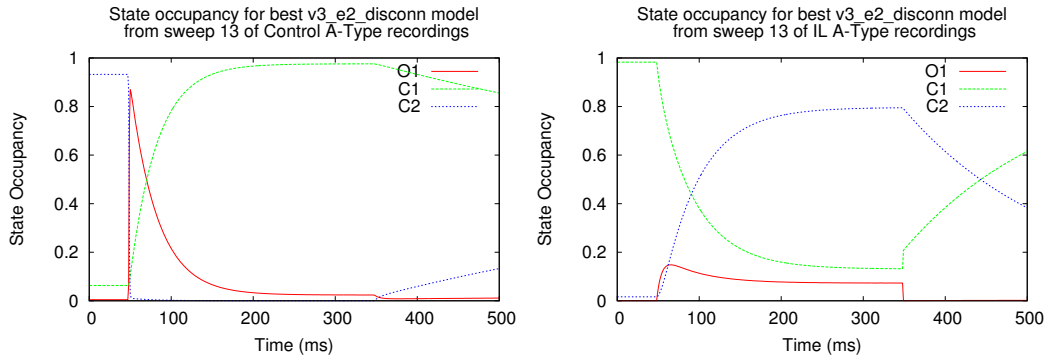


Figure 5.16: State occupancy plots of best v3_e2_disconn fittings for *A-Type* recordings. Note that v3_e2_disconn is isomorphic so C1 and C2 can be switched.

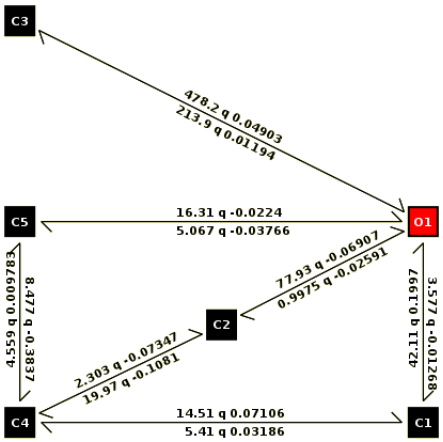


Figure 5.17: Selected model for *Control Delayed Rectified* recording. Rate parameters between states are shown as 'k0 q k1'

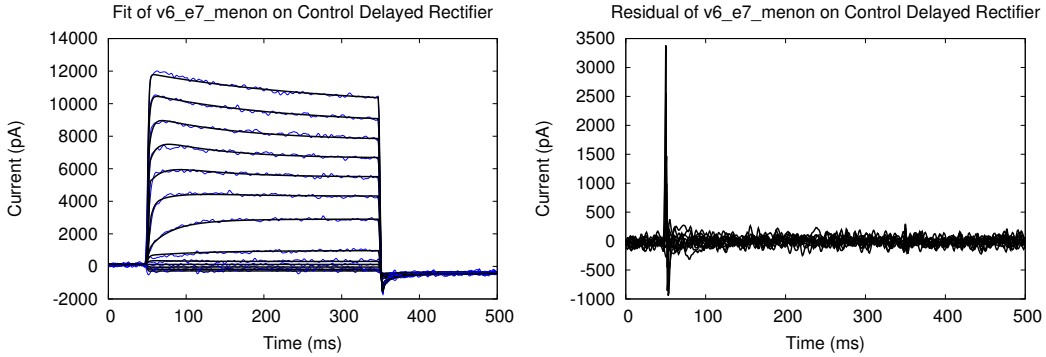


Figure 5.18: Selected fitting for *Control Delayed Rectified* recording

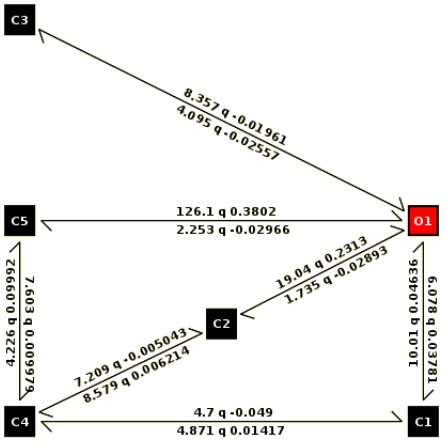


Figure 5.19: Selected model for *IL Delayed Rectifier* recording. Rate parameters between states are shown as 'k0 q k1'

