UNIVERSITY OF CALIFORNIA, SAN DIEGO

Systematic Annealing Approach for Statistical Data Assimilation

A Dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Physics

by

Jingxin Ye

Committee in charge:

Professor Henry D. I. Abarbanel, Chair
Professor Daniel P. Arovas
Professor Gert Cauwenberghs
Professor Philip E. Gill
Professor Julius Kuti

2016

The Dissertation of Jingxin Ye is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____

Chair

University of California, San Diego

2016

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor Henry Abarbanel, for his guidance, support, and patience. His deep knowledge of and enthusiasm for physics was an inspiration to me.

I appreciate the valuable time that each of my dissertation committee members, Professors Daniel Arovas, Gert Cauwenberghs, Philip Gill, and Julius Kuti, has devoted.

I would like to thank all my fellow students in the research group: Uriel Morone, Daniel Rey, Nirag Kadakia, Paul Rozdeba, Michael Eldridge and Zhe An for making things more interesting and enjoyable. Thanks to Arij Daou for stimulating discussions and help with neuron model construction.

I must make some technical acknowledgement at this point regarding publications that have been used in the dissertation. Chapter 2, 3 and 4 are a reprint of the material it appears in Jingxin Ye, Daniel Rey, Nirag Kadakia, Michael Eldridge, Uriel I. Morone, Paul J. Rozdeba, Henry D. I. Abarbanel, John C. Quinn Systematic variational method for statistical nonlinear state and parameter estimation, *Physical Review E*, 92(5), 052901 (2015) and Jingxin Ye, Nirag Kadakia, Paul J. Rozdeba, Henry D. I. Abarbanel, John C Quinn, Precision Variational Approximations in Statistical Data Assimilation, *Nonlinear Processes in Geophysics*, 22 (2), 205-213 (2015). Chapter 5, in full, is a reprint of the material as it appears in Jingxin Ye, Paul J. Rozdeba, Uriel I. Morone, Arij Daou, Henry D. I. Abarbanel, Estimating the Biophysical Properties of Neurons with Intracellular Calcium Dynamics, *Physical Review E*, 89(6), 062714 (2014). I was the primary investigator and author of those papers.

The completion of this dissertation would not have been possible without

emotional support from family and friends. I especially appreciate the friendship of fellow graduate students and roommates. Most of all, I would like to thank my parents, grandma and aunt.

VITA

2007 - 2011	B.S. in Applied Physics University of Science and Technology of China (USTC)
2011 - 2012	M.S. in Physics University of California, San Diego (UCSD)
2011 - 2016	Ph.D. in Physics University of California, San Diego (UCSD)

PUBLICATIONS

Nirag Kadakia, Daniel Rey, Jingxin Ye, Henry D. I. Abarbanel, Symplectic Structure of Statistical Variational Data Assimilation, *Quarterly Journal of the Royal Meteorological Society*, submitted (2016)

Zhe An, Daniel Rey, Jingxin Ye, Henry D. I. Abarbanel Estimating the State of a Geophysical System with Sparse Observations: Time Delay Methods to Achieve Accurate Initial States for Prediction, *Nonlinear Processes in Geophysics Discussion*, in review (2016)

Jingxin Ye, Daniel Rey, Nirag Kadakia, Michael Eldridge, Uriel I. Morone, Paul J. Rozdeba, Henry D. I. Abarbanel, John C. Quinn Systematic variational method for statistical nonlinear state and parameter estimation, *Physical Review E*, 92(5), 052901 (2015)

Jingxin Ye, Nirag KadaKia, Paul J. Rozdeba, Henry D. I. Abarbanel, John C Quinn, Precision Variational Approximations in Statistical Data Assimilation, *Nonlinear Processes in Geophysics*, 22 (2), 205-213 (2015)

Jingxin Ye, Paul J. Rozdeba, Uriel I. Morone, Arij Daou, Henry D. I. Abarbanel, Estimating the Biophysical Properties of Neurons with Intracellular Calcium Dynamics, *Physical Review E*, 89(6), 062714 (2014)

Jingxin Ye, Bin Zhao, Jian Zheng, Extraction of Effective Ion Pair Interactions in Warm Dense Beryllium and Helium Plasmas within Integral Equation Theory, *Physics of Plasmas*, 18(3), 032701 (2011)

ABSTRACT OF THE DISSERTATION

Systematic Annealing Approach for Statistical Data Assimilation

by

Jingxin Ye

Doctor of Philosophy in Physics

University of California, San Diego, 2016

Professor Henry D. I. Abarbanel, Chair

Data assimilation transfers information from observations of a complex system to physically-based system models. Typically, the observations are noisy, the model has errors, and the initial state of the model is uncertain, so the data assimilation is statistical. In statistical data assimilation one evaluates the conditional expected values, conditioned on measurements, of interesting quantities on the path of a model through observation and prediction windows. This often requires working with very high dimensional integrals in discrete time descriptions of the observations and model dynamics, which become function integrals in the continuous time limit. Two familiar

methods for performing these integrals include (1) Monte Carlo calculations and (2) variational approximations using the method of Laplace plus perturbative corrections to the dominant contributions. We attend here to aspects of the Laplace approximation and develop an annealing method for locating the variational path satisfying the Euler-Lagrange equations that comprises the major contribution to the integrals. This begins with the identification of the minimum action path starting with a situation where the model dynamics is totally unresolved in state space, and the consistent minimum of the variational problem is known. We then proceed to slowly increase the model resolution, seeking to remain in the basin of the minimum action path, until a path that gives the dominant contribution to the integral is identified. After a discussion of some general issues, we give examples of the assimilation process for some simple, instructive models from the geophysical literature. Then we explore a slightly richer model of the same type with two distinct time scales. This is followed by a model characterizing the biophysics of individual neurons.

Chapter 1

Introduction

1.1 Data Assimilation

One type of data analysis procedure frequently encountered in a broad spectrum of scientific field is to combine observations of physical systems to numerical models. The numerical model may be created from empirical summary/assumptions, or the ideas of some underlying mechanisms. In either case, the common problem is that the number of observations of the actual system available for analysis is orders of magnitude smaller than the number of values required to specify the model, including the time-independent parameters in the model that cannot be determined from first principle and also some time-dependent dynamical state variables that cannot be measured directly. Data assimilation provides a systematic approach to transfer the information from the observations to the quantitative, predictive model of the observed physical system, which allows us to estimate of those unknown parameters and state variables, and predict the future behavior of the physical system.

Data assimilation has been applied to varieties of fields including meteorol-

ogy [30, 71], geochemistry [27], systems biology [69], and many others. Most of the physical systems can be expressed or converted as a set of first-order differential equations. The deterministic dynamics of the model state $\mathbf{x}$ is taken to satisfy

$$\frac{dx_a(t)}{dt} = F_a(\mathbf{x}(t));\ a = 1, 2, ..., D, \tag{1.1}$$

in continuous time, where D is the dimensionality of the model state.

In each temporal observation window, it is usual that only a sparse set of the D-dimensional model dynamical variables $\mathbf{x}(t)$ are measured. From L-dimensional observations $y_l(t_n);\ l = 1, 2, ..., L$ at times $t_n = \{t_0, t_1, ..., t_m = t_f\}$, we must estimate the full D-dimensional state $x_a(t);\ a = 1, 2, ..., D$. Typically the measurements are sparse, $L \ll D$; We must also estimate any unknown fixed parameters in the model or in the measurement functions $h_l(\mathbf{x}(t)),\ l = 1, 2, ..., L$ relating the model output $\mathbf{x}(t)$ to the observations $y_l(t)$.

We usually discrete the dynamical model at times $t_n = \{t_0, t_1, ..., t_m = t_f\}$ and write $x(n) \equiv x(t_n)$ interchangebly. And in discrete time Eq(1.1) becomes

$$x_a(n+1) = f_a(\mathbf{x}(n)). \tag{1.2}$$

One step mapping function f_a may depend on t if the time discretization is not uniform.

The use of measurements in the window $[t_0, t_f]$ to estimate N_p time-independent parameters $\mathbf{p} = \{p_1, p_2, \ldots, p_{N_p}\}$ and unknown states completes the model [3] and permits us to test or validate the model through predictions for $t > t_f$ where a selected metric compares new observations $\mathbf{y}(t > t_f)$ to new model outputs $\mathbf{h}(\mathbf{x}(t > t_f))$.

1.2 Probabilistic Formulation of Data Assimilation

Since the measurements are always noisy and the model always has errors, one requires a statistical description of the assimilation of information from the observations into the model. Such a description is based on estimating the conditional distribution of states and parameters $P(\mathbf{x}(t_n)|\mathbf{Y}(n))$ conditioned on measurements up to time t_n : $\mathbf{Y}(n) = \{\mathbf{y}(t_0), ..., \mathbf{y}(t_{n-1}), \mathbf{y}(t_n)\}$ to predict the conditional distribution for $t > t_f$.

Our focus is not on the evaluation of $P(\mathbf{x}(t_n)|\mathbf{Y}(n))$ itself, but on the quantities of physical (or biological) interest: the conditional expected value of functions $G(\mathbf{X})$ along the path $\mathbf{X} = \{\mathbf{x}(t_0), ...\mathbf{x}(t_{m-1}), \mathbf{x}(t_m), \mathbf{p}\}$ of the state through the observation window and beyond. $\mathbf{X}$ is a $(m+1)D+N_p$ dimensional vector. Writing the conditional probability distribution $P(\mathbf{X}|\mathbf{Y}) = P(\mathbf{x}(t_0), ..., \mathbf{x}(t_m)|\mathbf{y}(t_0), ..., \mathbf{y}(t_m))$, the expected value of $G(\mathbf{X})$ is

$$E[G(\mathbf{X})|\mathbf{Y}] = \frac{\int d\mathbf{X}\, G(\mathbf{X}) P(\mathbf{X}|\mathbf{Y})}{\int d\mathbf{X}\, P(\mathbf{X}|\mathbf{Y})}. \tag{1.3}$$

Important examples of $G(\mathbf{X})$ include the mean or expected path, in which case $G(\mathbf{X}) = \mathbf{X}$, moments about this expected path, and marginal distributions of, say, $x_b(t_k)$ in which case $G(\mathbf{X}) = \delta(\theta - x_b(k))$ giving $P(\theta)$.

1.2.1 Conditional Probability $P(\mathbf{X}|\mathbf{Y})$

To evaluate the conditional expected value of functions $G(\mathbf{X})$ with Eq.(A.2), we would like to figure out $P(\mathbf{X}|\mathbf{Y})$, the probability distribution of the path $\mathbf{X}$ conditioned on the measurements $\mathbf{Y}$. Before we derive the exact $P(\mathbf{X}|\mathbf{Y})$ forumation, let us prepare two ingredients needed to cook $P(\mathbf{X}|\mathbf{Y})$: the transition probability $P(\mathbf{x}(n)|\mathbf{x}(n-1))$

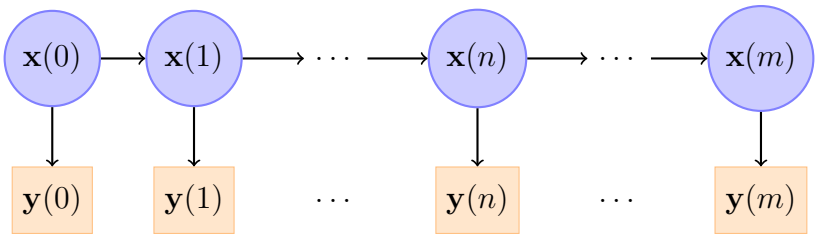

Figure 1.1: Dynamical model states and observations relationship diagram: $\mathbf{x}(n)$ represents the state variable of the model, and $\mathbf{y}(n)$ represents the obtained measurements at the discrete time $t = t_n$.

from $\mathbf{x}(n-1)$ to $\mathbf{x}(n)$, and the emission probability from the system state $\mathbf{x}(n)$ to the measurement time series $\mathbf{y}(n)$.

After we represent the state of the model at each time step $t_n = \{t_0, t_1, ..., t_m = t_f\}$ as D-dimensional vector $\mathbf{x}(n)$. The one-step mapping process from t_{n-1} to t_n is assumed to be Markovian. The time evolution represented by the conditional probability $P(\mathbf{x}(n)|\mathbf{x}(n-1))$ which only depends on $\mathbf{x}(n-1)$ but not any previous states before t_{n-1}. When there is no model error existing in Eq.(1.2), the transition from $\mathbf{x}(n-1)$ to $\mathbf{x}(n)$ is deterministic, and $P(\mathbf{x}(n)|\mathbf{x}(n-1))$ is reduced to be delta function. The distribution will be broader than delta function as the model error ϵ increases. One-step mapping function Eq.(1.2) becomes

$$x_a(n+1) = f_a(\mathbf{x}(n)) + \epsilon. \tag{1.4}$$

When model error is Gaussian distributed, i.e. $\epsilon \sim \mathcal{N}(0, \Sigma_f)$,

$$P(\mathbf{x}(n)|\mathbf{x}(n-1)) = P(\epsilon) \propto \exp\left[||\mathbf{x}(n+1) - \mathbf{f}(\mathbf{x}(n))||_{\Sigma_f^{-1}}\right]. \tag{1.5}$$

Another important component is the probabilistic relationship between the model state and the measurement time series. Without losing generality, we can

assume the measurement function $\mathbf{y}(n) = \mathbf{h}(\mathbf{x}(n))$ is simplified as identity function,

$$\mathbf{y}(n) = \mathbf{x}(n) + \eta,$$

where η is measurement noise. The noise will be distributed as some known distribution which is determined by many factors like experimental instruments. The emission probability from $\mathbf{x}(n)$ to $\mathbf{y}(n)$ is also assumed be Markovian, which means the previous states $\mathbf{x}(t < n)$ have no effects on the measurement $\mathbf{y}(n)$. For Gauassian case with $\eta \sim \mathcal{N}(0, \Sigma_m)$,

$$P(\mathbf{x}(n)|\mathbf{y}(n)) = P(\eta) \propto \exp\left[||\mathbf{y}(n) - \mathbf{x}(n)||_{\Sigma_m^{-1}}\right]. \tag{1.6}$$

The formulation of $P(\mathbf{X}|\mathbf{Y})$ can be derived by employing its Markovian propertiy and the definition of conditional probability repeatedly. Applying the definition of conditional probability $P(A|B) = P(A, B)/P(B)$, we have

$$\begin{aligned} P(\mathbf{X}|\mathbf{Y}) =& P(\mathbf{x}(0), \ldots, \mathbf{x}(m)|\mathbf{y}(0), \ldots, \mathbf{y}(m)) \\ =& \frac{P(\mathbf{x}(0), \ldots, \mathbf{x}(m), \mathbf{y}(0), \ldots, \mathbf{y}(m))}{P(\mathbf{y}(0), \ldots, \mathbf{y}(m))} \\ =& \frac{P(\mathbf{x}(0:m), \mathbf{y}(0:m))}{P(\mathbf{y}(0:m))} \end{aligned} \tag{1.7}$$

where the semicolon are used to simplify the time notations $\mathbf{x}(0:m) = \{\mathbf{x}(0), \ldots, \mathbf{x}(m)\}$. Because $\mathbf{y}(0:m)$ are measurements, the denominator term $P(\mathbf{y}(0:m))$ is known to us and will be dropped as constant later. We will only forcus on the numerator term $P(\mathbf{x}(0:m), \mathbf{y}(0:m))$. Firstly, let us calculate the recursive formula of

$P(\mathbf{x}(0:n), \mathbf{y}(0:n))$ in terms of $P(\mathbf{x}(0:n-1), \mathbf{y}(0:n-1))$,

$$\begin{aligned} P(\mathbf{x}(0:n), \mathbf{y}(0:n)) =& P(y(n)|\mathbf{x}(0:n), \mathbf{y}(0:n-1))\ P(\mathbf{x}(0:n), \mathbf{y}(0:n-1)) \\ =& P(\mathbf{y}(n)|\mathbf{x}(0:n), \mathbf{y}(0:n-1))\ P(\mathbf{x}(n)|\mathbf{x}(0:n-1), \mathbf{y}(0:n-1)) \\ & \times P(\mathbf{x}(0:n-1), \mathbf{y}(0:n-1)) \end{aligned}$$

where the definition of conditional probability $P(A, B) = P(A|B)P(B)$ is used. Since $\mathbf{y}(n)$, $\mathbf{x}(n)$ only depend on $\mathbf{x}(n)$ and $\mathbf{x}(n-1)$, respectively. By Markovian propertiy, the two terms can be simplified as

$$\begin{aligned} P(\mathbf{y}(n)|\mathbf{x}(0:n), \mathbf{y}(0:n-1)) &= P(\mathbf{y}(n)|\mathbf{x}(n)), \\ P(\mathbf{x}(n)|\mathbf{x}(0:n-1), \mathbf{y}(0:n-1)) &= P(\mathbf{x}(n)|\mathbf{x}(n-1)). \end{aligned}$$

Therefore, we have

$$P(\mathbf{x}(0:n), \mathbf{y}(0:n)) = P(\mathbf{y}(n)|\mathbf{x}(n))\ P(\mathbf{x}(n)|\mathbf{x}(n-1))\ P(\mathbf{x}(0:n-1), \mathbf{y}(0:n-1)). \tag{1.8}$$

With $P(\mathbf{x}(0), \mathbf{y}(0)) = P(\mathbf{y}(0)|\mathbf{x}(0))P(\mathbf{x}(0))$, by induction, $P(\mathbf{x}(0:m), \mathbf{y}(0:m))$ can be written as

$$P(\mathbf{x}(0:m), \mathbf{y}(0:m)) = \prod_{n=0}^{m} P(\mathbf{y}(n)|\mathbf{x}(n))\ \prod_{n=1}^{m} P(\mathbf{x}(n)|\mathbf{x}(n-1))\ P(\mathbf{x}(0)). \tag{1.9}$$

The conditional probability Eq.(1.7) can expressed as

$$P(\mathbf{X}|\mathbf{Y}) = \frac{\prod_{n=0}^{m} P(\mathbf{y}(n)|\mathbf{x}(n))\ \prod_{n=1}^{m} P(\mathbf{x}(n)|\mathbf{x}(n-1))\ P(\mathbf{x}(0))}{P(\mathbf{y}(0:m))}. \tag{1.10}$$

1.2.2 Data Assimilation Action

This expression for $P(\mathbf{X}|\mathbf{Y}) \propto \exp(-A_0)$ defines the "action" $A_0(\mathbf{X})$. We do not further explicitly show the dependence of the action on the measurements. In discrete time this is an integral of dimension D times the number of discrete time steps in the observation plus prediction windows. In continuous time it is a functional path integral [36, 103].

The action $A_0(\mathbf{X})$ has the exact representation [3] in discrete time

$$\begin{aligned} A_0(\mathbf{X}) = &- \sum_{n=0}^{m-1} \ln[P(\mathbf{x}(n+1)|\mathbf{x}(n))] - \ln[P(\mathbf{x}(0))] \\ &- \sum_{n=0}^{m} \ln\left[P(\mathbf{y}(n)|\mathbf{x}(n)\right] \\ &+ \text{terms independent of } \mathbf{X}. \end{aligned}$$

$P(\mathbf{x}(n+1)|\mathbf{x}(n))$ is the transition probability for the state $\mathbf{x}(n)$ at time t_n to arrive at the state $\mathbf{x}(n+1)$ at time t_{n+1}, and $P(\mathbf{x}(0))$ is the distribution of the state at time t_0 when observations commence. The dynamics moving the model state $\mathbf{x}(n)$ through time resides in $P(\mathbf{x}(n+1)|\mathbf{x}(n))$.

The discretization in time of Eq.(1.1) can be explicit or implicit. Either choice defines an action $A_0(\mathbf{X})$ as a function of the components of $\mathbf{X}$. Throughout this thesis, the trapezoidal rule is used to discretize the model equations,

$$x_a(n+1) = x_a(n) + \frac{\Delta t}{2}[F_a(\mathbf{x}(n+1)) + F_a(\mathbf{x}(n))]. \tag{1.11}$$

The model equations could then be written as a function of both $\mathbf{x}(n)$ and $\mathbf{x}(n+1)$: $g_a(\mathbf{x}(n), \mathbf{x}(n+1), \mathbf{p}) = 0$. When there is no model error, $P(\mathbf{x}(n+1)|\mathbf{x}(n)) =$

$\delta^{D+N_P}(\mathbf{g}(x(n), \mathbf{x}(n+1), \mathbf{p}))$.

Once a model has been selected by physical considerations, and choices are made for the distributions of errors in the measurements and of errors in the model, the central challenge of statistical data assimilation is the estimation of the path integral Eq. (A.2).

One approach is to use Monte Carlo methods for evaluating the high dimensional integral [81, 37, 65, 13].

In this thesis we investigate results associated with the estimation of the integral Eq. (A.2) using Laplace's method [66]. This is a variational calculation seeking extremum paths $\mathbf{X}^q$ of the action, where the Jacobian

$$\left.\frac{\partial A_0(\mathbf{X})}{\partial \mathbf{X}}\right|_{\mathbf{X}=\mathbf{X}^q} = 0, \quad q = 0, 1, \ldots . \tag{1.12a}$$

and the Hessian

$$\left.\frac{\partial^2 A_0(\mathbf{X})}{\partial \mathbf{X}^2}\right|_{\mathbf{X}=\mathbf{X}^q} \text{ is positive definite.} \tag{1.12b}$$

Laplace's method also allows the evaluation of corrections to Eq. (A.2) using any of the $\mathbf{X}^q$ as the leading approximation for the expected value. Furthermore, when $\mathbf{f}(\mathbf{x})$ is not linear in the model state variables, there may be multiple solutions to Eq. (1.12), and we must select which among them provide the most important contributions to the integral. Finding the extremum path is widely known as 4DVar [30, 71] in the geophysical literature.