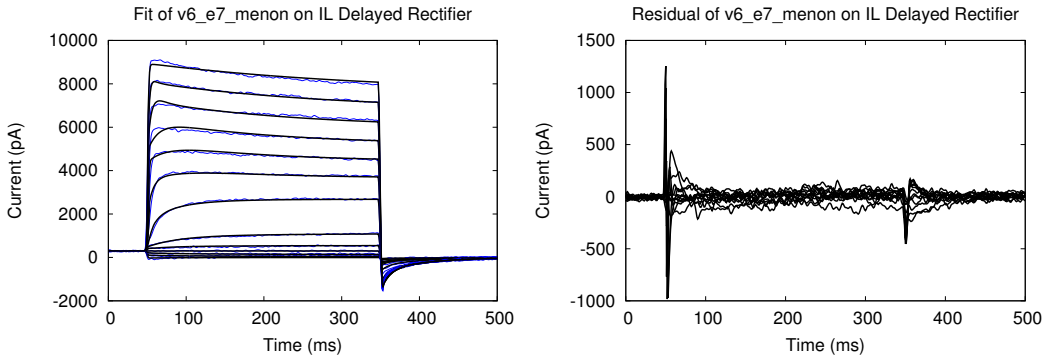


Figure 5.20: Selected fitting for *IL Delayed Rectifier* recording

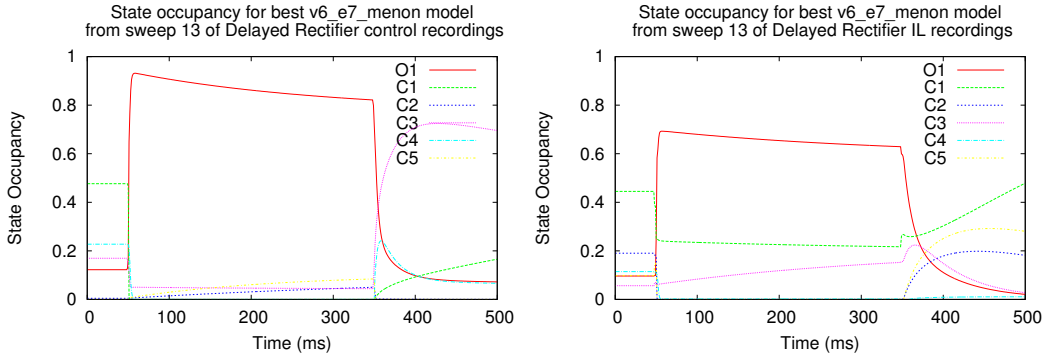


Figure 5.21: State occupancy plots of selected v6_e7_menon fittings for *Delayed Rectifier* recordings.

Chapter 6

Conclusions

We have investigated the problem of fitting ion channel recordings using hidden Markov models of undetermined topology. The major obstacle to producing high quality models is the difficulty of the optimization procedure, translating into a high failure rate for the optimization tools and a lack of certainty about the proximity of any optimization result to the global maxima. The difficulty of this problem and the uncertainty around generated optimization results was the major difficulty we were attempting to address.

Our strategy involved creating a single configuration to be used in each fitting and then performing multiple optimizations across a range of preselected configured topologies. Yet even with the simplest recordings the best fittings for a single configured topology spanned a large range of LL values with models from ancestor topologies regularly outperforming descendant topologies. Some configured topologies were clearly easier for the optimizer to fit on the given recordings, but this may be more a characteristic of the optimizer than the configured topologies, and in either case does not necessarily speak as to the ability of a properly fitted model to fit the data.

This meant that to effectively assess the groups of models we needed an approach that did more than look at the model with the best score. Instead we defined a selection criteria, the peak LL range, that could identify a group of models that generated fits of the same general quality. Within this group of models we applied a new metric, the *max deviation*.

The max deviation is a measure of the ability of a configured topology to show the same observed behaviour using different behaviours within its hidden states. The motivation is that if a model is at the limit of its expressive capability its hidden states will be constrained

in what behaviour they can exhibit to produce an optimal output. Generally only having one set of behaviours that delivers the optimal result for that topology will translate into a low max deviation. Conversely an overparameterized topology will be able to generate the optimal behaviour with multiple unique solutions. This will translate into a wide range of behaviour in the internal states, resulting in a high max deviation.

The procedure we developed around this metric was designed to determine if a configured topology had sufficient complexity to model the data based on a group of fittings from that configured topology. Moreover it would identify a range of models estimated to contain the configured topology most representative of the data.

This approach worked well on our experiments using simulated data. In four of five experiments it produced estimates which included the model that generated the data in question. On actual cell recordings the results were mixed. For both pairs of control and IL experiments it indicated consistent groups of configured topologies for both channel types, which would be expected if the IL treatment does not alter the underlying structure producing the recordings. However, for the *Delayed Rectifier* channel recordings the difficulty of the data was such that even with thousands of models we were unable to generate enough quality fittings in each topology to make a meaningful application of the metric. A significant issue we did not address is the degree to which adding states decreases the influence of each individual state, potentially increasing the ability of a highly constrained configured topology to show a large max deviation.

To our knowledge this is a novel approach and has potential for further development. In Future Work we outline several ideas that build upon our methods including a correction for additional states, a method that does not require a pre-determined list of topologies, and an investigation of the relationship to related topics such as variance.

Chapter 7

Future Work

7.1 Improved Topology Exploration Protocol

One of the major limitations of our current approach is the limited and somewhat arbitrary selection of topologies used. Instead of assembling a growing list of topologies one could attempt a search through the topology space.

The main hurdle in a search is the breadth of the topology space and the difficulty in properly evaluating the suitability of any single topology due to the high failure rate of optimizations. By relying on the rules in table 3.4 we can ascertain whether a model has sufficient complexity with more confidence, but the process is far too expensive to perform any significant exploration of the model space.

Instead we suggest an approach that focuses on aggressively increasing complexity until evidence of a global peak LL is reached. One strategy we suggest is an iterative approach that starts with a simple v2_e1 model. At each level we take the best model and considers all the descendants that can be generated by adding a single edge. We then start optimizing each child topology while attempting to evaluate the suitability of the topology for optimization. Ideally we can quickly determine the topologies that generate peak LL models at the highest rate and concentrate only on those topologies until we reach 10 peak LL models. If the models have a high max deviation it suggests we reached a global peak LL and may terminate, otherwise we continue to increase the complexity.

If the approach is successful it could supply a handful of models of similar structure and complexity that are all able to reproduce the data, and do so with topologies that are well

suited to the optimization problem, relative to other topologies of similar complexity.

7.2 Max Deviation Oriented Optimization

One of the main difficulties in measuring max deviation is the extent to which we are reliant on a random set of good models from the optimizer. We cannot solve this problem but we may be able to reduce its influence if we can increase the variety of models generated by the optimizer. One way to do this would be to include a max deviation related penalty in the objective function.

Take the function $D(A)$ that compute the largest deviation between a specific model A and another model B in the set (as opposed to max deviation which takes the further step of performing this calculation for all models A). If we include $D(A)$ in the objective function we can bias the optimizer toward models that show distinct behaviour and get closer to the true max deviation with fewer models. If $Peak_{cutoff}$ is the cutoff level for peak models, 1.05 in our standard formulation, then a potential objective function for the optimization would be:

$$LL - LL \times D(A) \times Peak_{cutoff} \tag{7.1}$$

This formulation will incentivize the optimizer to push $D(A)$ towards 1 and generate novel behaviours, potentially increasing the rate at which models in the peak LL range with a high max deviation are generated. Note this objective function would be used for optimization but peak LL models would still be determined by a pure LL objective function.

In addition to improving the generation rate of high max deviation models we suspect this objective function could potentially improve the optimization process in specific scenarios.

Considering a maximization problem where the LL surface contains a large hill containing a sub-optimal solution the optimizer regularly gets stuck in. The formulation in equation 7.1 alters the surface, lowering well explored regions such as the hill and encouraging the optimizer to explore novel regions of behaviour. By biasing the optimizer away from regions that were well explored by previous iterations we effectively improve territory explored by the optimization process as a whole.

7.3 Improved Optimizer Profile

We are aware of little research on the distribution of suboptimal fittings discussed in section 3.3. There are multiple benefits that could arise from a better understanding of the distribution of created models.

One of the first benefits is a better statistical model of the optimizer results could improve ability to categorize the performance of different configurations and understand how those results track between different configured topologies and data sets. When performing a large series of optimizations an automated pre-optimization stage could evaluate the performance of several configurations against the problem set and select the most appropriate one. A more ambitious strategy would be to evaluate optimizer configurations as they generated the results, effectively optimizing the optimization parameters themselves.