Paths with distinct values of $A_0(\mathbf{X})$ lead to exponentially different contributions

to the expected value path integral Eq. (A.2). The path which gives the smallest value of the action,

$$A_0(\mathbf{X}^0) = \min_{\mathbf{X}^q} A_0(\mathbf{X}^q), \tag{1.13}$$

$\mathbf{X}^0$ is the conditional mode of the distribution and also the maximum likelihood estimate. When the action level $A_0(\mathbf{X}^0)$ is much less than any other action level coming from paths $\mathbf{X}^{q\neq 0}$, it exponentially dominates the integral.

1.3 Challenge of Action Minimization

We now discuss the stability issues associated with minimizing the action A_0 when the dynamical model Eq.(1.2) is enforced. In the case, there is no model error, once the initial condition $\mathbf{x}(0)$ of the system is determined, other state variables can be obtained by

$$\mathbf{x}(n) = \mathbf{f}^n[\mathbf{x}(0)].$$

The action $A_0(X)$ can be rewritten as function of the initial condition $\mathbf{x}(0)$. The minimization problem turns to be

$$\begin{aligned} \underset{\mathbf{x}(0)}{\text{minimize}} \quad & A_0(\mathbf{x}(0)) = \sum_{n=0}^{m}\sum_{l=1}^{L}[x_l(n) - y_l(n)]^2 \\ \text{subject to} \quad & \mathbf{x}(n) = \mathbf{f}[\mathbf{x}(n-1)],\ n = 1, \ldots, m. \end{aligned}$$

The necessary condition for the minimum of $A_0(\mathbf{x}(0))$ is

$$\frac{\partial A_0(\mathbf{x}(0))}{\partial \mathbf{x}(0)} = 0,$$
$$\frac{\partial A_0(\mathbf{x}(0))}{\partial \mathbf{p}} = 0.$$

In the derivatives above, it contains $\partial \mathbf{x}(n)/\partial \mathbf{x}(0)$ satisfying

$$\frac{\partial \mathbf{x}(n)}{\partial \mathbf{x}(0)} = \frac{\partial \mathbf{f}(\mathbf{x}(n-1))}{\partial \mathbf{x}(n-1)} \frac{\partial \mathbf{x}(n)}{\partial \mathbf{x}(0)} = \frac{\partial \mathbf{f}(\mathbf{x}(n-1))}{\partial \mathbf{x}(n-1)} \frac{\partial \mathbf{f}(\mathbf{x}(n-2))}{\partial \mathbf{x}(n-2)} \cdots \frac{\partial \mathbf{f}(\mathbf{x}(0))}{\partial \mathbf{x}(0)}$$

Let $Df(\mathbf{x}(n))$ denote the Jacobian matrix $\partial \mathbf{f}(\mathbf{x}(n-1))/\partial \mathbf{x}(n-1)$. In many natural systems, lots of nonlinear dynamical systems that are highly sensitive to initial conditions are referred as chaotic systems, such as weather and climate. In chaotic systems, the Jacobian matrix $Df(\mathbf{x}(n))$ has its largest eigenvalue larger than 1, which leads the value of $\partial \mathbf{x}(n)/\partial \mathbf{x}(0)$ increasing exponentially as function of n. This can give rise to the instability of minimization manifold quite irregular.(Fig.5.3)

In addition to the instability caused by chaos, as we will show in the latter chapters, the sensitivity to parameters $\mathbf{p}$ and the dynamics of different scales also causes similar irregular behavior for the minimization problem.

Therefore, the search for minima of action $A_0(\mathbf{X}^0)$ in nonlinear problems requires some care. In the core of this thesis, we will show how to use annealing approach to search for the maximum likelihood estimate $\mathbf{X}^0$.

In Chapter 2, our goal is to expand on the details of an annealing method to locate the saddle paths with the smallest action. A simple dynamical model Lorenz96 $D = 5$ is employed to illustrate the details of the annealing method. Additionally, in Chapter 3 we use the annealing method to explore several interesting nonlinear

dynamical models. These include Lorenz96 model in higher dimensionally $D = 20$, a model with combined "fast" and "slow" time scales of the dynamics as well as a standard Hodgkin-Huxley model of an isolated neuron. The latter is a prelude for more complex, biophysically realistic neuron models both in isolation and within functional networks with biophysical connections among them. No model is perfect, it is meaningful to study the action levels for wrong models. In the last section of Chapter 3, models with totally wrong dynamics, missing terms and incorrect coefficients are investigated.

Further, we estimate the corrections to the approximation of retaining only $\mathbf{X}^0$ as dominating the integral Eq. (1.12) in Chapter 4. Since the correction may fail in non-Gaussian model error case, the effects of distribution tail on the action levels are discussed. We also compare this work with the QSVA method [78], which seeks to solve a related problem by manipulating the length of the assimilation window.

Chapter 5 demonstrates the application of data assimilation methods to a biophysical neuron model with detailed calcium dynamics.

Chapter 2

Annealing Method

In this chapter, we give the formulation of the minimization problem in both discrete time and continuous time, and then present the details of our annealing approach. Finally the method is illustrated using Lorenz96 $D = 5$ model.

2.1 Formulation of Action A_0 in Discrete Time and in Continuous Time

2.1.1 Discrete Time

To simplify the discussion we make two familiar assumptions about how the measurement errors and the model errors enter the expression for the action $A_0(\mathbf{X})$: the errors in each are taken to be distributed as a Gaussian,

- the measurement error enters with an inverse covariance matrix $\mathbf{R}_m(l,k,t) = R_m(l,t)\delta_{lk},\ k,l = 1,2,...,L,$
- and the model error enters with an inverse covariance matrix $\mathbf{R}_f(a,b) = R_f(a)\delta_{ab},$

$a, b = 1, 2, ..., D.$

As the dynamics $\mathbf{x}(n+1) = \mathbf{f}(\mathbf{x}(n))$ is nonlinear, the overall path integral is not Gaussian with this choice for the error distributions.

In the presence of Gaussian additive model error, the dynamics satisfies the D-dimensional stochastic discrete time map

$$x_a(n+1) = f_a(\mathbf{x}(n)) + R_f(a)^{-1/2}\eta_a(n), \tag{2.1}$$

and each component $\eta_a(n)$ is Gaussian distributed as $\mathcal{N}(0,1)$. Here we assume there is no cross-covariance between different state variables, and the case with cross-covariances needs further study. We also assume the measurement function $h_l(\mathbf{x}(t)) = x_l(t_n),\ n = 0, 1, \ldots, m.$

The Gaussian error action $A_0(\mathbf{X})$ in discrete time takes the form

$$\begin{aligned} A_0(X) = &\sum_{n=0}^{m}\sum_{l=1}^{L}\frac{R_m(l,n)}{2}[x_l(n) - y_l(n)]^2 - \ln[P(\mathbf{x}(0))] \\ &+ \sum_{n=0}^{m-1}\sum_{a=1}^{D}\frac{R_f(a)}{2}[x_a(n+1) - f_a(\mathbf{x}(n))]^2. \end{aligned} \tag{2.2}$$

The distribution of initial states $P(\mathbf{x}(0))$ in the action is often assumed to be uniformly distributed or Gaussian distributed. For the uniform distribution case $-\ln[P(\mathbf{x}(0)]$ is a constant and cancels between the numerator and denominator of expected values Eq. (A.2). When $P(\mathbf{x}(0))$ is Gaussian, suppose the variation of $\mathbf{x}(0)$ is given about some base state $\mathbf{x}_{\text{base}}$, so $-\ln[P(\mathbf{x}(0)] = (\mathbf{x}(0) - \mathbf{x}_{\text{base}})^2 R_{\text{base}}/2$. This has the form of the measurement term evaluated at $n = 0$. This expression can be incorporated into the term with coefficient R_m in the action.

We no longer display $-\ln[P(\mathbf{x}(0))]$ in the following discussion. The resulting action Eq. (A.1) we call the Gaussian error action. Many other actions may be of physical interest, and they depend in detail on our representation of errors in the measurements and errors in the model.

2.1.2 Continuous Time

Although all calculations are performed in discrete time, we use continuous time to gather insight into the saddle paths for the action, and return to discrete time with the lessons in mind. Let time become continuous between the initiation of observations at time t_0 and the completion of measurements at time t_f, we identify the action in continuous time as

$$A_0(\mathbf{x}(t)) = \int_{t_0}^{t_f} dt\, \mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t) \tag{2.3}$$

where the Lagrangian $\mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t)$, also called the Onsager-Machlup functional [77], is

$$\mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t) = \sum_{l=1}^{L} \frac{R_m(l,t)}{2}[x_l(t) - y_l(t)]^2 + \sum_{a=1}^{D} \frac{R_f(a)}{2}[\dot{x}_a(t) - F_a(\mathbf{x}(t))]^2.$$

$R_m(l,t)$ is nonzero only near the observations times $t \approx t_n$.

The transition from discrete to continuous time has a subtlety which we note here, then bypass as we will return to discrete time for all of our examples and for any applications. As carefully explained in the book of Zinn-Justin [103] and the papers [52, 45] there is a Jacobian involved in the transformation from discrete to continuous time which affects the action through a term involving the divergence of

the vector field $\mathbf{F}(\mathbf{x})$. It adds the term

$$\Theta(0)\int_{t_0}^{t_f} dt\,\nabla_x\cdot\mathbf{F}(\mathbf{x}(t)), \tag{2.4}$$

to the action, where $\Theta(x)$ is the Heaviside function. The value of the quantity $\Theta(0)$ depends on the stochastic discretization scheme we choose. $\Theta(0) = 0, 1/2$ corresponding to the Itô and the Stratonovich scheme, respectively. As noted, we work with the discrete time version of the path, so we do not further consider this term.

2.2 Laplace's Approximation: Saddle Paths of the Action $A_0(\mathbf{x}(t))$

2.2.1 General Results in Continuous Time

In continuous time the expected values are written as [36, 103]

$$E[G(\mathbf{x}(t)|\mathbf{Y}] = \frac{\int \mathcal{D}x(t)G(\mathbf{x}(t))\exp[-A_0(\mathbf{x}(t))]}{\int \mathcal{D}x(t)\exp[-A_0(\mathbf{x}(t))]}, \tag{2.5}$$

following Eq. (A.2). There are no restrictions on the variation of the endpoints in this expected value. This is seen directly in the discrete time formulation where we integrate $G(\mathbf{X})$ over all locations on the path including the initial point $\mathbf{x}(t_0)$ and the ending point $\mathbf{x}(t_m)$.

The expansion of the action Eq. (2.3) about saddle paths $\mathbf{x}^q(t)$; $q = 1, 2, \ldots,$ satisfying

$$\left.\frac{\delta A_0(\mathbf{x}(t))}{\delta\mathbf{x}(t)}\right|_{\mathbf{x}(t)=\mathbf{x}^q(t)} = 0, \tag{2.6}$$

at $\mathbf{x}(t) = \mathbf{x}^q(t)$ yields, writing $\delta\mathbf{x}(t) = \mathbf{x}(t) - \mathbf{x}^q(t)$,

$$\begin{aligned} A_0(\mathbf{x}(t)) =& A_0(\mathbf{x}^q(t)) + \delta x_a(t)\, \frac{\partial \mathcal{L}(\mathbf{x}, \dot{\mathbf{x}}, t)}{\partial \dot{x}_a}\bigg|_{t_0}^{t_f} \\ &+ \int_{t_0}^{t_f} dt\ \delta x_a(t)\, \left(\frac{\partial \mathcal{L}(\mathbf{x}, \dot{\mathbf{x}}, t)}{\partial x_a} - \frac{d}{dt} \frac{\partial \mathcal{L}(\mathbf{x}, \dot{\mathbf{x}}, t)}{\partial \dot{x}_a} \right)\bigg|_{\mathbf{x}(t)=\mathbf{x}^q(t)} \\ &+ \frac{1}{2} \int_{t_0}^{t_f} dt\, \delta x_a(t) M_{ab}(\mathbf{x}^q(t), \dot{\mathbf{x}}^q(t), t, d/dt) \delta x_b(t) + \cdots . \end{aligned} \tag{2.7}$$

The first variation of the action must be zero as a necessary condition for a possible minimum [68]. The integration by parts term leading to the Euler-Lagrange equations requires at the endpoints that

$$\delta x_a(t)\, \frac{\partial \mathcal{L}(\mathbf{x}^q(t), \dot{\mathbf{x}}^q(t), t)}{\partial \dot{x}_a^q(t)}\bigg|_{t_0}^{t_f} = 0, \tag{2.8}$$

which leads to, as $\delta\mathbf{x}(t)$ is not zero,

$$\frac{\partial \mathcal{L}(\mathbf{x}^q(t), \dot{\mathbf{x}}^q(t), t)}{\partial \dot{x}_a^q(t)}\bigg|_{t_0}^{t_f} = 0, \tag{2.9}$$

and are boundary conditions on the saddle path $\mathbf{x}^q(t)$. These are known as "natural boundary conditions" [55, 62, 35, 68]. This is a necessary condition for the minimum path.

The second variation of the action contains the term in $(\delta\mathbf{x}(t), \delta\dot{\mathbf{x}}(t))$ space

$$\begin{aligned} &\delta x_a(t) \frac{\partial^2 \mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t)}{\partial x_a(t)\, \partial x_b(t)} \delta x_b(t) + 2\delta x_a(t) \frac{\partial^2 \mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t)}{\partial x_a(t)\, \partial \dot{x}_b} (t) \delta \dot{x}_b(t) \\ &+ \delta \dot{x}_a(t) \frac{\partial^2 \mathcal{L}(\mathbf{x}(t), \dot{\mathbf{x}}(t), t)}{\partial \dot{x}_a(t)\, \partial \dot{x}_b(t)} \delta \dot{x}_b(t). \end{aligned} \tag{2.10}$$

This is familiar from many books on the calculus of variations [55, 62, 35, 68]. Sufficient

conditions for a minimum of the action are discussed in [55, 62, 35].

The Euler-Lagrange equations determining $\mathbf{x}^q(t)$ are

$$R_f(a)\left\{\frac{d}{dt}[\dot{x}_a(t) - F_a(\mathbf{x}(t))] + [\dot{x}_b(t) - F_b(\mathbf{x}(t))]\frac{\partial F_b(\mathbf{x}(t))}{\partial x_a}\right\} = R_m(l,t)[x_l(t) - y_l(t)]\delta_{al}. \tag{2.11}$$

Since the variations of the locations in state space $\delta\mathbf{x}(t_f)$ and $\delta\mathbf{x}(t_0)$ are unconstrained and independent when we are evaluating the expected value of a function on the path $G(\mathbf{X})$, the Euler-Lagrange equation is a second order differential equation in time with endpoint conditions

$$\frac{\partial\mathcal{L}(\mathbf{x}(t),\dot{\mathbf{x}}(t),t)}{\partial\dot{x}_a} = R_f(a)[\dot{x}_a(t) - F_a(\mathbf{x}(t))] = 0, \tag{2.12}$$

at $t = t_0, t_f$.

These Euler-Lagrange equations have been considered by Bröcker [15, 17, 16] as well as being considered over some years in the work of Bennett [7, 21]. We will discuss the relation of our work with the work by Bröcker below.

2.2.2 Boundary Conditions on the Euler-Lagrange Equations

In most treatments of variational problems in data assimilation, one addresses a slightly different set of boundary conditions for the Euler-Lagrange equations defining the saddle path. That is because a slightly different question is asked.

To see the difference we return to discrete time. We are concerned with the expected value of a function on the path which includes integration over both the final point in state space $\mathbf{x}(t_f)$ as well as the initial point in state space $\mathbf{x}(t_0)$. We can

rewrite the numerator of the expected value of a function on the path $\mathbf{X} = \{\mathbf{X}', \mathbf{x}(0)\}$ as

$$\int d\mathbf{X}' G(\mathbf{X}', \mathbf{x}(0)) K(\mathbf{X}; t_f; \mathbf{x}(0), t_0)\, d^D x(0). \tag{2.13}$$

If we hold off in performing the integral over the initial state at the fixed time t_0, then the Laplace approximation which includes a variational principle for the quantity $K(\mathbf{X}; t_f; \mathbf{x}(0), t_0)$ has natural boundary conditions associated with the end point $\mathbf{x}(t_f)$ which is integrated over and not fixed, and it has the restriction that $\delta\mathbf{x}(t_0) = 0$ as $\mathbf{x}(0)$ is not integrated over in the variational principle.

This leads to the common statement that the relevant path has a fixed initial point and a free end point, changing the two point boundary value problem to be addressed to the one stated in the literature [21]. Each choice of boundary conditions is correct for the question posed.

2.2.3 Finding Saddle Paths

In continuous time, the saddle path condition is the two point boundary value problem we have described in Eq. (2.11) and Eq. (2.12). There are many discussions of how to numerically solve these [6, 54, 90], and in a sense we now use the collocation solution method discussed in these references by our return to the discrete time problem.

If we begin with the Gaussian error action

$$A_0(\mathbf{X}) = \sum_{n=0}^{m} \sum_{l=1}^{L} \frac{R_m(l,n)}{2} [x_l(n) - y_l(n)]^2 + \sum_{n=0}^{m-1} \sum_{a=1}^{D} \frac{R_f(a)}{2} [x_a(n+1) - f_a(\mathbf{x}(n))]^2, \tag{2.14}$$

finding saddle paths, where $\partial A_0(\mathbf{X})/\partial \mathbf{X} = 0$, of this nonlinear function at sizable R_f

entails a search in a high dimensional space in which the saddle paths $\mathbf{X}^q$ are located in narrow, possibly deep, valleys not easily found by an arbitrary selection of an initial path for a numerical optimization routine [78, 81].

We have investigated direct searches of saddle paths for this form of the action using a quasi-Newton BFGS method [80] and the public domain optimization program [95] called IPOPT. For the BFGS method, the analytical form of the first-order derivative for the action Eq. (2.14) is provided to optimization routines; When IPOPT is used, in addition to the analytical gradient, the Hessian matrix is also presented in analytical form to IPOPT, both of which are obtained by using a Python script we developed [100]. In each case the paths found via a direct search from a more or less arbitrary initial selection were not correct. The metric for 'correct' is whether the estimated parameters for the model and the full state of the model at the end of an estimation window, $\mathbf{x}(t_f)$, give good predictions for $t > t_f$. We will see some aspects of this as we begin to explore examples.

2.2.4 First-order Optimality Conditions of Continuous and Discrete Action

The Euler-Lagrangian equation Eq.(2.11) and the boundary conditions Eq.(2.12) can be explicitly written as

$$\sum_{a=1}^{D} R_f(a)\frac{\partial F_a(t)}{\partial x_b}[-\dot{x}_a(t) + F_a(t)] - R_f(b)\ddot{x}_b(t) + R_f(b)\sum_{a=1}^{D}\frac{\partial F_b(t)}{\partial x_a}\dot{x}_a(t) = 0 \quad (2.15)$$

$$R_f(b)[\dot{x}_b(t=0) - F_b(\mathbf{x}(t=0))] = R_f(b)[\dot{x}_b(t=t_m) - F_b(\mathbf{x}(t=t_f m))] = 0 \quad (2.16)$$

Actually, the first-order optimality conditions of action in discrete action is the same as the Euler-Lagrangian equation in continuous time. We will start from the first-order optimality conditions of action in discrete action and recover Eq.(2.15) and Eq.(2.16).

In the discretized case, the trapezoidal integration scheme is used for center points and the forward Euler scheme at $t = 0$ and the backward Euler scheme at $t = m\Delta t$. The measurement error terms are easy to deal with since they are quadratic in $\mathbf{x}$. We omit the measurement terms in the following discussion.

Center Points The derivative of A_0 in respect to $x_b(n)$ for $0 < n < m$,

$$0 = \frac{\partial}{\partial x_b(n)} \sum_{n=0}^{m-1} \sum_{a=1}^{D} \frac{R_f(a)}{2} [x_a(n+1) - f_a(x(n))]^2.$$

Keeping the terms that contains $x_b(n)$ only,

$$\begin{aligned} 0 = & \sum_{a=1}^{D} R_f(a)[x_a(n) - f_a(x(n-1))] \left[\delta_{ab} - \frac{\partial f_a(n-1)}{\partial x_b(n)}\right] \\ & + \sum_{a=1}^{D} R_f(a)[x_a(n+1) - f_a(x(n))] \left[-\frac{\partial f_a(n)}{\partial x_b(n)}\right] \end{aligned} \tag{2.17}$$

Under the trapezoidal integration rule, $f_a(n+1) = x_a(n) + \frac{\Delta t}{2}[F_a(\mathbf{x}(n+1)) + F_a(\mathbf{x}(n))]$, and the two elementary derivatives in Eq.(2.17) are

$$\begin{aligned} \frac{\partial f_a(n-1)}{\partial x_b(n)} &= \frac{\Delta t}{2} \frac{\partial F_a(\mathbf{x}(n))}{\partial x_b(n)} \\ \frac{\partial f_a(n)}{\partial x_b(n)} &= \delta_{ab} + \frac{\Delta t}{2} \frac{\partial F_a(\mathbf{x}(n))}{\partial x_b(n)} \end{aligned}$$

Substitute them into Eq.(2.17), and divide Δt^2 on both sides:

$$\begin{aligned}
0 = &- R_f(b)\frac{x_b(n+1) - 2x_b(n) + x_b(n-1)}{\Delta t^2} + R_f(b)\frac{F_b(\mathbf{x}(n+1)) - F_b(\mathbf{x}(n-1))}{2\Delta t} \\
&- \sum_{a=1}^{D} R_f(a)\left[\frac{x_a(n+1) - x_a(n-1)}{2\Delta t} - F_a(\mathbf{x}(n))\right]\frac{\partial F_a(n)}{\partial x_b(n)} \\
&+ \sum_{a=1}^{D} R_f(a)\frac{1}{4}(F_a(\mathbf{x}(n+1)) - 2F_a(\mathbf{x}(n)) + F_a(\mathbf{x}(n-1)))\frac{\partial F_a(n)}{\partial x_b(n)} \qquad (2.18)
\end{aligned}$$

It is easy to recover Eq.(2.15) by replacing those finite difference formula with the corresponding derivatives.