For either of these approaches to be feasible and useful one would need to develop a relatively accurate statistical model of the results distribution from a modest set of suboptimal results.

Of particular interest is how the distribution of suboptimal results may contain clues about the geography of the problem. In several instances our data suggested that a topology would have a higher failure rate the more underparameterized it was. Therefore a high failure rate in these simple topologies is may suggest the data requires at least two or more additional states to potentially fit, or at least that these topologies may be discarded as insufficient after fewer generations of fittings.

7.4 Relation to Bias and Variance

The bias-variance tradeoff is a major problem in modeling. Bias is the tendency of the models to show a systemic error while variance is their tendency to give different estimates. Ideally models should have both low bias and low variance, but in general as models grow more complex bias decreases and variance increases.

More formally an estimator is a rule that attempts to create estimates of values from a population based on samples from that population. The bias of the estimator is the difference between the average estimate and the true value, so an unbiased estimator will deliver estimates with the same mean as the actual data. While the variance refers to the variance in the estimates that the estimator delivers. High variance implies that the

estimator is overfitting the data or has underconstrained features, this is particularly an issue on data the models have not been fitted against.

If we create an estimator g for function f given dataset $\{x, y\}$ where $y = f(x) + \epsilon$, with ϵ being the measurement error. Then for every dataset $\{x, y\}$ the estimator $g(x)$ delivers slightly different estimates. The mean estimator over all possible datasets is $E[g(x)]$. Bias is defined as the error between the mean estimator and the true function.

$$bias = E[g(x)] - f(x)$$

If the mean estimator has no systemic difference from the true function it is said to be unbiased.

The variance of the estimator is defined in the same fashion as the variance of any function:

$$variance = E[(g(x) - E[g(x)])^2$$

It is the average squared difference between any specific estimator $g(x)$ and the mean estimator $E[g(x)]$.

Bias and variance are difficult to apply to ion channel modeling for several reasons. For one they are typically estimated by with multiple datasets by either taking multiple samples of the distribution or using cross-validation. But when modelling we typically only have one recording and the data points are not independent so cross-validation is inappropriate. Instead we would have to use other strategies such as creating synthetic protocols to see the models operate on fresh data.

We also run into the same problem as we did with max deviation where a substantial portion of the models are poorly fitted and provide a poor basis for analysis. We would likely need a device similar to the peak LL range to create a subset of models available for comparison. And even then we still have one of the problems we were trying to address with max deviation since the selection of models will be skewed by what the optimizer is best suited to generating.

We believe it would be worthwhile to investigate the applicability of bias and variance to the problem of ion channel modeling in conjunction with an approach such as filtering in the peak LL range. As well since both variance and max deviation generally increase with overparameterization and work on a similar concept it would be useful to study their

relationship in more detail.

7.5 Numerical Stability

Prior to using QUB Express our early investigations made extensive use of the simulation environment NEURON [5]. When comparing model simulations between the two environments we found that implementing the model formulation described by QUB Express in NEURON would lead to inconsistent results. Some models would behave almost identically but others would beak down completely.

We suspect the source of the discrepancy is the calculation of the transition matrix Q as described in section 2.6. The form of the equation to determine the transition rates $k0 * \exp(k1 * V_m)$ is extremely sensitive to changes in V_m. Running models fit to the live recordings from section 4.1.2 we discovered values in the Q matrix of 10^{19} or more. This lead to an error when calculating the flow rate matrix equation 2.8.

Matrix exponentials are difficult to accurately calculate [31], the method used by QUB Express uses an eigendecomposition of Q. However, the matrix generated by reversing the decomposition of Q contains significant discrepancies. We attempted to improve the stability of the models by adding a penalty term based on the size of the error to the LL calculation used by QUB Express during optimization. This resulted in models showing insufficient dynamic response to the voltage.

We believe it is important to understand the degree to which these issues exist in different ion channel simulation environments based on Markov processes and to understand what influence they may have on the reproducibility and interpretation of models in the literature.

Bibliography

[1] Osvaldo Alvarez, Carlos Gonzalez, and Ramon Latorre. COUNTING CHANNELS : A TUTORIAL GUIDE ON ION CHANNEL. *Advances in Physiology Education*, 26:327–341, 2002.

[2] C M Armstrong and F Bezanilla. Inactivation of the sodium channel. II. Gating current experiments. *The Journal of general physiology*, 70(5):567–590, November 1977.

[3] Leonard E Baum and Ted Petrie. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 1966.

[4] RH Richard H Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 1995.

[5] M.L Carnevale, N.T. and Hines. *The NEURON Book*. Cambridge University Press, Cambridge, UK, 2006.

[6] SH Chung and JB Moore. Characterization of single channel currents using digital signal processing techniques based on Hidden Markov Models. *Philosophical Transactions of The Royal Society B Biological Sciences*, 329(1254):265–285, 1990.

[7] D Colquhoun and AG Hawkes. Relaxation and fluctuations of membrane currents that flow through drug-operated channels. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 199(1135):231–62, 1977.

[8] D Colquhoun and AG Hawkes. On the stochastic properties of bursts of single ion channel openings and of clusters of bursts. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 300(1098):1–59, 1982.

[9] K Colquhoun, D.; Hawkes, A. G.; Srodzinski. Joint distributions of apparent open and shut times of single-ion channels and maximum likelihood fitting of mechanisms. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 354(1718):2555–2590, 1996.

[10] Willian C Davidon. Variable metric method for minimization. *SIAM Journal on Optimization*, 1(1):1–17, 1991.

[11] Pieter-Tjerk de Boer, Dirk Kroese, Shie Mannor, and Reuven Rubinstein. A Tutorial on the Cross-Entropy Method. *Annals of Operations Research*, 134(1):19–67, February 2005.

[12] B Efron and C Stein. The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596, 1981.

[13] IreneS. Gabashvili, BerndH.A. Sokolowski, CynthiaC. Morton, and AnneB.S. Giersch. Ion Channel Gene Expression in the Inner Ear. *Journal of the Association for Research in Otolaryngology*, 8(3):305–328, 2007.

[14] KR Gegenfurtner. PRAXIS: Brent's algorithm for function minimization. *Behavior Research Methods, Instruments, & Computers*, 24(4):560–564, 1992.

[15] Stuart Geman, Elie Bienenstock, and René Doursat. Neural Networks and the Bias/Variance Dilemma. *Neural Computation*, 4(1):1–58, January 1992.

[16] Meron Gurkiewicz and Alon Korngreen. A numerical approach to ion channel modelling using whole-cell voltage-clamp recordings and a genetic algorithm. *PLoS computational biology*, 3(8):e169, August 2007.

[17] O P Hamill, A Marty, E Neher, B Sakmann, and F J Sigworth. Improved patch-clamp techniques for high-resolution current recording from cells and cell-free membrane patches. *Pflügers Archiv European Journal of Physiology*, 391(2):85–100, August 1981.

[18] Bertil Hille. *Ion Channels of Excitable Membranes (3rd Edition)*. Sinauer Associates Inc 2001-07, 3rd editio edition, July 2001.

[19] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500, 1952.

[20] Green Peter J. Hodgson Matthew E. A. Investigating Markov model discrimination for ion channels. *In Proceedings of the Royal Society of London Series A*, 455:3425–3448, 1999.

[21] Youxing Jiang, Alice Lee, Jiayun Chen, Vanessa Ruta, Martine Cadene, Brian T Chait, and Roderick MacKinnon. X-ray structure of a voltage-dependent K+ channel. *Nature*, 423(6935):33–41, May 2003.

[22] Eric Jones, Travis Oliphant, Pearu Peterson, and Others. SciPy: Open Source Scientific Tools for Python. http://www.scipy.org/.

[23] Wael Khreich, Eric Granger, Ali Miri, and Robert Sabourin. A survey of techniques for incremental learning of HMM parameters. *Information Sciences*, 197:105–130, August 2012.

[24] Rachel J. Mackay. Estimating the order of a hidden markov model. *Canadian Journal of Statistics*, 30(4):573–589, December 2002.

[25] KL Magleby and BS Pallotta. Calcium dependence of open and shut interval distributions from calcium-activated potassium channels in cultured rat muscle. *The Journal of physiology*, pages 585–604, 1983.

[26] Vilas Menon, Nelson Spruston, and William L Kath. A state-mutating genetic algorithm to design ion-channel models. *Proceedings of the National Academy of Sciences of the United States of America*, 106(39):16829–34, September 2009.