Boundary Points

$$\begin{aligned}
0 = \frac{\partial A_0}{\partial x_b(0)} &= \sum_{a=1}^{D} R_f(a)[x_a(1) - f_a(x(0))]\left[-\delta_{ab} - \frac{\Delta t}{2}\frac{\partial F_a(0)}{\partial x_b(0)}\right] \\
0 = \frac{\partial A_0}{\partial x_b(m)} &= \sum_{a=1}^{D} R_f(a)[x_a(m) - f_a(x(m-1))]\left[\delta_{ab} - \frac{\Delta t}{2}\frac{\partial F_a(m)}{\partial x_b(m)}\right]
\end{aligned}$$

Divide Δt on both sides:

$$\begin{aligned}
0 = &\sum_{a=1}^{D} R_f(a)\left[\frac{x_a(1) - x_a(0)}{\Delta t} - F_a(x(0)) + \frac{1}{2}(F_a(x(0)) - F_a(x(1)))\right] \\
&\times \left[-\delta_{ab} - \frac{\Delta t}{2}\frac{\partial F_a(0)}{\partial x_b(0)}\right] \\
0 = &\sum_{a=1}^{D} R_f(a)\left[\frac{x_a(m) - x_a(m-1)}{\Delta t} - F_a(x(m)) + \frac{1}{2}(F_a(x(m)) - F_a(x(m-1)))\right] \\
&\times \left[\delta_{ab} - \frac{\Delta t}{2}\frac{\partial F_a(m)}{\partial x_b(m)}\right]
\end{aligned}$$

Ignore the second-order terms, it is the boundary condition Eq.(2.16).

Therefore, in the limit that $\Delta t \to 0$, we have shown the equivalence of the

first-order optimality conditions between the continuous and discrete cases.

2.3 Annealing Method

We have proposed a strategy for dealing with this 'submersion' of the paths with smallest action. The idea is that when the model errors are forced, by large R_f, to be small, the nonlinearity of the vector field $\mathbf{f}(\mathbf{x})$ manifests itself at the smallest scales in the phase space of the paths $\mathbf{X}$ where we are searching. This causes complicated fine structure seen as multiple local minima [3] in the action, especially when the number of measurements L is too small.

As one reduces R_f, the loss of resolution of the flow of the nonlinear dynamics becomes large, and there is an averaging effect over the multiple local minima encountered. In fact at $R_f = 0$, there is no influence of the nonlinear dynamics of $\mathbf{x}(n+1) = \mathbf{f}(\mathbf{x}(n))$ at all, and the action we are working with then is

$$A_0(\mathbf{X}) = \sum_{n=0}^{m} \sum_{l=1}^{L} \frac{R_m(l,n)}{2} [x_l(n) - y_l(n)]^2. \tag{2.19}$$

The minimum of this consists of $x_l(t) = y_l(t)$ for the observed states with the unobserved states completely unspecified. The huge degeneracy of this minimum action is broken as R_f increases from zero, and we propose to adiabatically track the $R_f = 0$ minimum, which is also the global minimum for $R_f = 0$, by slowly raising the magnitude of R_f from zero.

We call this tracking of extremum path in R_f "annealing" for short as it treats the importance of the deterministic dynamics in a steady, slowly growing manner, as if small R_f corresponds to a high effective temperature in which the nonlinear

interaction among "particles" located at $\mathbf{x}(t)$ is initially in a harmonic well centered near the observations $\mathbf{y}(t)$. As we increase R_f it is as if we cool down a 'temperature' proportional to R_f^{-1} and impose structure on the trajectories $\mathbf{x}(t)$. Note our annealing method is totally different from the well-known simulated annealing method [56].

Our 'annealing schedule' is written as

$$R_f = R_{f0}\alpha^{\beta} \tag{2.20}$$

with $\alpha > 1$ and $\beta \geq 0$. We proceed in the following manner

- Start at a small value of $R_f = R_{f0}$. We take as an `initial path` for our optimization algorithm the solution at $R_f = 0$ just described with $x_l(t) = y_l(t)$ and the other elements of the path drawn from a uniform random distribution covering the dynamical range of their variation. In practice we have selected R_{f0} to be between 0.001 and 0.01.

  We have also performed this initial stage of the annealing calculation starting at $R_f = R_{f0}$ with an initial set of N_0 random choices for the components of $\mathbf{X}$. Since the influence of the dynamics is so small at R_{f0}, the paths quickly become those suggested in the previous paragraph. No difference in the subsequent calculations have been seen by us.

- Using a selected optimization procedure, we then utilize N_0 choices for initial paths $\mathbf{X}_0^q;\ q = 1, 2, ..., N_0$ with $x_l(t) = y_l(t)$ and randomly chosen unobserved state variables. Fixed parameters in the model or in an observation function $\mathbf{h}(\mathbf{x})$ are also randomly selected over a finite range. The optimization procedure takes these initial paths and, with $R_f = R_{f0}$ (that is $\beta = 0$), results in N_0 new

paths which we call $\mathbf{X}_1^q;\ q = 1, 2, ..., N_0$. We then evaluate the action $A_0(\mathbf{X}_1^q)$ on each of these N_0 paths. At this early stage we usually find the infinitely degenerate action values for the initial path $\mathbf{X}_0^q$ have begun to split.

- We proceed by raising the value of R_f from R_{f0} to $R_{f0}\alpha$, that is $\beta = 1$, and use the N_0 paths $\mathbf{X}_1^q$ as initial paths for this application of the optimization procedure. This results in a new set of N_0 paths $\mathbf{X}_2^q$.

- The paths $\mathbf{X}_2^q$ are now used as N_0 initial paths for $R_f = R_{f0}\alpha^2$, that is $\beta = 2$. This results in a set of N_0 paths resulting from our optimization procedure which we call $\mathbf{X}_3^q$.

- We continue this annealing schedule until by using the N_0 paths $\mathbf{X}_{J-1}^q$ to initialize the optimization procedure at $\beta = J-1$ we arrive at a set of N_0 paths $\mathbf{X}_J^q$ where we terminate the schedule.

When the annealing schedule is completed, we often encounter in our plots of $A_0(\mathbf{X}^q)$ versus $\log_\alpha[R_f]$ a region for β large enough where some $A_0(\mathbf{X}^q)$ becomes independent of R_f. The lowest action level that splits off in action level value from the action on other paths, will, when the number of measurements L is large enough, provide the dominant contribution to $\langle G(\mathbf{X})\rangle$.

Independence of $A_0(\mathbf{X}^q)$ from R_f indicates that the model output has matched the deterministic dynamics $\mathbf{x}(n+1) = \mathbf{f}(x(n))$ quite well. The remaining term in the action is then

$$A_0(\mathbf{X}) = \sum_{n=0}^{m} \sum_{l=1}^{L} \frac{R_m(l,n)}{2}[x_l(n) - y_l(n)]^2. \tag{2.21}$$

As the values $[y_l(n) - x_l(n)]$ are distributed as $\mathcal{N}(0, \sigma^2)$ by our choice, the measurement error term $\sum_{n=0}^{m}\sum_{l=1}^{L}[(x_l(n) - y_l(n))/\sigma]^2/2$ has a χ^2-distribution with $L(m+1)$ degrees

of freedom [32]. The mean and RMS variation of this distribution over different choices of noise waveforms are

$$\mu = \frac{1}{2}(m+1)L \tag{2.22}$$

$$\sigma = \sqrt{\frac{(m+1)L}{2}} \tag{2.23}$$

This level is shown in our action value versus R_f plots by a heavy horizontal line. When the action levels as a function of R_f reach this expected χ^2 lower limit, we have a path $\mathbf{X}^0$ on which the model behavior is consistent with the data within the noise level of the data.

In the examples we will discuss below, we have selected $N_0 = 10 \sim 100$ and taken $20 \sim 50$ annealing steps as a stopping point. We have also used $\alpha = 2$ on the whole, but also selected $\alpha = 1.5$ when we chose to take smaller annealing steps in R_f within our schedule.

2.4 An Example Illustrating Annealing; Lorenz96 Model with $D = 5$

We begin by examining the dynamical equations introduced by Lorenz [73]:

$$\frac{dx_a(t)}{dt} = x_{a-1}(t)(x_{a+1}(t) - x_{a-2}(t)) - x_a(t) + \nu \tag{2.24}$$

and $a = 1, 2, ..., D$; $x_{-1}(t) = x_{D-1}(t)$; $x_0(t) = x_D(t)$; $x_{D+1}(t) = x_1(t)$. ν is a fixed parameter which we take to be in the range 8.0 to 8.2 where the solutions to the dynamical equations are chaotic [58]. The equations for the states $x_a(t);\ a = 1, 2, ..., D$

are meant to describe 'stations' on a periodic spatial lattice.

We perform a twin experiment wherein we generate D time series using a standard adaptive fourth order Runge-Kutta algorithm with a time step $\Delta t = 0.025$ with no model error. To these we add Gaussian noise with mean zero and variance $\sigma^2 = 0.25$ to each time series $x_a(t)$. These noisy versions of our model time series constitute our 'data. Our choice of σ^2 means a signal to noise ratio about 60 dB. We selected $\nu = 8.17$ in these calculations.

The measurement window is from $t_0 = 0$ to $t_f = 4.0$, so $m = 160$. L 'measurements' are made at each time step; these are the $\mathbf{y}(t_n)$. The measurement error matrix $\mathbf{R}_m$ is taken to have diagonal elements at each measurement time t_n and is zero at other times. Its magnitude is taken as $R_m = 1/\sigma^2 = 4$. The model error matrix is also taken as diagonal, with elements along the diagonal $R_f = R_{f0}2^\beta$, and we take $\beta = 0, 1, 2, \ldots,$. R_{f0} was chosen 0.01 for these calculations.

We begin with one measurement $y_1(n)$ among the five possible states, i.e. $L = 1$, and in Fig. 2.1 we display the $\log_{10}$ of the action $A_0(\mathbf{X})$ evaluated at each of the $N_0 = 100$ saddle paths for $R_f = R_{f0}2^\beta$ with $R_{f0} = 0.01$ We begin with $\beta = 0$ and increase it to $\beta = 22$.

The BFGS quasi-Newton method [80] was used as our search algorithm in this example. The optimization stopping conditions for scaled gradient norm, scaled stepsize norm and function change are all 1×10^{-8} and they are fixed throughout the annealing procedure. Low tolerance (larger than 1×10^{-6}) may cause incorrect results or no convergence. We provided the gradient of $A_0(\mathbf{X})$ in an analytical form to the algorithm. We initialized the search at $R_f = R_{f0}$ with N_0 initial paths $\mathbf{X} = \{\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(m)\}$ as described above. At $R_f = R_{f0}$ we selected the unobserved states at each time step

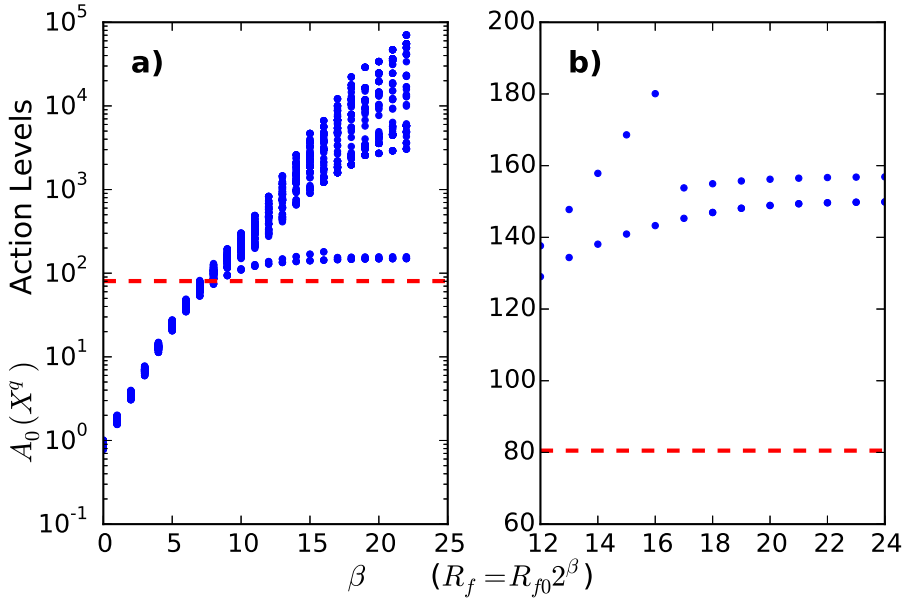

Figure 2.1: **(a)**: Action levels as a function of $\log_2[R_f/R_{f0}] = \beta$ for the Lorenz96 model, $D = 5$, $L = 1$, $R_{f0} = 0.01$. As R_f increases, the model error is decreased. The horizontal line shows the expected value of the measurement error terms in the action. The measurement error term is distributed as χ^2 with this expected value. **(b)**: Zooming in on the Left Panel for large β showing close action levels. The resulting saddle paths all have action levels above the χ^2 expected value for the measurement error action alone. This is an indication that $L = 1$ is not sufficient for identifying a good path for the Lorenz96 $D = 5$ model.

from a uniform distribution in the interval [-10,10]. This is approximately the dynamic range of state variables in the Lorenz96 model. At fixed R_m each search procedure as we slowly increase R_f yields N_0 saddle paths $\mathbf{X}^q$ and associated action levels $A_0(\mathbf{X}^q)$. As R_f increases many initial paths may lead to the same action level.

As one can see in Fig. 2.1 the degenerate action levels at $R_f = 0$ are split at $\beta = 0$ and then rise until, around $\beta = 12; R_f \approx 100$, two levels split off from the rest and become rather independent of R_f. There are still two quite close levels. Also shown is the expected value of the χ^2-distributed measurement error at 80.5. The distance of the action levels of the paths giving $A_0(\mathbf{X})$ near values about 150 tell us

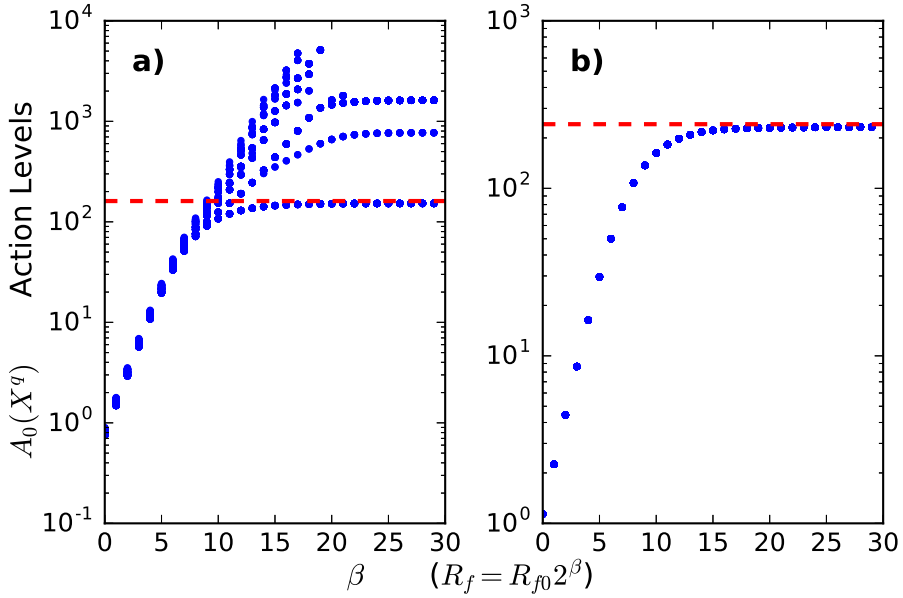

Figure 2.2: **(a)**: Action Levels as a function of R_f for the Lorenz96 model, D = 5, L = 2, $R_{f0} = 0.01$. We used $y_1(t)$ and $y_3(t)$ as data in the action. **(b)**: Action Levels as a function of R_f for the Lorenz96 model, D = 5, L = 3, R_{f0} = 0.01. We used $y_1(t)$, $y_3(t)$, and $y_5(t)$ as data in the action.

that these paths are unlikely to give consistency of the model with the data. Using either of these two paths to give us the full model states at the end of the estimation window $t_f = 4$ to predict beyond t_f gives quite inaccurate predictions.

Next we present $L = 2$ measurements, $y_1(n)$ and $y_3(n)$, to the model and again evaluate saddle paths as we vary β. Each path has $(m + 1)D = 805$ components, so the annealing problem is a search for saddle paths of the action Eq. (A.1) in an 805-dimensional space. We take the distribution of the three unobserved states at t_0, the beginning of the observation window, to be uniform over the dynamical range of $\mathbf{x}(t_0)$.

At $\beta = 0$ the degenerate action levels from $R_f = 0$ are split slightly. We follow these to larger values of R_f. (Fig. 2.2) At low R_f the resolution in path space is very coarse, and our search is successful for finding low lying action levels. Fig. 2.2

shows quite clearly that there are many paths with similar action level until we reach $\beta \approx 12$, and after that only one remains independent of R_f as the other action levels rise. The expected value of the χ^2-distribution of measurement errors in the action is $N_{\text{data}} R_m \sigma^2/2$. This is 161 here, and it is shown in the figure as a heavy horizontal line. The action level for $\mathbf{X}^0$ is very near this χ^2 consistency condition suggesting that the path $\mathbf{X}^0$ expresses consistency of the model and the data.

When we increase the number of measured time series to $L = 3$, the results in Fig. 2.2 show that one path alone emerges from the degeneracy at $R_f = 0$ and after $\beta \approx 12$ is again nearly independent of R_f and close to the expected limit from the χ^2 distribution.

To get some insight into how the annealing procedure proceeds in the sequence of estimates for the observed and unobserved states of the model to which $L = 2$ 'data' time series are presented, we show in Fig. 2.3 the estimated and the 'data' time courses for both an observed state variable $x_1(t)$ and an unobserved state variable $x_2(t)$ for selected values of $\beta = 0, 12$ and 21.

In Fig. 2.3 representative time series which are part of the path for different values of β in the lowest action level are plotted to illustrate the annealing process in detail. For very small β, say 0, $R_f = 0.01$, the top two panels of Fig. 2.3 show the known and estimated components $x_1(t)$, observed, and $x_2(t)$, unobserved, from one of the saddle paths. Since $R_m \gg R_f$, and the measurement error dominates the overall size of the action, paths are forced to follow the noisy measurements almost exactly so as to minimize the measurement error, i.e. $x_l(t) \approx y_l(t)$. The effect of the model error term is quite small with $R_f = 0.01$, the unmeasured states are usually undetermined. Its form depends on the initial random guess path. In this example, the initial path

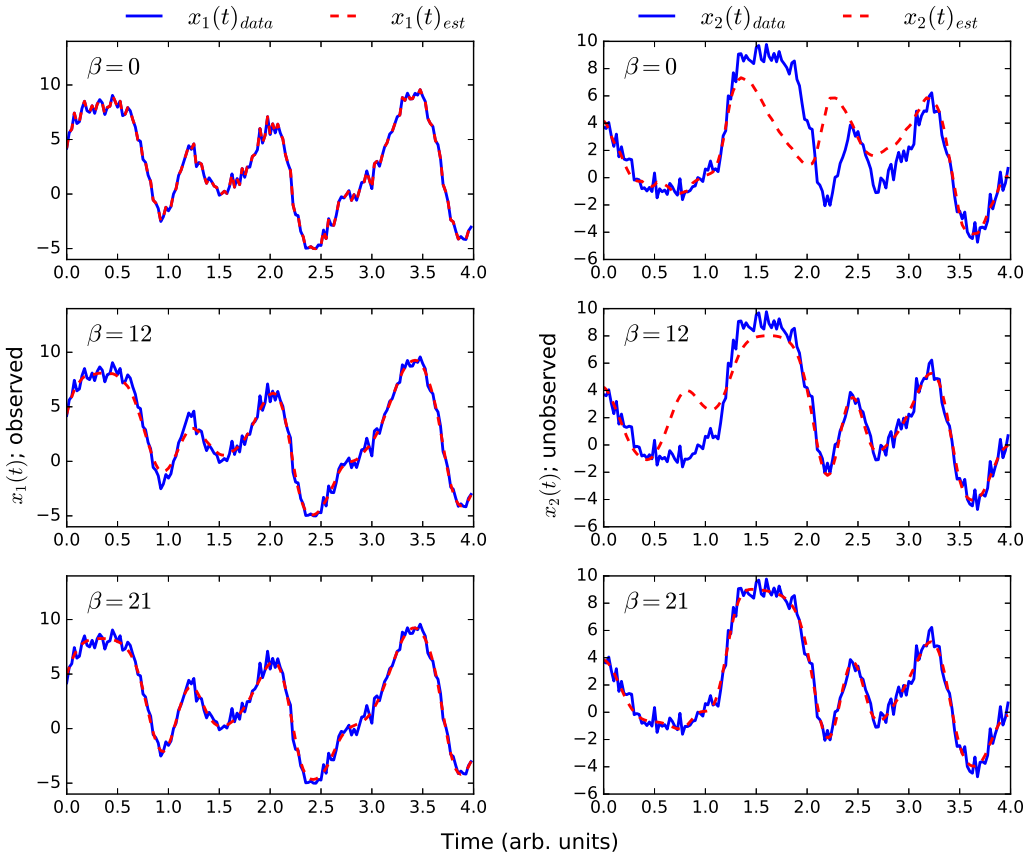

Figure 2.3: Estimation results for one observed state variable $x_1(t)$ and one unobserved (not assimilated) state variable $x_2(t)$ during the annealing procedure. $\beta = 0$, The estimation of $x_1(t)$ overfits its measurement. As β increases to 12, the estimated observed state becomes smoother. The observed and unobserved state variables arrive at their true states when β is large enough. Here $\beta = 21$.

happens to be chosen near the true path, and the unobserved state x_2 is close to the known data at the beginning and end of the window.

As β is increased to 12, $R_f \approx 40$, we have moved from a regime where R_f is quite small to a regime where R_f has become sizable. The role of the model error is no longer insignificant. The trajectory of the observed state $x_1(t)$ is smoother, passing through the middle of the noise fluctuations, but not tracking the noise as was done at $\beta = 0$. The greater R_f, the more information is input from the model. This information from the model helps the observed state filter out the noise to some extent. When β increases up to 21, $R_f \gg R_m$, it enforces the model more and more exactly, $x_a(n+1) \approx f_a(\mathbf{x}(n))$. Both observed states and unobserved states converge to the true path for the lowest action level. The size of action $A_0(\mathbf{X})$ matches the observation error residual $N_{data} R_m \sigma^2/2$.

It is important to note that if we begin our search for the saddle paths $\mathbf{X}^q$ at large values of R_f, we are almost sure to miss the actual path $\mathbf{X}^0$ which gives the lowest action level, since the Hessian matrix of $A_0(\mathbf{X})$ is ill-conditioned when R_f is large and the lowest action level occupies a tiny corner of the large (here 805 dimensional) path space. See Fig. 4.6 in [81].

2.5 Endpoint Conditions on the Minimum Paths

In the integral for conditional expected values Eq. (A.2), when expressed in discrete time, each of the integrals for states $\int d^D\mathbf{x}(n)$ along the path $\mathbf{X} = \{\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(m)\}$ is unconstrained as we do not specify or hold fixed any of the state values in $\mathbf{X}$. Yet, when we proceed to the continuous time limit, as we have argued from the derivation of the Euler-Lagrange equations the canonical momentum

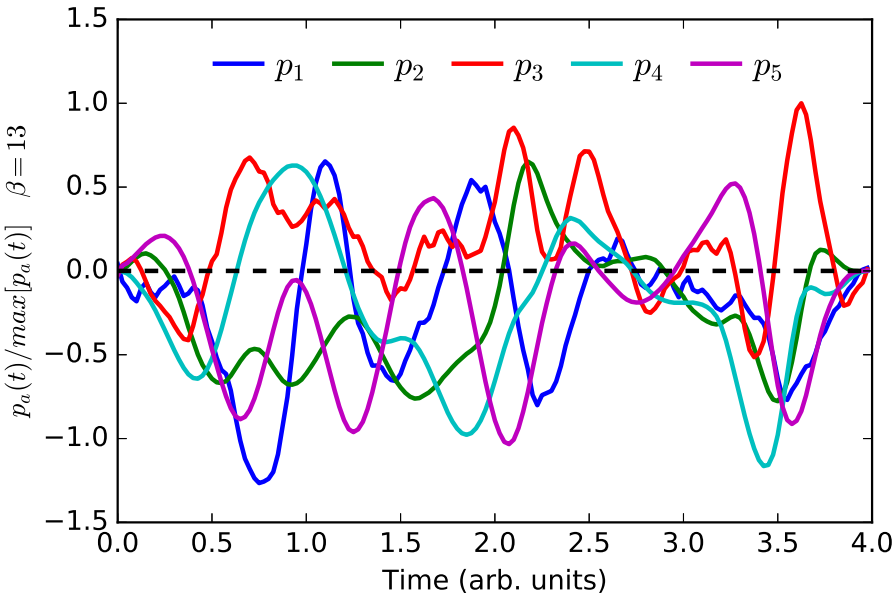

Figure 2.4: Five canonical momenta $p_a(t) = R_f[\dot{x}_a(t) - F_a(\mathbf{x}(t))]$ of one local minimum path for the Lorenz96 model, $D = 5$, $L = 2$. $\beta = 13$. $R_f(a) = R_f$ for all a = 1, ...,5. At the endpoints $t = 0$ and $t = 4$, all the $p_a(t)$ go to zero, satisfying the boundary conditions associated with the vanishing of the first variation of the action. Each $p_a(t)$ is scaled by its maximum magnitude over $0 \leq t \leq 4$

$\partial\mathcal{L}(\mathbf{x}, \dot{\mathbf{x}}, t)/\partial\dot{\mathbf{x}}(t)$ must vanish at the temporal end points of the integration.

We investigated this by examining the saddle paths for $\beta = 13$ using the Lorenz96 model with $D = 5$ and $L = 2$. In Fig. 2.4 we show the canonical momentum $p_a(t) = R_f(a)[\dot{\mathbf{x}}_a(t) - F_a(\mathbf{x}(t))]$ for $a = 1, ..., 5$ scaled by the maximum value within the observation window.

We randomly picked one of the saddle paths at $\beta = 13$, and we evaluated the $\dot{\mathbf{x}}(t)$ in the canonical momentum using the second-order central finite difference scheme

$$\dot{\mathbf{x}}(n) = \frac{\mathbf{x}(n+1) - \mathbf{x}(n-1)}{2\Delta t} + O(\Delta t^2),$$

for interior points in the observation window. We maintained second-order accuracy

for the evaluation of $\dot{\mathbf{x}}(t)$ at the boundary by using

$$\dot{\mathbf{x}}(0) = \frac{-3\mathbf{x}(0) + 4\mathbf{x}(1) - \mathbf{x}(2)}{2\Delta t} + O(\Delta t^2)$$
$$\dot{\mathbf{x}}(m) = \frac{3\mathbf{x}(m) - 4\mathbf{x}(m-1) + \mathbf{x}(m-2)}{2\Delta t} + O(\Delta t^2).$$

The scaled trajectories in $t_0 \leq t \leq t_f$ of the five components of canonical momentum $p_a(t)$ are plotted in Fig. 2.4, which shows the required boundary condition is satisfied rather well. As this boundary condition is necessary for an extremum of the action when the end points are not constrained, this result may be only a consistency check on the accuracy of our calculation. Of course, it is good that the result is positive.

Chapter 2, in full, is a reprint of the material as it appears in Jingxin Ye, Daniel Rey, Nirag Kadakia, Michael Eldridge, Uriel I. Morone, Paul J. Rozdeba, Henry D. I. Abarbanel, John C. Quinn Systematic variational method for statistical nonlinear state and parameter estimation, *Physical Review E*, 92(5), 052901 (2015). The dissertation author was the primary investigator and author of this paper.

Chapter 3

Further Examples

Continuing the topic of the illustrative example, we check the behavior of the annealing method on larger dimensional Lorenz96 model with $D = 20$. To further explore the annealing method and examine an example from the atmospheric sciences literature [73] and an example involving a standard Hodgkin-Huxley neuron model. The first has both fast and slow variables, namely fast and slow time scales in the governing differential equations, representing small scale (fast) and large scale (slow) atmospheric variations. The challenge to a variational method is to reliably capture both time scales in identifying an accurate saddle path. The second moves away from the simple vector fields in the Lorenz96 model, and fluid dynamical models in a general sense, which are at most quadratic in their nonlinearities. The kinetics of gating variables associated with voltage dependent conductances in neurons involves parameters and states entering the vector fields through exponentials reflecting the underlying statistical properties of the cellular processes.

3.1 Lorenz96 model $D = 20$

Larger dimensional Lorenz96 [73] model used frequently in geophysical data assimilation discussions as a testbed for our proposed annealing method. We studied $D = 20$ and added f as an additional degree of freedom satisfying $\dot{f} = 0$. We performed a twin experiment in which we solved these equations with an arbitrary choice of initial conditions $x_a(0)$ using a fourth order Runga-Kutta solver with $\Delta t = 0.025$ over 160 steps in time. $t_0 = 0$ and $t_m = T = 4$. We then added iid Gaussian noise with zero mean and variance $\sigma^2 = 0.25$ to each time series. $L = 1, 2, \ldots$ of the data series were represented in the action at each measurement time t_n during our annealing procedure.

In the action we selected $R_m = 4$, the inverse variance of the noise added to the data in our twin experiment, so the minimum action level we expect is $161\, L/2$. The paths are $(m + 1)(D + 1) = 3381$-dimensional. Our search for minimum paths used a BFGS routine [80] to which we provided an analytical form of the gradient of $A_0(X)$. The search was initialized with 100 initial paths from a uniform distribution of values in the interval [-10,10].

In Fig.3.1 we display the computed action levels for $L = 5, 7, 8$ and 9. For $L = 5$ there are many close action levels associated with the extremum paths of the action Eq.(A.1). The expected lowest action level Eq.(2.22), minimum $A_0(X^0)$ and next $A_0(X^1)$ action levels are for $L = 5$: 402.5, 373.5, 403.8; for $L = 7$: 573.5, 545.2, 749.8; $L = 8$: 644.0, 613.2, 1161.6, and $L = 9$: 724.5, 685.4, 2256.1. Our estimate for the forcing parameter, set to 8.17, was 8.22 at large β.

The real test of an estimation procedure is not accuracy in the estimation, but accuracy in prediction beyond the observation window. As this is a twin experiment,

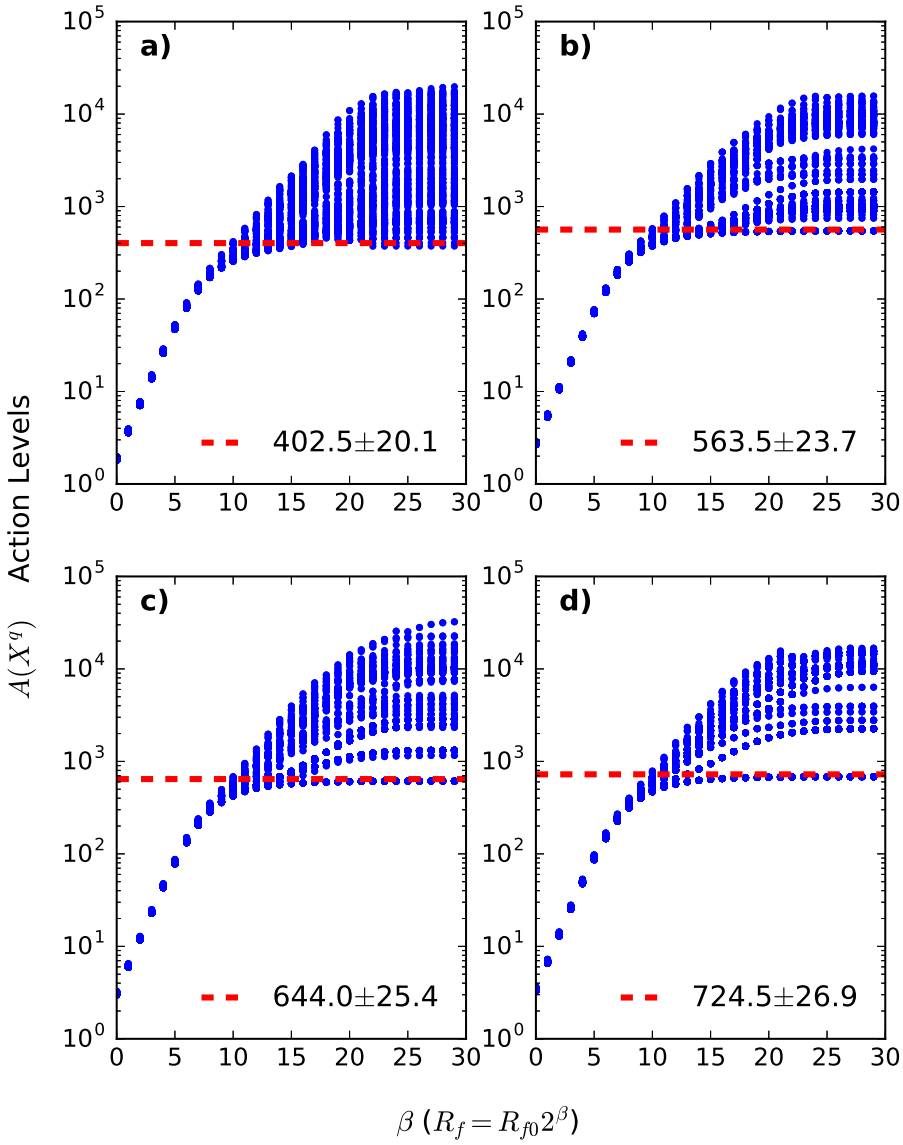

Figure 3.1: Action levels as a function of R_f for Lorenz96 model, $D = 20$, $R_{f0} = 0.01$. **a)** $L = 5$ we used $y_1(t),\ y_3(t),\ y_5(t),\ y_7(t),\ y_9(t)$ in the action; **b)** at $L = 7$, $y_{11}(t),\ y_{13}(t)$ are added; **c)** at $L = 8$, $y_{15}(t)$ is added; **d)** at $L = 9$ $y_{17}(t)$ is added. The expected values of the lowest action level are denoted by black dashed lines.

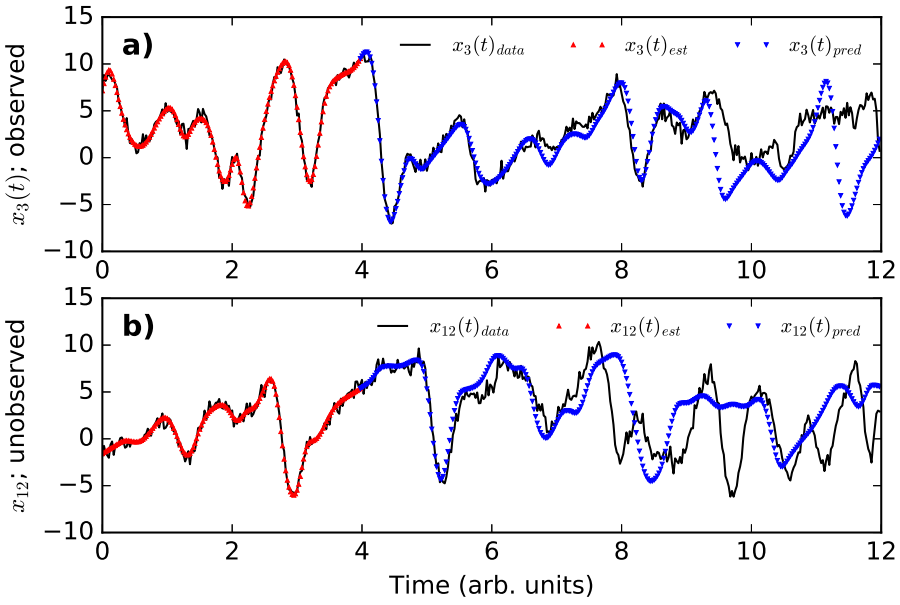

Figure 3.2: Data, estimated, and predicted time series for the Lorenz96 model [73] with $D = 20$, $L = 8$. **a)** $x_3(t)$ was an observed state variable and **b)** $x_{12}(t)$ was unobserved. The data (black) the estimated state variable (red) and the predicted state variable (blue) are shown for each of them.

we show in Fig.3.2 the data, the estimated state variable and the predicted state variable for an an observed variable $x_3(t)$ and for an unobserved variable $x_{12}(t)$ for $L = 8$. In a real experiment, we could not compare our estimates for the parameters or the unobserved state variables. Although the estimation procedure for the path X^0 with the minimum action value is rather good, estimating 12 unobserved states and one parameter, there are, of course, errors in our knowledge of the full state $\mathbf{x}(T = 4)$. The predictions lose their accuracy in time because of the chaotic nature of the trajectories at $f = 8.17$.

To see how well our procedure works for several unknown parameters, we introduced 10 different forcing parameters f_a into the Lorenz96 model at $D = 10$: $\dot{x}_a(t) = x_{a-1}(t)(x_{a+1}(t) - x_{a-2}(t)) - x_a(t) + f_a$. There the lowest action level stands out from the rest at $L = 4$. In Table 3.2 we show our estimates for the ten forcing

parameters for $L = 4$, 5, and 6, as well as the actual value used in the calculations of the data. In these estimates and for the single forcing parameter reported above for $D = 20$, there is a known source of bias [59]. As one can see in the examples it is small here.

Table 3.1: Known and Estimated forcing parameters for the Lorenz96 Model at $D = 10$, L =4, 5, and 6.

Known f_a	$L = 4$	$L = 5$	$L = 6$
5.7	5.742	5.737	5.768
7.1	7.096	7.080	7.094
9.6	9.696	9.686	9.654
6.2	6.156	6.174	6.131
7.5	7.605	7.592	7.604
8.4	8.353	8.330	8.349
5.3	5.310	5.278	5.214
9.7	9.679	9.703	9.643
8.5	8.632	8.629	8.626
6.3	6.334	6.336	6.308

3.2 Lorenz96 model with both fast and slow variables

Dynamical systems varying with several distinct time scales are commonly seen in earth system models. In the same paper where Lorenz introduced the Lorenz96 model we reported on earlier, he also introduced a modified model with both fast and slow variables to study the local instability responsible for convective activity. [73]

This model is given by

$$\frac{dx_k(t)}{dt} = - x_{k-1}(t)(x_{k-2}(t) - x_{k+1}(t)) - x_k(t) + \nu - \frac{hc}{b} \sum_{j=J(k-1)+1}^{kJ} z_j(t);$$
$$\frac{dz_j(t)}{dt} = - cbz_{j+1}(t)(z_{j+2}(t) - z_{j-1}(t)) - cz_j(t) + \frac{hc}{b} x_{\text{floor}[(j-1)/J]+1}(t). \qquad (3.1)$$

where $k = 1, 2, ..., D_{\text{slow}} = K$; $j = 1, 2, ..., D_{\text{fast}} = JK$ and floor$[x]$ is the floor function. $x_0(t) = x_K(t), x_{-1}(t) = x_{K-1}(t)$, etc. and likewise for the $\mathbf{z}(t)$ variables. The first equation describes the linked dynamics of a set of K slow, large-amplitude variables $x_k(t)$, each of which is associated with J fast, small-amplitude variables $z_j(t)$ whose dynamics are described by the second equation. $z_j(t)$ represents a convective-scale quantity coupling with $x_k(t)$ that favors the convective activity. They can be visualized as sectors on a lattice circle (Figure 1 in Ref.[97]). Each $x_k(t)$ sector contains many $z_j(t)$ sectors. We chose $D_{\text{slow}} = K = 5$ and $D_{\text{fast}} = J = 5$; this means that five $z_j(t)$ sectors are contained in one $x_k(t)$ sector. The detailed time scales for x_k and z_j are determined by the parameters h, c, b. We follow Lorenz and select $h = 1, c = 10, b = 10$, so the fast variables $z_j(t)$ vary approximately 10 times more rapidly than the slow variables $x_k(t)$, while their amplitudes are about one tenth of those of the $x_k(t)$. The forcing parameter ν is taken to be 18 following Wilks' work to make both $x_k(t)$ and $z_j(t)$ chaotic [97].

To generate our data, we integrate the slow equations with a time step $\Delta t = 0.001$ for the temporal window $t = [0, 4] = 4000\Delta t$ using a standard adaptive fourth order Runge-Kutta scheme. From these time series 'data' are obtained by adding white Gaussian measurement noise to the computed time series: $\mathcal{N}(0, 0.5)$ for $x_k(t)$ and $\mathcal{N}(0, 0.05)$ for $z_j(t)$.

Fixing the parameters h, c, b and f at the values used to generate the data, we seek to estimate the unobserved state variables as $L < K(J+1)$ time series are presented to the model. As above we perform our calculation using the annealing method. Following our definition, R_m equals the inverse of the variance of measurement noise, i.e. $R_m = 4$ for $x_k(t)$ and $R_m = 400$ for $z_j(t)$ when the corresponding variables are measured, and 0 otherwise.

Instead of choosing the same R_f value for every variable as we did in our illustrative example Lorenz96 model, the values of R_f need to be determined by how rapidly the variables changes. The R_f works as a penalty parameter during the annealing process. A more rapidly varying variable requires a larger R_f to regulate its fluctuations, and vice versa. By manipulating the values of R_f, we can ensure the model error terms are of the similar scale. Therefore, during the variational process, all the elements of each path from different variables will be well adjusted simultaneously to the path.

As we stated above, the amplitude of the $dz_j(t)/dt$ is approximately ten times smaller than that of the $dx_k(t)/dt$, so the variance of model error for $x_k(t)$ is about 100 times larger. The R_f's, as the inverse of model error variance, are chosen to be

$$R_{f0} = \begin{cases} 0.01 & \text{for } x_k(t) \\ 1 & \text{for } z_j(t). \end{cases}$$

According to our results above in the Lorenz96 $D = 5$ model the action level plots suggest that when measurements of $y_1(t), y_3(t)$ are presented, all the unmeasured states can be accurately estimated and lead to excellent prediction. We proceed then by presenting the noisy $x_1(t)$ and $x_3(t)$, namely, $y_1(t), y_3(t)$ to the fast/slow model, and

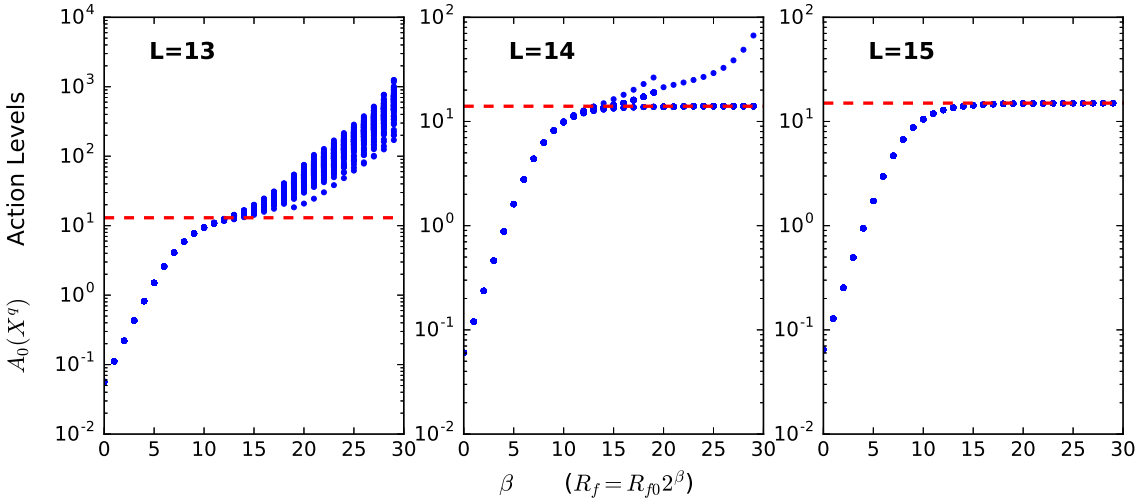

Figure 3.3: Action Levels as a function of R_f for the Lorenz96 Fast/Slow model, $K = 5$, $J = 5$: when $L = 13$ we used noisy measurements of $x_1(t), x_3(t)$ and $z_{2j-1}(t)$'s with $j = 1, 2, \ldots, 11$ as measured variables. At $L = 14$, $z_{23}(t)$ is added; at $L = 15$, $z_{25}(t)$ is also added. Note that when $L = 14$ and $L = 15$, the lowest action level splits off from the other allowed action levels corresponding to other paths meeting the saddle path condition.

then gradually increase the number of noisy measurements of $z_j(t)$ until we can find the consistent lowest action level we expect. We call the total number of measurements L and recognize it is comprised of 2 observations of the $x_k(t)$ and $L - 2$ observations of the $z_j(t)$.