[27] Claude Meunier and Idan Segev. Playing the Devil's advocate: is the Hodgkin-Huxley model useful? *Trends in Neurosciences*, 25(11):558–563, 2002.

[28] Lorin S Milescu, Gustav Akk, and Frederick Sachs. Maximum likelihood estimation of ion channel kinetics from macroscopic currents. *Biophysical journal*, 88(4):2494–515, May 2005.

[29] Lorin S Milescu, Tadashi Yamanishi, Krzysztof Ptak, Murtaza Z Mogri, and Jeffrey C Smith. Real-time kinetic modeling of voltage-gated ion channels using dynamic clamp. *Biophysical journal*, 95(1):66–87, July 2008.

[30] J. Milescu, L.S.; Nicolai, C.; Bannen and J. Milescu, L.S., Nicolai, C., Bannen. QuB Software. http://www.qub.buffalo.edu/, 2013.

[31] Cleve Moler and Charles Van Loan. Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later. *SIAM Review*, 45(1):3–49, January 2003.

[32] JA A Nelder and R Mead. A simplex method for function minimization. *Computer journal*, 7:308–313, 1965.

[33] F Qin, a Auerbach, and F Sachs. Estimating single-channel kinetic parameters from idealized patch-clamp data containing missed events. *Biophysical journal*, 70(1):264–80, January 1996.

[34] F Qin, a Auerbach, and F Sachs. Maximum likelihood estimation of aggregated Markov processes. *Proceedings. Biological sciences / The Royal Society*, 264(1380):375–83, March 1997.

[35] F Qin, a Auerbach, and F Sachs. A direct optimization approach to hidden Markov modeling for single channel kinetics. *Biophysical journal*, 79(4):1915–27, October 2000.

[36] F Qin, a Auerbach, and F Sachs. Hidden Markov modeling for single channel kinetics with filtering and correlated noise. *Biophysical journal*, 79(4):1928–44, October 2000.

[37] L Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[38] ST Roweis. Constrained Hidden Markov Models. *NIPS*, pages 1–7, 1999.

[39] Ivo Siekmann, Larry E Wagner, David Yule, Colin Fox, David Bryant, Edmund J Crampin, and James Sneyd. MCMC estimation of Markov models for ion channels. *Biophysical journal*, 100(8):1919–29, April 2011.

[40] Patrick L Stemkowski and Peter a Smith. Long-term IL-1β Exposure Causes Subpopulation Dependent Alterations in Rat Dorsal Root Ganglion Neuron Excitability. *Journal of neurophysiology*, 7(June), December 2011.

[41] Patrick Lester Stemkowski. *The effect of long-term interleukin-1beta exposure on sensory neuron electrical membrane properties : implications for neuropathic pain.* Ph.d., University of Alberta, 2011.

[42] AF Strassberg and LJ DeFelice. Limitations of the Hodgkin-Huxley formalism: Effects of single channel kinetics on transmembrane voltage dynamics. *Neural Computation*, 655, 1993.

[43] R.C. Vasko, a. El-Jaroudi, J.R. Boston, RC Vasko Jr, and El-Jaroudi Amro. An algorithm to determine hidden Markov model topology. *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, 6:3577–3580, 1996.

[44] Lalitha Venkataramanan and JL Walsh. Identification of hidden Markov models for ion channel currents. I. Colored background noise. *IEEE Transactions on Signal Processing*, 46(7):1901–1915, 1998.

[45] Shimin Wang, Vladimir E Bondarenko, Yu-jie Qu, Glenna C L Bett, Michael J Morales, Randall L Rasmusson, and Harold C Strauss. Time- and voltage-dependent components of Kv4.3 inactivation. *Biophysical journal*, 89(5):3026–41, November 2005.

[46] AR Willms, DJ Baro, and RM Harris-Warrick. An improved parameter estimation method for Hodgkin-Huxley models. *Journal of Computational Neuroscience*, 6(2):145–168, 1999.

[47] XH H Zhou, S Gao, and SL L Hui. Methods for comparing the means of two independent log-normal samples. *Biometrics*, 53(3):1129–1135, September 1997.

[48] Yu Zhou, John E Pearson, and Anthony Auerbach. Phi-value analysis of a linear, sequential reaction mechanism: theory and application to ion channel gating. *Biophysical journal*, 89(6):3680–5, December 2005.