The action level plots are shown in Fig. 3.3. We start the annealing calculation with $N_0 = 100$ initial random paths, so that whether there are paths located at the expected lowest action level is a stochastic event. When $L \leq 13$, it is rare to find a path finally reaching the expected lowest action level. The same calculations for $L = 12, 13$ are repeated 10 times, and no action level near the overall expected values is observed. Fig. 3.3 shows one example of $L = 13$ such that no action level levels out with increasing R_f. This suggests none of the paths is consistent with the model and that model errors are amplified as R_f increases. It is a sign that indicates the

information provided by the selected measurements is not enough. Another sign denoting lack of information in the measurements is that there are many close action levels near the expected lowest action level as shown in Figure 1 of Ref.[99].

It is worth pointing out that not having enough measurement information is not the only possible cause of having no levels becoming independent of R_f for large R_f. Anything introducing significant non-zero model errors can cause this as well. For example, numerical discretization error in action may appear to be important in fast/slow dynamical systems when one doesn't use a small enough observation time step to meet the requirement of the resolution of their fastest dynamics. In the Lorenz96 fast/slow model $K = 5$, $J = 5$ if we use $\Delta t > 0.0025$, one can observe this even when $L = 14$.

We added another measurement to the action, and the complicated action levels at $L = 13$ are immediately reduced to three distinct levels when $12 \leq \beta \leq 19$ and then collapses into two levels, including one associated with the expected value, after $\beta \geq 20$. Adding in $z_{25}(t)$, we found all of the 100 random initial paths converge to the solution of the consistent action minimum.

The estimated state variables of the Lorenz96 fast/slow model at $L = 14$ are displayed in Fig. 3.4. Predictions are obtained by advancing the model forward in time using the estimated state variables at $t = 4.0$ as an initial condition. (Fig. **??**) Both estimation and prediction show excellent agreement with the data generated in this twin experiment. In time the prediction loses its accuracy because of the chaotic nature of solutions to this dynamical system. This is especially apparent for the fast variables $z_j(t)$.

The annealing method is successful in locating a dominant lowest action path

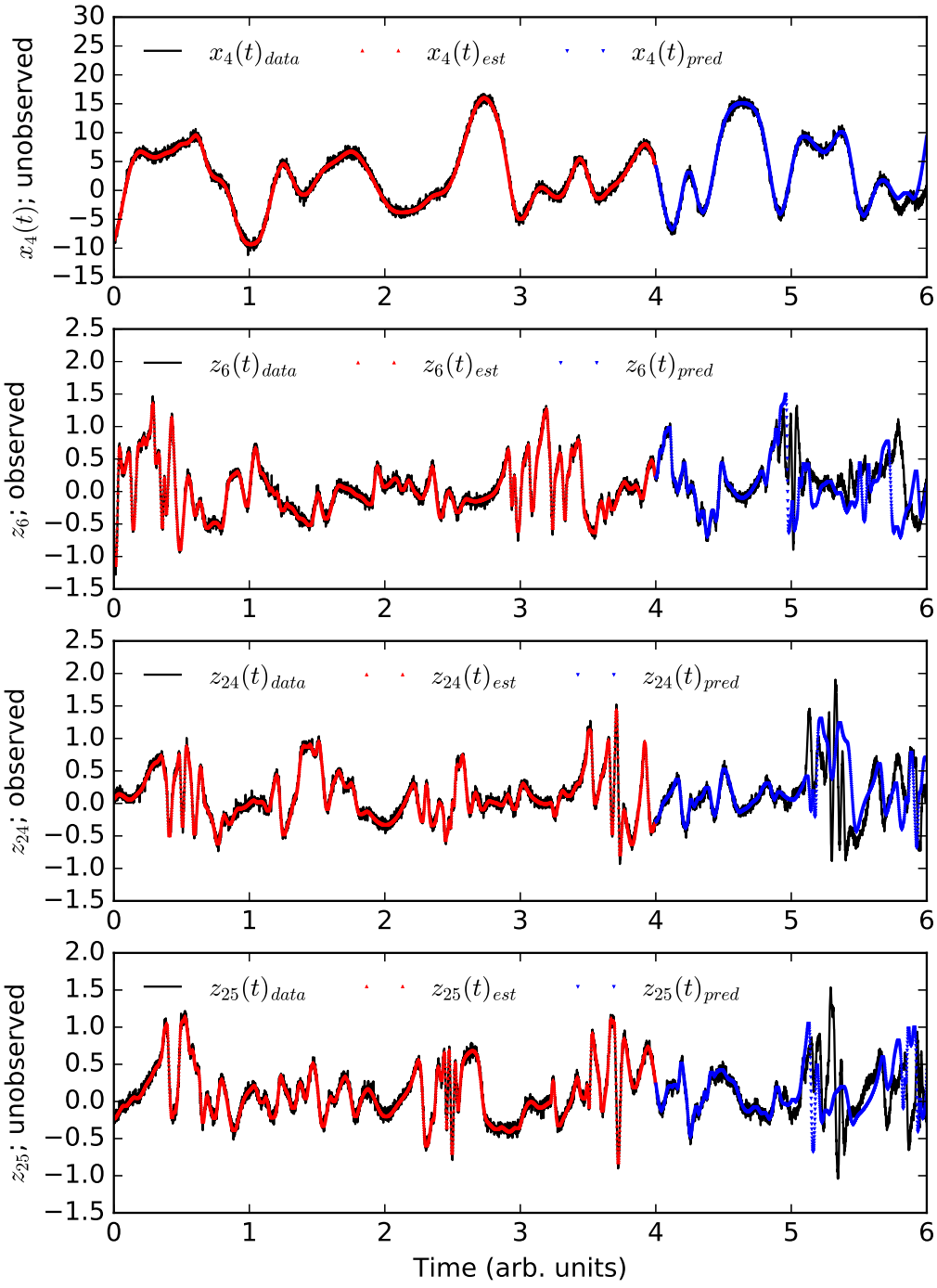

Figure 3.4: Data, estimated and predicted time series for the Lorenz96 Fast/Slow model with $K = 5$, $J = 5$ and $L = 14$. f = 18. We used noisy versions of the $x_1(t)$, $x_3(t)$ and of the $z_{2j-1}(t), j = 1, 2, \ldots, 12$ as measured variables. $x_4(t)$ is an `unobserved` slow variable. Both of the fast variables $z_6(t)$ and $z_{24}(t)$ are `observed`. $z_{23}(t)$ is an `unobserved` fast variable.

even in the presence of distinct times scales in the dynamics. One must sample the observations fast enough to capture the higher frequency variations of the fast variables. In the annealing search for lowest action levels, one must also select the ratios of the maximal values of R_f to reflect the different dynamical time scales.

3.3 Hodgkin-Huxley neuron model: NaKL model

We selected a fairly standard Hodgkin-Huxley (HH) neuron model [50, 89] consisting of four state variables: the voltage $V(t)$ across the cell membrane as well as three voltage dependent gating variables for Na^+ and K^+ channels $m(t), h(t)$ and $n(t)$. The equation governing changes in voltage across the cell membrane is current conservation with conductances for Na^+ and K^+ ions through the membrane that depend on the voltage $V(t)$. This reflects the change in permeability to these ions of proteins that transect the membrane and change their conformation as a function of the voltage across the membrane. The specific forms of the voltage dependent conductivities in this HH model are taken from textbook descriptions based on the 1940s and 1950s work of Hodgkin, Huxley, Katz, and many others. The reversal potentials are determined by the competition of diffusion associated with ion concentration differences within and without the cell and transport of charged ions by the electric field associated with the difference in voltage across the membrane. The Nernst equation which determines these reversal potentials is directly from statistical physics.

The cell responds to external currents as a driving force by its cross membrane voltage $V(t)$ rising if the current causes depolarization of the cell, or the voltage decreases when the cell becomes more polarized. (Fig. 3.5) The rise in voltage triggers

an instability in the phase space of the HH model associated with a sudden influx of Na^+ ions which is then counterbalanced by a flow of K^+ ions out of the cells as the voltage rises to order + 50 mV. All this takes place on the order of 5 ms and is seen as a 'spike' in the voltage time series.

The model is governed by the following four first-order differential equations:

$$\begin{aligned}
C\frac{dV(t)}{dt} =& I_{inj}(t) + g_{Na}m(t)^3h(t)(E_{Na} - V(t)) \\
&+ g_K n(t)^4(E_K - V(t)) + g_L(E_L - V(t)) \\
\frac{da(t)}{dt} =& \frac{a_0(V(t)) - a(t)}{\tau_a(V(t))} \qquad a(t) = \{m(t), h(t), n(t)\} \\
a_0(V) =& \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{V - V_a}{\Delta V_a}\right) \\
\tau_a(V) =& \tau_{a0} + \tau_{a1}\left(1 - \tanh^2\left(\frac{V - V_a}{\Delta V_a}\right)\right)
\end{aligned}$$

In these equations the g_{ion}'s are maximum conductances for the ion channels, the E_{ion} are reversal potentials for those ion channels, $I_{\text{inj}}(t)$ is the external stimulating current injected into the neuron. This current is selected by the experimenter and has no independent dynamics.

The gating variables $a(t)$ are taken to satisfy first order kinetic equations and range between zero and unity. The overall strength of an ion channel is set by the maximal conductances, and this represents the number of individual ion channels. These are phenomenological choices.

The quantities $a_0(V)$ and $\tau_a(V)$ are the voltage dependent activation function and the voltage dependent time constant of the gating variable $a(t)$. The forcing to the cell $I_{\text{inj}}(t)$ is known to us. In our parametrization of the cell dynamics there are 19 fixed parameters and three unobserved state variables $a(t) = \{m(t), h(t), n(t)\}$ to

be determined. All $a(t)$ lie between zero and one.

Only the voltage across the cell membrane is measurable in real neurobiological experiments, however, successful data assimilation, in effect, 'measures' the gating variable time series as well as the unknown parameters. We present only noisy time series of $V(t)$ to the model; these are our $\mathbf{y}(t)$ in the notation we have used for the general discussion above.

The parameters used to generate data are listed in Table 3.2. The waveform of the injected current is chosen to be a combination of step functions and segments of a chaotic time series taken from one of the variables of the Lorenz63 model [72]. This current is displayed in the bottom panel of Fig. 3.5. A standard adaptive fourth order Runge-Kutta solver is used to produce the data using time steps of $\Delta t = 0.025$ms, and white Gaussian noise with an RMS level of 1mV is added to the $V(t)$ time series to represent the noise accompanying the measurements in laboratory biological experiments. This voltage time course is in the top panel of Fig. 3.5.

The previous two examples, the Lorenz96 $D = 5$ model and the Lorenz96 fast/slow model contain only quadratic nonlinear terms in their differential equations. The difficulties of state and parameter estimation result from their chaotic trajectories. The NaKL model in the selected parameter region is not chaotic, and the challenge of data assimilation comes from the richer nonlinearity in the dynamics of the gating variables and the sensitivity of the model behavior to changes in parameter values.

In the numerical optimization used to find the saddle paths for any model one must specify search bounds for each parameter and each state variable. The goal is to find appropriate choices for these values that constrain the model states and parameters to biologically acceptable regions. The bounds for the voltage $V(t)$ are

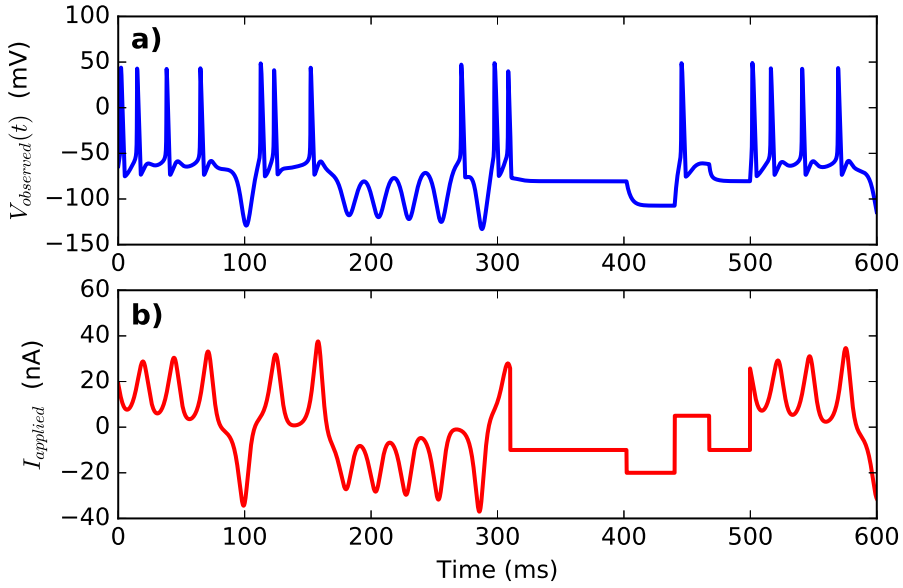

Figure 3.5: **(a)** Voltage response of the standard Hodgkin-Huxley neuron model with Na^+, K^+ and leak channels, our NaKL model, in response to the applied (injected) stimulus current shown in the **(b)**

taken as -150mV and $+70$mV based upon our solutions to the equations. The gating variables are bounded between 0 and 1, since they represent the probability whether ion channels are open or closed.

The optimization is implemented with IPOPT using an interior-point method [95]. We found the interior-point method both more stable and substantially faster than the L-BFGS-B method [102].

In this twin experiment only a noisy voltage $V(t)$ is 'measured' and presented to the model; so $L = 1$. As the dynamical range of voltage is a hundred times larger than that of the gating variables, we first calculated the action levels with $R_m^{(V)} = 1, R_{f0}^{(V)} = 10^{-3}, R_{f0}^{(m)} = 10, R_{f0}^{(h)} = 10, R_{f0}^{(n)} = 10$ and $\alpha = 3/2$. The largest β was taken as 50. Also we decreased α from 2 to 3/2 so that the pace of increasing resolution in model state space is slower than in our earlier examples as we increment changes in R_f. This allows us to stay well within the basin of attraction of the lowest

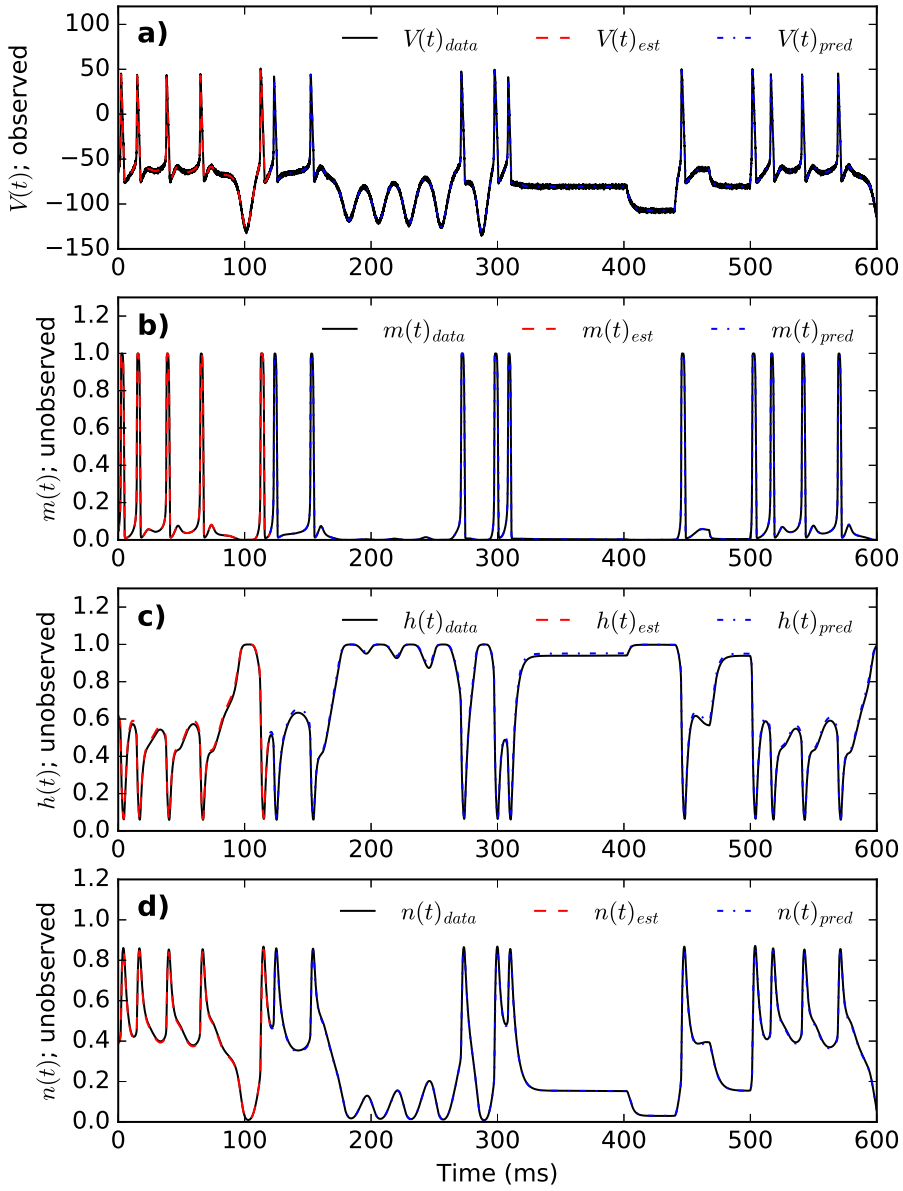

Figure 3.6: **(a, b)**: Data (black), estimated (red) and predicted (blue) state variables $V(t), m(t)$ for the NaKL model when only the noisy membrane voltage $V(t)$ is measured and presented to the model. **(c,d)**: Data (black), estimated (red) and predicted (blue) state variables $h(t), n(t)$ when only the noisy membrane voltage $V(t)$ is measured and presented to the model

Table 3.2: Known and estimated parameters for the NaKL model. We also display the bounds used for the nonlinear search algorithm.

Parameters	Known	Estimated	Search Lower Bound	Search Upper Bound
g_{Na}	120.0	108.4	50.0	200.0
E_{Na}	50.0	49.98	0.0	100.0
g_K	20.0	21.11	5.0	40.0
E_K	-77.0	-77.09	-100.0	-50.0
g_L	0.3	0.3028	0.1	1.0
E_L	-54.0	-54.05	-60.0	-50.0
C	0.8	0.81	0.5	1.5
V_m	-40.0	-40.24	-60.0	-30.0
ΔV_m	0.0667	0.0669	0.01	0.1
τ_{m0}	0.1	0.0949	0.05	0.25
τ_{m1}	0.4	0.4120	0.1	1.0
V_h	-60.0	-59.43	-70.0	-40.0
ΔV_h	-0.0667	-0.0702	-0.1	-0.01
τ_{h0}	1.0	1.0321	0.1	5.0
τ_{h1}	7.0	7.76	1.0	15.0
V_n	-55.0	-54.52	-70.0	-40.0
ΔV_n	0.0333	0.0328	0.01	0.1
τ_{n0}	1.0	1.06	0.1	5.0
τ_{n1}	5.0	4.97	2.0	12.0

action level.

The top panel of Fig. 3.7 displays the action level plot with the configuration above. When $25 \leq \beta \leq 39$, there are several different levels, and they reveal the expected action level after $\beta \geq 40$. The action level plot suggests that the voltage measurement alone is sufficient to determine the three unobserved state variables as well as the 19 parameters. The estimates of these parameters are displayed in Table 3.2.

We can select the values of R_{f0}'s for each state variable according to our knowledge about the amplitudes and time scales of the state variables by looking at the time series of solutions of the model. The time constants of gating variables characterize the their response to the change of voltage. The sodium activation variable

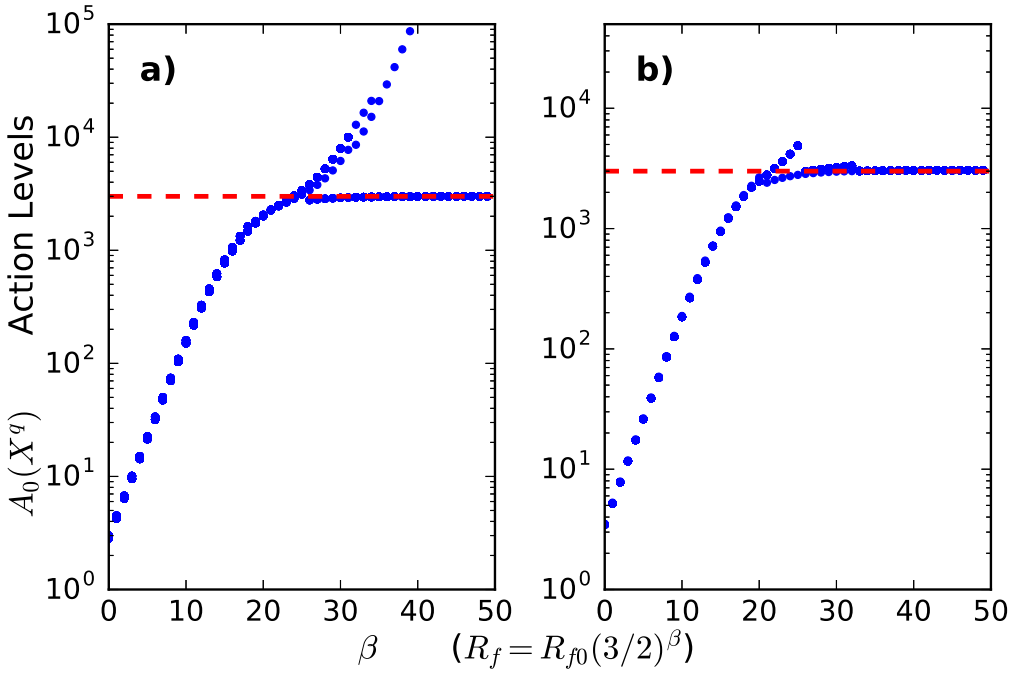

Figure 3.7: Action levels as a function of R_f for the NaKL model with only the noisy membrane voltage $V(t)$ measured and presented to the model. **(a)**: we selected $R_{f0}^{(V)} = 10^{-3}, R_{f0}^{(m)} = 10, R_{f0}^{(h)} = 10, R_{f0}^{(n)} = 10$ and $\alpha = 3/2$.; **(b)**: $R_m = 1, R_{f0}^{(V)} = 10^{-3}, R_{f0}^{(m)} = 10, R_{f0}^{(h)} = 1000, R_{f0}^{(n)} = 1000$, and $\alpha = 3/2$

$m(t)$ is the fastest, with a time constant of several hundreds of microseconds, which is a little bit slower than $V(t)$. $h(t)$ and $n(t)$ are much slower having a time constant of a few milliseconds. We set the ratio of $R_{f0}^{(m)}/R_{f0}^{(V)} = 5 \times 10^4$ and raised the ratio of $R_{f0}^{(n)}/R_{f0}^{(m)}$ and $R_{f0}^{(h)}/R_{f0}^{(m)}$ from 1 to 10 to compensate for the effects induced by different time constants. With $R_m = 1, R_{f0}^{(V)} = 10^{-3}, R_{f0}^{(m)} = 50, R_{f0}^{(h)} = 500, R_{f0}^{(n)} = 500$, the action level plot in the bottom panel of Fig. 3.7 shows this configuration of R_{f0} can effectively enforce that most saddle paths stay near the expected lowest action level.

The detailed action levels can depend on the choice of R_{f0}. In the NaKL example when the ratios of $R_{f0}^{(n)}/R_{f0}^{(m)}$ and $R_{f0}^{(h)}/R_{f0}^{(m)}$ are as large as 100, we often observed another action level with a value close to the lowest one. This can also depend on the specific choice of the N_0 initial paths with which we start the annealing.

A lesson we learn from this example is that we should take both the amplitude and the time scale of the state variables into consideration when selecting the scale of R_f values. The proper configurations of R_f values enlarge the probability to have the candidate paths converge to the expected action level, and also accelerate the convergence rate to the optimal paths.

3.4 Action Levels for Wrong Models

To give some sense of what one might expect if the model were `totally wrong`, we presented data from a collection of 1963 Lorenz model [72] oscillators oscillators to a Lorenz96 $D = 10$ model.

Twelve time series data are generated by four individual Lorenz63 [72] systems with different initial conditions. Gaussian white noise with zero mean and standard deviation $\sigma = 0.5$ are added to each time series. All these 'data' $y_l(t)$ are rescaled to lie in $[-10, 10]$.

We then place these signals as 'data' in the action with the model taken as Lorenz96 $D = 10$, the single forcing parameter is treated a time-dependent state variable obeying $\dot{f} = 0$. We use $L = 6$ as measurements using the data time series taken, $y_1(t), y_3(t), y_5(t), y_7(t), y_9(t), y_2(t)$. In Fig.3.8 (a) we display the action levels associated with this for $L = 6$. Results for other values of L are consistent with these. Another two common types in real data assimilation: fixed the parameter at wrong value and one or more terms in the formula are wrong. To simulate the two cases, we did the following two experiments:

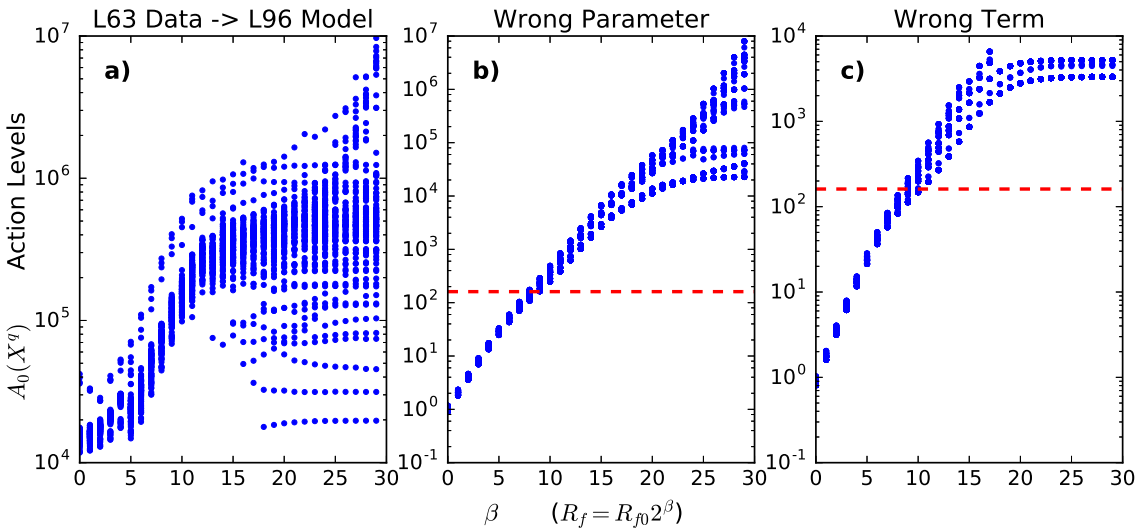

Figure 3.8: **a)** Action levels as a function of R_f for Lorenz96 model, $D = 10$, $R_{f0} = 0.01$, $L = 6$ when the wrong data is used for the Lorenz96 model. We actually used data from four realizations of the Lorenz63 model [72]. The action levels are also quite large, and, for $L = 6$, numerous and not well separated. **b)** The forcing parameter f in Lorenz96 model is fixed at wrong value. **c)** The decay term in Lorenz96 model $-x_i$ is set to be $-2x_i$. The structure of the action levels versus R_f here shows no trace of the minimum allowed level Eq.(2.22) and indicates the data and the model are incompatible.

1. Wrong parameter in Lorenz96 Model: $F = 18$ instead of 8.17

$$\dot{x_i} = -x_{i-1}(x_{i-2} - x_{i+1}) - x_i + F$$

2. Wrong term in Lorenz96 Model: $\eta = 2$ instead of $\eta = 1$

$$\dot{x_i} = -x_{i-1}(x_{i-2} - x_{i+1}) - \eta x_i + F$$

The twin data are generated with the right model with $f = 8.17$ and correct term $-x_i$. The two wrong models defined above are used to assimilate the generated data. And the action levels are displayed in Fig.3.8 (b) and (c). Those two cases can be clearly

identified that no action level goes to the expected action level. The structure of the action levels versus R_f here shows no trace of the minimum allowed level Eq.(2.22) and indicates the data and the model are incompatible.

Part of Chapter 3 is a reprint of the material as it appears in Jingxin Ye, Daniel Rey, Nirag Kadakia, Michael Eldridge, Uriel I. Morone, Paul J. Rozdeba, Henry D. I. Abarbanel, John C. Quinn Systematic variational method for statistical nonlinear state and parameter estimation, *Physical Review E*, 92(5), 052901 (2015) and Jingxin Ye, Nirag Kadakia, Paul J. Rozdeba, Henry D. I. Abarbanel, John C Quinn, Precision Variational Approximations in Statistical Data Assimilation, *Nonlinear Processes in Geophysics*, 22 (2), 205-213 (2015). The dissertation author was the primary investigator and author of this paper.

Chapter 4

Discussion

4.1 Corrections to the Approximation of the Dominant Saddle Path $\mathbf{X}^q$ to $\langle G(\mathbf{X})\rangle$

The path integral formulation of $\langle G(\mathbf{X})\rangle$ allows more than just the leading variational approximation to this expected value, as is always the case in the Laplace method. The idea is to identify the path $\mathbf{X}^0$ with the smallest action level and then expand the integral in $\mathbf{X}$ about $\mathbf{X}^0$.

Near $\mathbf{X}^0$ we write

$$A_0(\mathbf{X}) = A_0(\mathbf{X}^0) + (\mathbf{X}-\mathbf{X}^0)_{\alpha_1}\gamma(\mathbf{X}^0)^2_{\alpha_1\alpha_2}(\mathbf{X}-\mathbf{X}^0)_{\alpha_2} + \cdots , \tag{4.1}$$

and the Hessian matrix $\gamma(\mathbf{X}^0)^2 = A_0^{(2)}(\mathbf{X}^0)/2$ is positive definite, if $\mathbf{X}^0$ gives a minimum of the action.

Changing integration variables to $U_\alpha = \gamma_{\alpha\beta}(\mathbf{X}^0)(\mathbf{X}-\mathbf{X}^0)_\beta$ leads to the numer-

ator of $\langle G(\mathbf{X})\rangle$ in Eq. (A.2) arising from $\mathbf{X}^0$

$$\int d\mathbf{X}\, \exp[-A_0(\mathbf{X})]G(\mathbf{X}) = \frac{\exp\left[-A_0(\mathbf{X}^0)\right]}{\det\gamma(\mathbf{X}^0)} \int d\mathbf{U}\, \exp\left(-\mathbf{U}^2 - V\right)\left[G(\mathbf{X}^0) + W\right] \tag{4.2}$$

where

$$V(\mathbf{U},\mathbf{X}^0) = \sum_{r=3} \frac{A^{(r)}(\mathbf{X}^0)_{\alpha_1\ldots\alpha_r}}{r!}[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]_{\alpha_1}\cdots[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]_{\alpha_r},$$
$$W(\mathbf{U},\mathbf{X}^0) = \sum_{k=1} \frac{G^{(k)}(\mathbf{X}^0)_{\alpha_1\ldots\alpha_k}}{k!}[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]_{\alpha_1}\cdots[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]_{\alpha_k}.$$

In the denominator we replace the numerator term $G(\mathbf{X}) = G(\mathbf{X}^0) + W(\mathbf{U},\mathbf{X}^0)$ by unity.

The terms in this integral are evaluated by expanding the Taylor series contributions $V(\mathbf{U},\mathbf{X}^0)$ and $W(\mathbf{U},\mathbf{X}^0)$ in powers of $\mathbf{U}$, and performing the resulting Gaussian integrals in $\mathbf{U}$. Terms with odd powers of $\mathbf{U}$ vanish by symmetry. The contributions to leading order in $1/R_f$ coming from the path $\mathbf{X}^0$ for $E[G(\mathbf{X})|\mathbf{Y}]$ are

$$\begin{aligned} E[G(\mathbf{X})|\mathbf{Y}] =& G(\mathbf{X}^0) + \int \frac{d\mathbf{U}\ \exp\left(-\mathbf{U}^2\right)}{\sqrt{\pi^{(m+1)D}}} \left\{ \frac{1}{2}G^{(2)}(\mathbf{X}^0)[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]^2 \right. \\ & - \frac{A^{(4)}(\mathbf{X}^0)}{24}G(\mathbf{X}^0)[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]^4 - \frac{A^{(3)}(\mathbf{X}^0)}{6}G^{(1)}(\mathbf{X}^0)[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]^4 \\ & \left. + \frac{A^{(3)}(\mathbf{X}^0)^2}{72}G(\mathbf{X}^0)[\gamma(\mathbf{X}^0)^{-1}\mathbf{U}]^6 \right\} + O\left(\frac{1}{R_f^2}\right) \end{aligned} \tag{4.3}$$

From the form of eion in Eq. (2.14) we see that for large R_f each factor of $\gamma(\mathbf{X}^0)$ is of order $\sqrt{R_f}$, and each 'vertex' $A_0^{(4)}(\mathbf{X}^0)$ and $A_0^{(3)}(\mathbf{X}^0)$ is of order R_f. The

Gaussian integrals over $\mathbf{U}$ are discussed in the opening chapters of Ref.[103].

Statistics such as the covariance about the saddle path $\mathbf{X}^0$ may be evaluated by selecting $G(\mathbf{X})$ to be the matrix in path $(a, n) = \alpha$ space as

$$G(\mathbf{X}) = (\mathbf{X} - \mathbf{X}^0)_\alpha \, (\mathbf{X} - \mathbf{X}^0)_\beta = U_\eta U_\kappa (\gamma (\mathbf{X}^0)^{-1})_{\kappa\,\alpha} \, (\gamma (\mathbf{X}^0)^{-1})_{\eta\,\beta}.$$

The Gaussian integral

$$\int d\mathbf{U} \; U_\alpha U_\beta \; \exp\left(-\mathbf{U}^2\right), \tag{4.4}$$

is easily performed, giving

$$\langle (\mathbf{X} - \mathbf{X}^0)_\alpha \, (\mathbf{X} - \mathbf{X}^0)_\beta \rangle = \frac{1}{2} (\gamma^2 (\mathbf{X}^0)^{-1})_{\alpha\,\beta}, \tag{4.5}$$

which behaves as $1/R_f$ for large R_f, implying a steep, narrow minimum in path space.

These results are specific for the form of the action in Eq. (2.14), and may not apply for other choices for the distribution of noise in the measurements or the distribution of the errors in the models. Distributions with power law or 'fat' tails require additional scrutiny.

4.2 Effects of Distribution Tail on Action Levels

In the previous discussion, we assume additive Gaussian errors in the measurement and model errors. We explore here a non-Gaussian distribution for model errors (can also be used for measurement errors) Cauchy-Lorentz distribution, allowing us to assess the influence of "fat" tails in the action for data assimilation problems.

When we looked at a model error that obeys Cauchy-Lorentz distribution

$$P_C(x) \propto \left(1 + \frac{1}{2} R_f x^2\right)^{-p}, \tag{4.6}$$

we found that the saddle path $\mathbf{X}^0$ no longer appeared to have corrections acting as inverse powers of R_f.

The Gaussian and the Cauchy distributions are two examples of distributions that they become delta function of x when $R_f \to \infty$. And they differ by the "fatness" of their tails. Gaussian distribution decays as $\exp(-x^2)$, while the Cauchy distribution acts as x^{-2p} for large x which gives a much "fatter" tail than a Gaussian.

We investigate an interpolating distribution for $\lambda > 0$,

$$P_I(x) \propto \exp\left[-p\frac{(1+\alpha x^2)^\lambda - 1}{\lambda}\right], \tag{4.7}$$

which is a Gaussian distribution when $\lambda = 1$ and a Cauchy distribution when $\lambda \to 0$. Thus $P_I(x)$ has familiar features of smooth distribution that approximate delta functions. If we select this distribution for the model errors and retain a Gaussian for the measurement errors, the action can be written as

$$\begin{aligned} A_0(\mathbf{X}) = &\sum_{n=0}^{m}\sum_{l=1}^{L} \frac{R_m(n,l)}{2}(y_l(n) - x_l(n))^2 - \log[P(\mathbf{x}(0)] \\ &+ \sum_{n=0}^{m-1}\sum_{a=1}^{D} \frac{p}{\lambda}\left[\left(1 + \frac{R_f(a)}{2}(x_a(n+1) - f_a(\mathbf{x}(n))^2\right)^\lambda - 1\right] \end{aligned} \tag{4.8}$$

Here we choose $\lambda = 5 \times 10^{-3},\ 0.5,\ 1,\ 3$ to study the effects of the distribution tail on action level. The same calculation was did for Lorenz96 model $D = 5, L = 2$ as in Section 2.4 but with the new nongassian action Eq. (4.8). (Fig. 4.1)

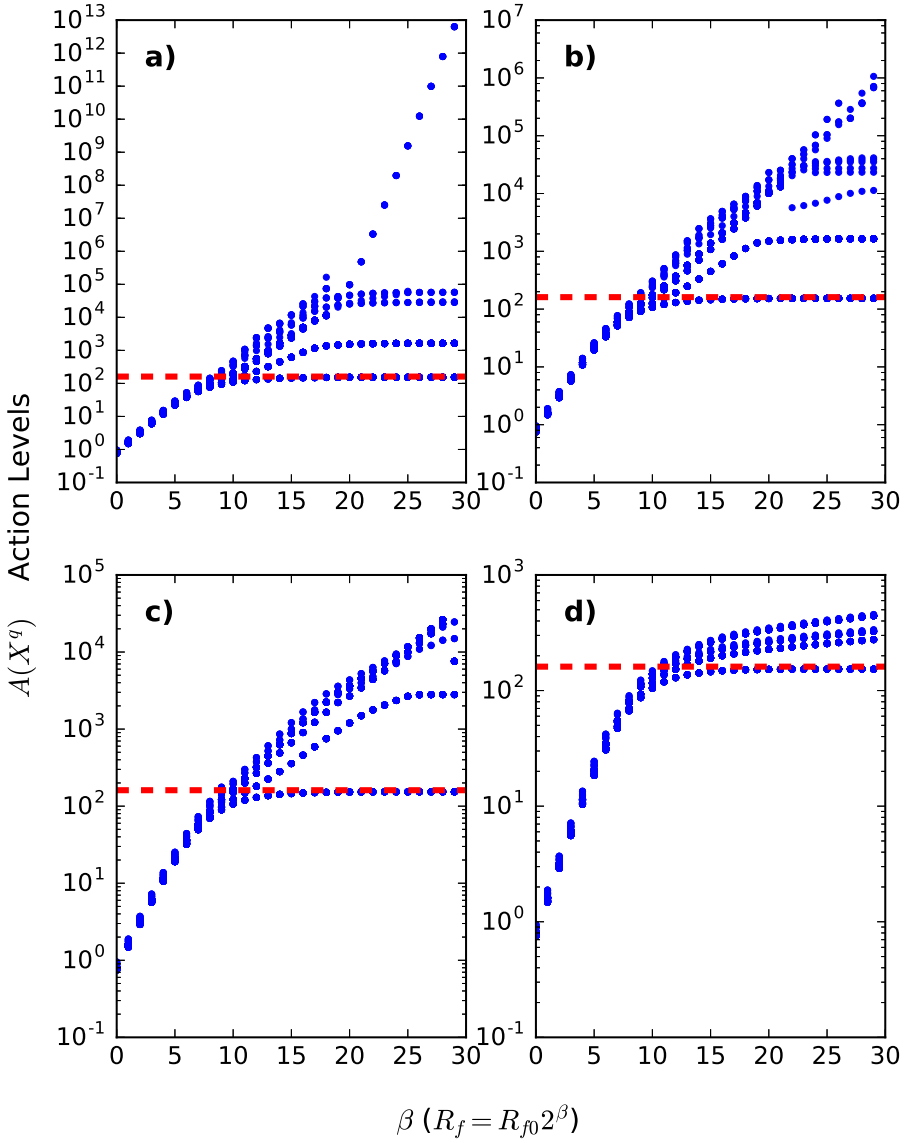

Figure 4.1: Action levels as a function of R_f for Lorenz96 model, $D = 5, L = 2$, $R_{f0} = 0.01$. Different values of λ are used in the Eq.(4.8): **a)** $\lambda = 3$; **b)** $\lambda = 1$; **c)** $\lambda = 0.5$; **d)** $\lambda = 5 \times 10^{-3}$;

The action levels diverges to 10^{13} at high R_f for $\lambda = 3$. As λ decreases, it slows down the procedure for the lowest action level approaching the expected action level value. For $\lambda = 3$, it reaches the expected action level at $\beta = 12$, and the corresponding $\beta = 18$ for $\lambda = 5 \times 10^{-3}$. In Panel d) of Fig. 4.1, the behavior of the action levels are largely regularized: only four action levels appear. Small λ would be useful to control the divergence of action levels for some sensitive models.

Another important thing needs to be pointed out that the λ has little effects on the convergence of the numerical optimization procedure. The numbers of paths out of 100 candidate paths that converges to the lowest action level at different λ values are listed in Table 4.1.

Table 4.1: For Lorenz96 model $D = 5, L = 2$, the numbers of paths out of 100 candidate paths that converges to the lowest action level at different λ values.

λ	Num. of Measurements (L)	Lowest Action ($A_0(\mathbf{X}^0)$)	Num. of Paths
1	2	137	2
3	2	137	4
0.5	2	137	3
5×10^{-3}	2	137	2
5×10^{-4}	2	137	2
5×10^{-5}	2	137	3

4.3 Connection with Bröcker's Results

The annealing method has a close relationship, and in some places a significant overlap with two very nice papers [15, 17] and a quite valuable and pedagogical Summer School presentation by J. Bröcker [16]. He considers the formulation of the assimilation problem in continuous time and identifies an action which corresponds to

our $A_0(\mathbf{X})$.

Missing in his discussions is the context of the path integral to be used to evaluate expected values of functions on the path, so no method is presented to evaluate the accuracy of the variational result as a function, say, of R_f. He identifies the boundary conditions on the Euler-Lagrange equations for the extrema of the action, and recognizes with great care how one selects among the possibilities.

His formulation of the actual action differs from ours in that he adds to the deterministic equations $\dot{\mathbf{x}}(t) = \mathbf{F}(\mathbf{x}(t))$ a control term which is employed in moving the output of a model toward the observations, then he imposes this control via a Lagrange multiplier. The Euler-Lagrange equations in Eq. (11) of Ref.[16] are, happily, in his notation the same as our Eq. (2.11). In Section 3, especially Exercise 3.1 of Bröcker's Summer School notes [16] he shows that his approach and our yield the same extremum conditions on the action.

Importantly he recognizes and explores with insight the manner in which the model error term in the action and the measurement error term in the action 'balance' to direct the solution to a chaotic model equation to those regions of phase space where the observations provide information about the model solutions.

Bröcker does not consider the question of how many measurements are required to allow the search for extrema of the action to be achievable [2] or whether any of the allowed extrema are, in fact, the overall minimum of the action. He does stress the importance of the model error term in the action and formulates its appearance in a very clear and useful fashion.

Our use of an annealing method to address these latter two questions has roots in our own work [3] and has connections with Bröcker's work. The route we follow now

is to return to discrete time formulations of the action with a focus on the questions one wishes to answer in the use of the data assimilation methods in physical and biological problems: (1) What is the expected value of the state and parameters of the model system at the end of an observation window–this includes unmeasured as well as measured states. (2) What are the RMS errors about this expected state ? (3) What is the accuracy, including RMS errors, of the predictions of the model, conditioned on information transferred to it by the data, for times after the observation window.

4.4 Comparison with the quasi-static variational assimilation (QSVA) Method

There is another strategy for determining the best path for a selected model with chaotic trajectories given observed data known as the quasi-static variational assimilation (QSVA) method [78]. In this approach the known equations of motion are initialized with some $\mathbf{x}^{(0)}(t=0)$ and integrated forward a small step in time of length τ. A cost function comparing the known observations with the model output over that time step τ is minimized to adjust the initial condition $\mathbf{x}^{(0)}(t=0) \rightarrow \mathbf{x}^{(1)}(t=0)$. Then the time interval is extended to 2τ and starting with $\mathbf{x}^{(1)}(t=0)$ the equations of motion are integrated forward to 2τ. The difference between the orbit from the initial condition $\mathbf{x}^{(1)}(t=0)$ is compared via the cost function to data in the longer interval 2τ and the cost function is minimized taking $\mathbf{x}^{(1)}(t=0) \rightarrow \mathbf{x}^{(2)}(t=0)$. This repeated adding increments of time to the observation window until it is $N\tau$ long, and a final $\mathbf{x}^{(N-1)}(t=0)$ is found via minimizing the cost function over the interval up to $N\tau$. Starting with this final selection of initial condition $\mathbf{x}^{(N-1)}(t=0)$ a trajectory to time

$N\tau$ and beyond is generated using the equations of motion. This gives a path through the estimation window $[0, N\tau]$ and beyond for prediction. By choosing the original initial condition $\mathbf{x}^{(0)}(t=0)$ within a resolution ball of uncertainty and drawing N_0 different selections, one can generate N_0 different paths during both the estimation and prediction windows. In the paper [78] the Lorenz 1963 model was used with parameters set to produce chaotic orbits, and it was assumed all the state variables were observed.

We have made a direct comparison of the QSVA approach with our annealing method. We used the Lorenz96 model with $D = 5$ and $L = 3$ to generate 'data', and then we added noise of the same level with variance $\sigma^2 = 1/4$ to the observations. We observed L = 3, $y_1(t), y_3(t)$, and $y_5(t)$ and in choosing $\mathbf{x}^{(0)}(t=0)$ for QSVA we selected the unobserved components $y_2(0), y_4(0)$ from a uniform distribution in the interval $[-10, 10]$. We then evaluated (1) $N_0 = 100$ initial conditions arriving at 100 final selections of initial conditions using time steps of $\tau = 4\Delta t = 0.1$ where $\Delta t = 0.025$ and (2) a second numerical trial with $\tau = 4\Delta t = 0.004$ where $\Delta t = 0.001$ with $N_0 = 60$ initial conditions. We generated the data with a fourth-order Runge-Kutta integrator for the $D = 5$ Lorenz96 equations with forcing $f = 8.17$. The minimization of the cost function

$$C(\mathbf{x}(0), N) = \sum_{n=0}^{N} \sum_{l=1}^{L} (x_l(n) - y_l(n))^2 \tag{4.9}$$

at each step to find an improved $\mathbf{x}^{(K)}(0)$ was performed using a quasi-Newton BFGS method [80]. The estimation window was taken as 160 $\Delta t = 4$, and the prediction window ranged from 4 to 8.

We compared this calculation of initial conditions and estimated and predicted orbits with the annealing method described in Sec.2.4 using annealing up to $\beta = 30$

as shown in the right panel of Fig. 2.2.

To compare the outcome of the two approaches we display the following results for the first set of calculations with $\Delta t = 0.025$ and $N_0 = 100$:

- In the autonomous Lorenz96 model $D = 5$ there is one unstable direction with positive Lyapunov exponent $\lambda = 0.53$ and one neutral direction as it is a differential equation. We projected the $N_0 = 100$ solutions from QSVA into the plane of these two directions along with the location in the same plane of the first state $\mathbf{x}(0)$ from the annealing process. These are shown in Fig. 4.2.

- The outcome of the annealing estimations gives the whole path in the time interval [0,4], including $\mathbf{x}(t = 0)$ and $\mathbf{x}(t = 4)$ at the start and end of the interval, respectively. The predictions of annealing method are obtained in two ways: one integrates dynamical model Eq. (2.24) forward from $\mathbf{x}(t = 0)$ and the other one starts from the end of the estimation window $\mathbf{x}(t = 4)$ directly which can largely reduce the numerical errors introduced by chaos. (Fig. 4.3) However, QSVA only allows the first option.

- We evaluated the RMS error for both the annealing and QSVA calculations by comparing the model output in all state variables both in the prediction window. The prediction RMS are defined as

$$\mathrm{RMS} = \sqrt{\frac{1}{DN} \sum_{n=N+1}^{2N} \sum_{a=1}^{D} (x_a(n) - y_a(n))^2}.$$

We performed a second set of calculations with $\Delta t = 0.001$ and $N_0 = 60$ initial conditions. We projected the $N_0 = 60$ solutions from QSVA into the plane of these

two directions, unstable and neutral, along with the location in the same plane of the first state $\mathbf{x}(0)$ from the annealing process. The projection and a zooming in on the small values of the error are shown in Fig. 4.2. The histogram of RMS errors during the prediction window are then shown in Fig. 4.3.

Our conclusion is that the QSVA method, while straightforward to implement, may not do as well in the realistic case of noisy sparse data, $L < D$, which is likely to be encountered in realistic situations. It is clear that as one decreases the time step between iterations of the QSVA protocol, the error as seen in the projection onto the unstable and neutral directions at the end of the assimilation window decreases, yet there are quite a few instances when the error is quite large. Similarly, in the histogram of the RMS errors in the prediction window, as shown in Fig. 4.2 shows much better results than for the larger value of Δt in the first calculation.

The annealing results are rather the same in each case with a clear clustering of errors in the neutral/unstable plane near small errors, and a tight cluster of RMS prediction errors. We can conclude that the annealing method produces a much narrower distribution of candidates for the path with smallest action level when dealing with noisy, sparse data than the QSVA method in the same conditions. QSVA does produce a selection of paths with excellent initial conditions for prediction, but these come along with paths quite far from those with small errors. How one is to choose among the paths with large deviations from small errors is not entirely clear in the QSVA algorithm applied to the circumstances presented here.

The two critical differences in this comparison are that the annealing methods seeks `paths` as the outcome of the variational principles, and it surveys those paths and selects that which gives rise to the minimum observed value of the action. It

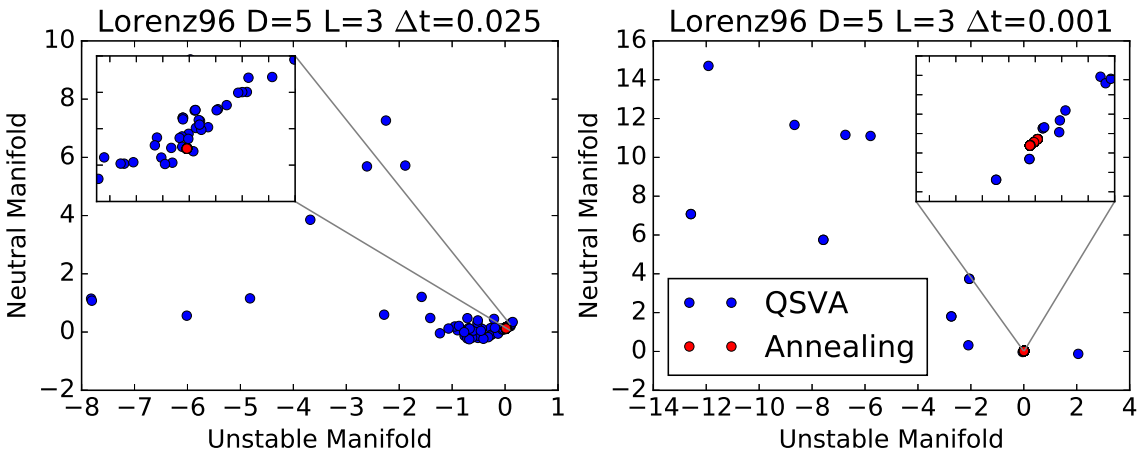

Figure 4.2: Lorenz96, D = 5, L = 3, projection of $\mathbf{x}(0)$ on the neutral W_0 and unstable W_u manifolds for annealing with $\beta = 30$ and for QSVA; $N_0 = 100$ and $\Delta t = 0.025$. **a)**: large time step $\Delta t = 0.025$ is used; **b)**: fine time step $\Delta t = 0.001$ is presented.

should also be noted, as we mentioned earlier in the paper, that when one looks for a path with minimum action in the deterministic case, called strong 4DVar or large R_f, the procedure does not find the desired minimum [81]. QSVA utilizes models with no model error ($R_f \to \infty$) throughout.

Part of Chapter 4 is a reprint of the material as it appears in Jingxin Ye, Daniel Rey, Nirag Kadakia, Michael Eldridge, Uriel I. Morone, Paul J. Rozdeba, Henry D. I. Abarbanel, John C. Quinn Systematic variational method for statistical nonlinear state and parameter estimation, *Physical Review E*, 92(5), 052901 (2015) and Jingxin Ye, Nirag Kadakia, Paul J. Rozdeba, Henry D. I. Abarbanel, John C Quinn, Precision Variational Approximations in Statistical Data Assimilation, *Nonlinear Processes in Geophysics*, 22 (2), 205-213 (2015). The dissertation author was the primary investigator and author of this paper.

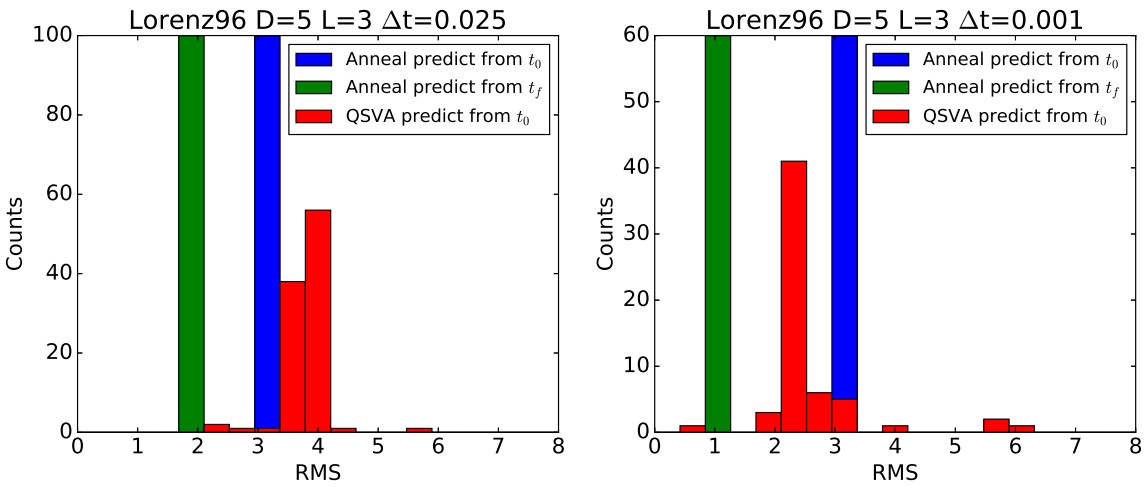

Figure 4.3: Lorenz96, D = 5, L = 3, histogram of RMS prediction error for annealing with $\beta = 30$ and for QSVA with $\Delta t = 0.025$, $N_0 = 100$, and histogram of RMS prediction error for annealing with $\beta = 30$ and for QSVA with $\Delta t = 0.001$, $N_0 = 60$.

Chapter 5

HVc Neuron Model with Calcium Dynamics

5.1 Dynamical Model Completion

In developing and using methods of statistical data assimilation to characterize the biophysical properties of functional networks of neurons, we have previously built a Hodgkin-Huxley type model (HH model) of the dynamics of individual neurons [46, 51, 4] . This model, as all such models, has numerous unknown fixed parameters that must be determined for each class of neuron under consideration. We used well designed stimulating currents for individual neurons in the avian song system nucleus HVC and measured the response of the membrane voltage to estimate the many biophysical parameters in the voltage and kinetic equations of such a model. This estimation 'completes' the model in the sense that once the fixed parameters are established, then given initial conditions for the state variables, observed and unobserved, we can predict the response of the model neuron to a new stimulus [4].

We tested/validated the biophysical HH model by showing that with the estimated parameters it could reliably predict the observed response to new stimulating currents. The required initial conditions for prediction in the completed model were established by using a very short (100 ms) segment of the data set for times beyond the observation window.

The methods we utilize here were quite instrumental in designing the large collection of data sets analyzed in [4], and we expect that to be the case again when we move from simulations of neurons with important Ca dynamics to the design of experiments to explore those dynamics.

Our stimulation/response protocols presented data comprising an applied stimulating current $I_{\text{app}}(t)$ and the observed response membrane voltage $V(t)_{\text{data}}$. These time series alone allow estimation of the fixed parameters and unobserved state variables of the neuron. The latter are the voltage dependent gating variables associated with ionic currents of Na^+, K^+, and Ca^{2+} ions.

The estimation procedure seeks to minimize a distance between the output of the HH model $V(t)$ and the observed data $V(t)_{\text{data}}$ over an observation window $[0, T]$ where we selected $T = 1500\,\text{ms}$ [4]. The cost function or objective function representing the error between the data and the model output is taken as

$$\sum_{n=0}^{m}(V(t_n)_{\text{data}} - V(t_n))^2, \tag{5.1}$$

where observations are made at times $t_n;\ n = 0, 1, 2, ..., t_m = T$. This objective function was minimized subject to the deterministic HH equations of motion, given below, as equality constraints dynamically taking the states between time t_n and time t_{n+1}. This cost function is also seen as the error in the synchronization of the model

output with the data.

The measurement in the experiments above only includes membrane voltage. That has been the conventional way to determine properties of neurons since the 1950s when the HH model was first built.[46] Although we did not encounter models of this form in our previous analysis of the HVC data [4], it is not clear that the single measurement of membrane voltage will suffice for this estimation procedure. As we will show here, even when very well sampled in time, a single measurement may simply contain insufficient information to determine the unmeasured state variables. The systems where more measurements are required appear to be those where chaotic solutions to the dynamical equations are possible for some biophysically plausible set of stimuli and model parameters. We found that Ca^{2+} dynamics could cause failure in determining properties of neurons with voltage measurement only.

Many examples are known of the necessity for more measurements at each measurement time to remove impediments associated with the instability of the manifold in state space where the data and the model output are synchronized [4]. When synchronization fails, the synchronization error Eq. (5.1) has multiple local minima as a function of the parameter or state value sought through the minimization. This impediment to estimation must be regulated to provide a smooth surface on which one implements a search procedure for the minimum of the cost function Eq. (5.1).

There is now substantial evidence of the role of Ca^{2+} dynamics in the neurons of HVC [96, 39, 53, 22, 70]. This adds a set of rather slow dynamical processes to the much faster voltage gated processes involving Na^{+} and K^{+} ions. When we extended these voltage dynamics models to include the important biophysical processes of Ca^{2+}

uptake and release from internal stores, the question of how many measurements are required changes. Mixing slow and fast dynamical processes, coupled nonlinearly to each other, can be a setting for the appearance of chaotic behavior. In the case of voltage plus calcium dynamics chaos does appear.

The appearance of chaos in calcium oscillations is quite natural. Calcium bursting behavior characterized by an irregular number of secondary spikes and irregular spacing between the initial spikes and nonperodic oscillations with varying amplitudes are observed widely under different stimuli.[23, 83, 41] Seemingly erratic, or irregular, time-series behavior with little apparent structure in the time course can be generated by a deterministic system operating in chaotic regime, This inspired many analyses of chaotic calcium models. [14, 47, 25] Houart et al. produced chaos by extending the simpler Dupont-Goldbeter [26] model to account for fluctuating IP_3 concentrations. [47] The IP_3 concentration is synthesized by the introduction Ca^{2+} ions, and in turn prompts Ca^{2+} release from internal stores. This simple feedback leads to oscillatory behavior that results in chaos. Borghans et al. found similar chaotic behavior when accounting for a variable IP_3 concentration. [14] Feedback dynamics can easily result in systems with chaotic regimes so it is instructive to analyze the difficulties of estimating such a system.

This leads to a situation where more than voltage measurements alone are required

- to synchronize data with model output,
- accurately estimate fixed parameters and unobserved state variables, and
- provide accurate predictions as validation of the consistency of the model with the observations.

The goal of this paper is to determine the difficulties of estimating model parameters and state variables in a model that exhibits chaotic oscillations. The model presented is not intended to be a quantitatively accurate description of real neurons. Rather, our more modest aim is to present a system which replicates qualitative features that we anticipate in the HVC neurons of the avian song system. HVC and the other nuclei of the song system are our target neurons for the experiments we plan [4], so our focus on their properties will appear several times within the general issues in this paper.

In particular, our model exhibits spiking bursts and has regimes of chaotic behavior. These features make it challenging to correctly estimate parameters and unmeasured state variables, as was described in [4]. Indeed, we show that the model in this paper requires more than just voltage measurements to completely estimate the full state of the model and use those estimates to make accurate predictions as a test of model validity.

5.2 The Biophysical Role of Ca^{2+} Dynamics

Calcium ions play an important role in regulating a great variety of neuronal processes. Calcium can act in signal transduction resulting from activation of ion channels or as a second messenger caused by indirect signal transduction pathways. Intracellular calcium signals regulate processes that operate over a wide range of time scales, from neurotransmitter release at the microsecond scale to gene transcription, which lasts for minutes and hours [12]. In the HVC of the songbird, the contributions of calcium channels and calcium mediated events to spiking and bursting have been observed. In vivo, L-type Ca^{2+} bursting activity in HVC_{RA} neurons [70], and calcium

transients show strong preference for the presentation of the bird's own song in identified HVC neurons where a strong correspondence between calcium signals and juxtacellular electrical activity is exhibited [39]. In vitro, T-type low voltage activated Ca channels are expressed in HVC neurons contributing to their postinhibitory rebound firing [22]. Moreover, the spike frequency adaptation seen in HVC projection neurons is largely due to a Ca^{2+} induced K channel [22, 63]. In addition multiple calcium binding proteins that determine the dynamics of free calcium inside neurons [86] are enriched in the HVC of songbirds [96, 53]. All these studies motivate an investigation of the roles that Ca^{2+} ions play in the electrical activity of HVC neuronal subpopulations in vitro and during singing.

Calcium is a crucial intracellular messenger in mammalian neurons where the final transduction of any neuronal signal involves the movement of calcium ions. At rest, the intracellular calcium concentration of most neurons is about 50∼100 nM, and that can rise to levels that are ten to 100 times higher during electrical activity [11]. The cytosolic calcium concentration is determined by the balance between calcium influx and efflux as well as by the exchange of calcium with internal stores. In addition, calcium-binding proteins such as parvalbumin or calretinin, acting as calcium buffers, determine the dynamics of free calcium inside neurons. Most importantly, only the free calcium ions inside the cytosol are biologically active.

Calcium influx from the extracellular space is controlled by various mechanisms including voltage gated calcium channels, ionotropic glutamate receptors, nicotinic acetylcholine receptors, and transient receptor potential type C channels [42, 34, 44, 82]. The extrusion of calcium ions from the cytosol is done via the plasma membrane calcium ATPase and the sodium-calcium exchanger [12]. The release of messenger

calcium ions from internal stores, mostly the endoplasmic reticulum (ER), is controlled by the inositol trisphosphate receptors and ryanodine receptors [10]. The high calcium level inside the ER is controlled by the sarco-endoplasmic reticulum calcium ATPase pump that transports calcium ions from the cytosol to the ER.

In addition to the ER, mitochondria can also play the role of calcium buffers by absorbing calcium ions during cytosolic calcium elevations via the calcium uniporter and then releasing the calcium ions back to the cytosol slowly via the sodium-calcium exchange [24].

Variation in the intracellular concentration of Ca^{2+} ions, $[Ca^{2+}]_i(t) \equiv Z_{Ca}(t)$, is governed by the flow of these ions through the cell membrane via voltage gated channels, as well as by uptake and release by the ER as a storage device. These properties and more are discussed in many research papers [33, 12], including a very informative review and summary by Falcke and colleagues [93].

Earlier work on coupling voltage and calcium dynamics [31, 76], as well as the dynamics of Ca^{2+} uptake and release [47] independent of its connection to voltage dynamics of Na^+ and K^+ ion flow, provide the foundation for the model we discuss here. The presence of Ca^{2+} channels and their interaction with voltage dynamics in HVC neuron cells has been established by [22]. These results have strongly motivated us to explore the inclusion of Ca^{2+} dynamics in the model utilized in conjunction with our analysis of experiments on HVC neurons [4].

Ca^{2+} ions are released and stored in the ER predominately through the mediation of inositol 1,4,5-triphosphate (IP_3) in many cell types [93], and we will incorporate a model of these processes in our overall model of the cellular dynamics. This uptake and release via IP_3 mediation has been suggested as a source of Ca^{2+} oscillations in

the work of Houart, et al [47], and we adopt their model, with a faster time scale, compared to their original idea of its role in more global rhythms of animals. This is not the only candidate for incorporating calcium dynamics into neuronal processes; we have analyzed two other models of Ca dynamics with some care [31, 76], but we do not report on them here.

Examples of the important role of calcium in intracellular dynamics abound. For instance, the permeability of some potassium channels in the cellular membrane are affected by the presence of calcium, and membrane calcium channels themselves may be voltage-gated [51, 84]. The interplay between these two mechanisms acts to regulate firing patterns [18]. Additionally, calcium is known to be a major determinant in the potentiation and depression of excitatory synaptic strength, and thus is expected to play an important role in memory and learning. This is widely thought to underly how networks of neurons "rewire" and learn. While these processes are certainly important, they act on a much longer time scale than the interacting voltage and calcium processes we include in our model.

The main issues we address in this paper are those raised when the relatively slow dynamics associated with Ca^{2+} ion uptake and release interact with the much faster voltage dynamics to produce chaotic behavior. These are issues in using methods of data assimilation to estimate parameters and unobserved state variables using time series of observed quantities from experiments.

We do not yet use any of these models in the analysis of experiments in this paper, but we perform "twin experiments". In these we generate data by solving the model with known parameters, and then by presenting observables such as membrane voltage $V(t)$ or intracellular Ca^{2+} concentration $Z_{\mathrm{Ca}}(t)$ to the model, we are able to

examine our estimation methods in a controlled context.

When additional complex dynamics is introduced into a neuron model, such as the intracellular calcium dynamics we present here, the synchronization manifold of the model may become unstable [5]. This can make it impossible to estimate unknown parameters and unobserved states with voltage measurements alone, raising the following interesting questions:

- How do calcium dynamics effect neuron behavior?
- How many measurements are needed to synchronize such a neuron model with observed data after the introduction of calcium dynamics?
- If other measurements are required in addition to membrane voltage, what can play that role?

We do not have full answers to these questions. However, in this paper, we construct a conductance-based neuron model, into which we couple a detailed model of intracellular calcium dynamics. Using this model, we investigate the estimation problems through numerical experiments to give a clear image of the biophysical issues raised via calcium dynamics. We will show that voltage measurements are not enough to 'cure' the instabilities just noted, and we will address a solution to this. When we move from simulations to experimental data, we expect to encounter the same issues in a richer context.

5.3 Models of Voltage Plus Calcium Dynamics

5.3.1 Dynamics of Voltage Dependent Channels

We now describe the coupled model of voltage and Ca^{2+} dynamics, beginning with the membrane voltage and currents. This is a single-compartment point neuron model, with ion channels comprising standard Na^+, K^+, and leak channels coupled to additional currents seen in the work of [22]: two voltage gated Ca^{2+} channels $I_{\mathrm{CaL}}(t)$ and $I_{\mathrm{CaT}}(t)$; an h channel $I_h(t)$ with a long time constant and operating primarily in the hyperpolarized regime; and, crucially, a potassium channel $I_{K/Ca}$ whose conductance depends on the intracellular calcium concentration $Z_{\mathrm{Ca}}(t)$. The consequences of the presence of $I_{K/Ca}$ are outlined in Section (5.3.2).

Table 5.1: Parameter values in the voltage dynamics: g_x's are the conductances of currents, E_x's are reversal potentials, $\kappa_{K/Ca}$ is the threshold for Hill function, and C_m is the membrane capacitance.

g_{Na}	450 nS	E_{Na}	45 mV	g_K	50 nS	E_K	-90 mV
g_{CaL}	10 nS	E_{CaL}	85 mV	g_{CaT}	3 nS	E_{CaT}	85 mV
$g_{K/Ca}$	2 nS	E_K	-90 mV	g_h	4 nS	E_h	-30 mV
g_L	2 nS	E_L	-70 mV	$\kappa_{K/Ca}$	0.35 μM	C_m	100 pF

With these currents, the HH equation for the voltage dynamics has the form

$$C_m \frac{dV(t)}{dt} = I_{Na}(t)+I_K(t)+I_{\mathrm{leak}}(t)+I_{\mathrm{CaL}}(t)+I_{\mathrm{CaT}}(t)+I_{K/Ca}(t)+I_h(t)+I_{app}(t), \quad (5.2)$$

where C_m is the membrane capacitance and $I_{app}(t)$ is an externally applied current

selected by us. The ion currents themselves are given by

$$I_{Na}(t) = g_{Na}\left[m_0(V(t))\right]^3 h(t)\left(E_{Na} - V(t)\right) \tag{5.3a}$$

$$I_K(t) = g_K n(t)^4\left(E_K - V(t)\right) \tag{5.3b}$$

$$I_L(t) = g_L\left(E_L - V(t)\right) \tag{5.3c}$$

$$I_{CaL}(t) = g_{CaL} s_0(V(t))\left(E_{Ca} - V(t)\right) \tag{5.3d}$$

$$I_{CaT}(t) = g_{CaT}\left[a_{T0}(V(t))\, b_{T0}(r_T(V(t)))\right]^3\left(E_{Ca} - V(t)\right) \tag{5.3e}$$

$$I_{K/Ca}(t) = g_{K/Ca}\frac{Z_{\rm Ca}(t)^4}{Z_{\rm Ca}(t)^4 + \kappa_{K/Ca}^4}\left(E_K - V(t)\right) \tag{5.3f}$$

$$I_h(t) = g_h\left[0.3\, r_f(t) + 0.7\, r_s(t)\right]\left(E_h - V(t)\right). \tag{5.3g}$$

The gating variables $X(t) = \{h(t), n(t), r_T(t), r_f(t), r_s(t)\}$ obey first-order kinetics according to the equations

$$\frac{dX(t)}{dt} = \frac{X_0(V(t)) - X(t)}{\tau_X(V(t))}, \tag{5.4}$$

where the voltage dependence of the X_0 and the τ_X's are given in Table 5.2. The other four gating variables $\{m, s, a_T, b_T\}$ are assumed to have fast time constants, so they do not have their own dynamics and in Eqs.(5.3) they are set to $X_0(V)$.

As it stands so far, $Z_{\rm Ca}(t)$ only plays the role of an external time dependent forcing of the neuron through the Hill function kinetics in $I_{K/Ca}(t)$, as we have not yet introduced a dynamical equation for $Z_{\rm Ca}(t)$ itself. Note that $Z_{\rm Ca}(t)$ would have also entered into the equations for $I_{\rm CaL}(t)$ and $I_{\rm CaT}(t)$ had we used the GHK form for the voltage/current relation. We did examine the effects of the change from the ohmic current dependence in $I_{CaL}(t)$ and $I_{CaT}(t)$ we use here to the full GHK form of these

currents arising because of the substantial difference in Ca concentration within and without the cell. These was no visible difference in the results we present relevant to our main question of the ability to estimate the states and parameters of the model neuron using voltage measurements alone. We retain the ohmic form here recognizing that use of the GHK formulation may be important elsewhere.

The model we used in [4] for describing the experimental data on stimulus/voltage response experiments we have conducted on HVC neurons comprised these currents, absent the $I_{K/Ca}(t)$ current, along with additional Na and K currents. The manner in which some voltage dependent conductances, especially $I_h(t)$, was represented there is different in that model, and the Ca currents used GHK [51] voltage current relations reflective of the 10,000:1 ratio of extracellular to intracellular Ca concentrations. As we are concentrating here on the role of the added slow Ca dynamics to be described in a moment, we adopted a subset of the full model used earlier.

When we utilize the lessons from the "twin experiment" analysis of this V+Ca model and select models with which to analyze the observed laboratory data, we will examine several variants, all with the same core issue as explored here, but with somewhat different realizations of the Calcium dynamics. To proceed we select one version of Calcium dynamics: the model of Houart, et al [47].

5.3.2 Coupling Ca^{2+} into voltage dynamics

We introduce a calcium-dependent potassium current $I_{K/Ca}$ as the first ingredient in coupling the voltage and calcium dynamics. The conductance of this channel

Table 5.2: Contributions to the kinetics of the voltage gated channels. $\sigma(x,y,z) \equiv 0.5\left[1+\tanh\frac{x-y}{2z}\right], \quad E(x,y,z) \equiv \exp\left[\frac{x-y}{z}\right]$.

$I_{channel}(t)$ (pA)	X(t)	$X_0(V)$	$\tau_X(V)(ms)$
$I_{Na}(t)$	m(t)	$\sigma(V(t),-35,-5)$	—
$I_{Na}(t)$	h(t)	$\sigma(V(t),-37.4,4.3)$	1
$I_K(t)$	n(t)	$\sigma(V(t),-30,-5)$	$10\left[\cosh\left(\frac{V(t)+30}{10}\right)\right]^{-1}$
$I_{CaT}(t)$	$a_T(V(t))$	$\sigma(V(t),-65,-7.8)$	—
$I_{CaT}(t)$	$r_T(t)$	$\sigma(V(t),-67,-2)$	$200+87.5\ \sigma(V(t),68,2.2)$
$I_{CaT}(t)$	$b_T(r_T(t))$	$\sigma(r_T(t),0.4,-0.1)-\sigma(0,0.4,-0.1)$	—
$I_{CaL}(t)$	s(t)	$\sigma(V(t),-40,-5)$	—
$I_h(t)$	$r_f(t)$	$\sigma(V(t),-105,5)$	$100\left[\frac{-7.4[V(t)+70]}{E(V(t),70,-0.8)-1}+65E(V(t),56,23)\right]^{-1}$
$I_h(t)$	$r_s(t)$	$\sigma(V(t),-105,25)$	1500

depends on $Z_{\mathrm{Ca}}(t)$ through a Hill function, which has the generic form

$$\mathbf{H}(x, \kappa_x, n) = \frac{x^n}{x^n + \kappa_x^n} \tag{5.5}$$

where the Hill coefficient n is a positive integer. This type of $I_{K/Ca}$ is also referred to as an SK current, distinct from a so-called BK current, whose conductance has dependence on both voltage and intracellular calcium [84]. This gives rise to a K/Ca current of the form

$$\begin{aligned} I_{K/Ca}(t) &= g_{K/Ca} \frac{Z_{\mathrm{Ca}}(t)^4}{Z_{\mathrm{Ca}}(t)^4 + \kappa_{K/Ca}^4} (E_K - V(t)) \\ &= g_{K/Ca} \mathbf{H}(Z_{\mathrm{Ca}}(t), \kappa_{K/Ca}, 4)(E_K - V(t)). \end{aligned} \tag{5.6}$$

The choice $n = 4$ gives it a high sensitivity to $Z_{\mathrm{Ca}}(t)$ and is widely used in similar analyses [67, 31]. Additionally, we selected $\kappa_{K/Ca} = 0.35\,\mu$M since intracellular calcium levels are normally about 0.1 μM. This means only a slight rise in internal calcium levels are required to activate $I_{K/Ca}$, which suppresses spiking behavior and "turns off" bursts. Since calcium levels are generally about 10^4 times larger in the extracellular medium than in the cytoplasm, this yields a relatively small influx of Ca^{2+} ions.

With an appropriate stimulus, a model containing only I_{Na} and I_K would produce a continuously repeating spike train. The introduction of $I_{K/Ca}$ means the neuron model intermittently activates another K current which drives the voltage response of the neuron towards $E_K = -90\,mV$ and turns off spiking behavior of the neuron. Combined with the hyperpolarization-activated I_h and I_{CaL}, the neuron model's subthreshold behavior is greatly enriched, as will be seen in the simulations to follow.

5.3.3 Modeling intracellular Ca^{2+} uptake and release

To fully model the complexity of calcium dynamics, we are looking for a model that exhibits various complex behaviors. Dupont and Goldbeter proposed a model to study complex Ca^{2+} oscillations [26], later studied in detail by [14, 47]. They demonstrated that the model shows complex oscillatory phenomena such as limit cycle oscillations, bursting, quasiperiodic oscillations, and deterministic chaos [47]. In many cell types, the uptake and release of Ca^{2+} ions by the ER is mediated predominantly by IP_3 [93]. This uptake and release via IP_3 mediation has been suggested as a source of intracellular Ca^{2+} oscillations. The complex $Z_{\mathrm{Ca}}(t)$ oscillations arising in this model are due to the release of Ca^{2+} from internal stores, with dynamics based mainly on mechanisms of Ca^{2+}-induced Ca^{2+} release (CICR) that take into account the Ca^{2+}-stimulated degradation of IP_3 by a 3-kinase [47].

CICR was originally found to occur in muscle and cardiac cells, and was later found in a variety of other cells including neurons [19]. Solovyova et al. have observed CICR in cultured rat dorsal root ganglia neurons [88]. However, they also observed CICR triggered by Ca^{2+} entry through voltage-gated Ca^{2+} channels. The sum of the two voltage-gated calcium currents I_{CaT} and I_{CaL} may therefore act as a calcium-release stimulus. We thus used the intracellular calcium dynamics model of Houart et al. [47], but with the replacement $\beta_{input} \to \beta_{input}\left(I_L(t) + I_T(t)\right)$ for the (previously constant) external stimulus. This change now acts to couple the calcium dynamics to the membrane voltage.

There are three dynamical quantities of interest in the intracellular medium. The first is the cytosolic calcium concentration $Z_{\mathrm{Ca}}(t)$, the second is the calcium concentration $Z_{\mathrm{ER}}(t)$ in the endoplasmic reticulum, and the third is the concentration

Table 5.3: All the parameters values entering the dynamics of the cytosolic calcium concentration $Z_{\rm Ca}(t)$, the calcium concentration, and the endoplasmic reticulum $Z_{\rm ER}(t)$.

Parameter	Value	Parameter	Value
ν_o (μM s^{-1})	3.22	$\gamma_{\rm pump}$ (μM s^{-1})	6
$\beta_{\rm input}$ (μM s^{-1}pA^{-1})	2×10^{-5}	$\kappa_{\rm p}$ (μM)	0.1
$\gamma_{\rm leak}$(s^{-1})	10	$\nu_{\rm synthesis}$ (μM s^{-1})	1.83
$\gamma_{\rm CICR}$ (μM s^{-1})	30	α_0 (unitless)	1.5
$\kappa_{\rm cCa}$ (μM)	0.6	$\gamma_{\rm IP3leak}$ (s^{-1})	13
$\kappa_{\rm cER}$ (μM)	0.3	$\gamma_{\rm degradation}$ (μM s^{-1})	50
$\kappa_{\rm cIP3}$ (μM)	0.1	$\kappa_{\rm dIP3}$ (μM)	0.3194
$\gamma_{\rm ERleak}$ (s^{-1})	1	$\kappa_{\rm pCa}$ (μM)	1

of catalytic IP_3, $Z_{\rm IP3}(t)$. Following [47] we adopt

$$\begin{aligned}
\frac{dZ_{\rm Ca}(t)}{dt} &= \nu_0 + \beta_{\rm input}(I_{\rm CaL}(t) + I_{\rm CaT}(t)) - \gamma_{\rm leak} Z_{\rm Ca}(t) \\
&\quad + \gamma_{\rm CICR}\ \mathbf{H}(Z_{\rm Ca}(t), \kappa_{\rm cCa}, 2)\ \mathbf{H}(Ca_{\rm ER}(t), \kappa_{\rm cER}, 2) \\
&\quad + \gamma_{\rm ERleak} Z_{\rm ER}(t) - \gamma_{\rm pump}\ \mathbf{H}(Z_{\rm Ca}(t), \kappa_p, 2), \\
\frac{dZ_{\rm ER}(t)}{dt} &= -\gamma_{\rm CICR}\ \mathbf{H}(Z_{\rm Ca}(t), \kappa_{\rm cCa}, 2)\ \mathbf{H}(Z_{\rm ER}(t), \kappa_{\rm cER}, 2) \\
&\quad - \gamma_{\rm ERleak} Z_{\rm ER}(t) + \gamma_{\rm pump}\ \mathbf{H}(Z_{\rm Ca}(t), \kappa_p, 2), \\
\frac{dZ_{\rm IP3}(t)}{dt} &= \nu_{\rm synthesis} + \alpha_0 \beta_{\rm input}(I_L(t) + I_T(t)) - \gamma_{\rm IP3leak} Z_{\rm IP3}(t) \\
&\quad - \gamma_{\rm degradation}\ \mathbf{H}(Z_{\rm IP3}(t), \kappa_{\rm dIP3}, 1)\ \mathbf{H}(Z_{\rm Ca}(t), \kappa_{\rm pCa}, 4),
\end{aligned}$$

where the form of $\mathbf{H}(x, \kappa_x, n)$ is given in Eq.(5.5).

In this model we assume free calcium ions are uniformly distributed across the cytosol. This avoids the additional complication of introducing a partial differential equation to model the spatial dependence of the $Z_{\rm Ca}(t)$ dynamics. This simplification is consistent with experimental results [88].

5.4 Data generation for twin experiments

As with previous V+Ca models [31, 76], we focus our attention in this paper on chaotic oscillations. This requires the parameterization of the calcium dynamics to lie within a particular parameter range specified in [47]. We adopted the intracellular calcium model of [9, 47] with the exception that all time constants were reduced by a factor of 60, so that the calcium oscillations would occur on the scale of seconds rather than minutes. Since the time between voltage spikes is on the order of some milliseconds, this rescaling makes the voltage behavior switch between a resting state with no spiking and a firing state with repeated production of action potentials lasting about 10 spikes. There is no loss in generality or information here, since the length of a burst is still much longer than that of single spike. This also allows the model to undergo several bursts within a reasonable timescale for numerical integration. Finally, the scaling coefficient (β_{input} in the model) of the calcium currents was adjusted to limit the stimulus to the chaotic regime according to the stability diagram in [47].

The model was integrated with an adaptive Runge-Kutta 4^{th} order scheme (see [79], Section 16.2), using a maximum integration time step $\Delta t = 0.02\,\text{ms}$ and an error tolerance $\epsilon = 10^{-8}$. Fig. 5.1 shows the resulting time course of the membrane voltage $V(t)$ and cytosolic Ca^{2+} concentration $Z_{\text{Ca}}(t)$ with a constant applied stimulating current $I_{app} = 600$ pA. The main characteristic in the behavior of the model neuron is the production of action potential bursts with varying durations controlled by the variation of $Z_{\text{Ca}}(t)$.

As in the other models of voltage plus calcium dynamics, we find irregular bursting of the action potential. The model neuron enters into a bursting period of action potentials when $Z_{\text{Ca}}(t)$ decreases. Low $Z_{\text{Ca}}(t)$ cuts off the current $I_{K/Ca}(t)$

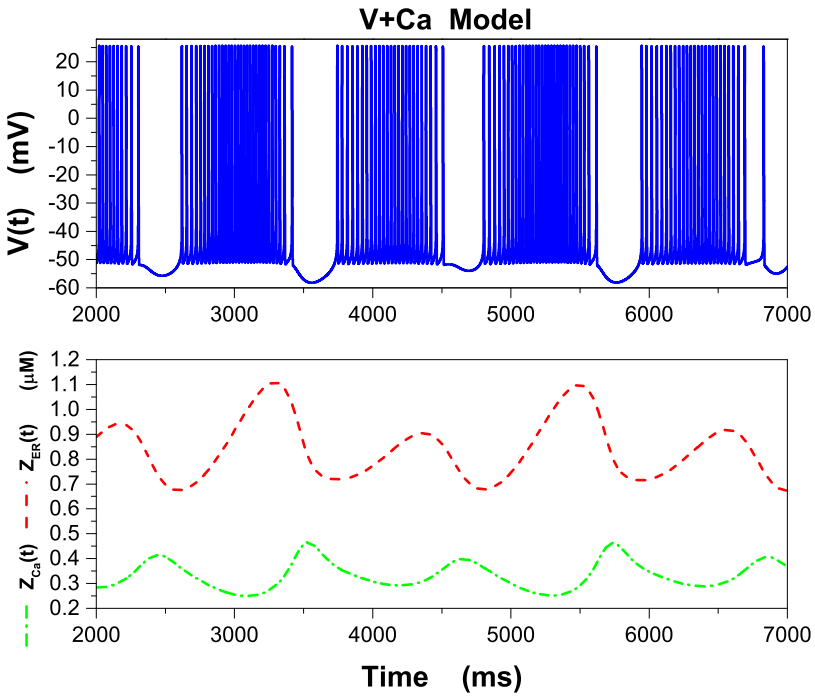

Figure 5.1: Time series for the V+Ca model. **Top Panel** Time course of membrane voltage **Bottom Panel** Time course of the cystolic calcium concentration $Z_{\mathrm{Ca}}(t)$ and time course of the calcium concentration in the ER $Z_{\mathrm{ER}}(t)$.

allowing the neuron to depolarize countering the tendency of $I_{K/Ca}(t)$ to drive the neuron to $E_K \approx -90\,\mathrm{mV}$ which involves deep hyperpolarization. Within a burst the spiking frequency is sensitive to the change of $Z_{\mathrm{Ca}}(t)$ through the current $I_{K/Ca}(t)$ This is called spike frequency accommodation; however, it is not visible in Fig. 5.1. During a burst, calcium ions flood into the cytosol through voltage-gated calcium channels. The free calcium ions in the cytosol induce release of Ca^{2+} from the ER, an internal calcium store, by CICR. This, along with the membrane calcium current, greatly increases $Z_{\mathrm{Ca}}(t)$ during a burst. As $Z_{\mathrm{Ca}}(t)$ rises, $I_{K/Ca}(t)$ activates, driving the neuron back to a hyperpolarized state. This competition between polarizing and depolarizing processes, results in chaotic oscillations in the parameter range we have selected.

To determine whether the irregular behavior visible in Fig.5.1 is chaotic, we evaluated the largest Lyapunov exponent for the observed time series, and we display that in Fig.5.2. We see that for β_{input}, which sets the scale of the driving force from voltage gated calcium channels into the ER dynamics, just above zero to about 4×10^{-5} μM/(ms pA), a positive Lyapunov exponent appears. This is similar to the behavior of the other two Voltage plus Ca dynamical models we analyzed [31, 76].

Lyapunov exponents are obtained by a recursive QR decomposition of the Oseledec matrix [1]. For each β_{input}, a trajectory of $T = 600\,\text{sec}$ is generated. To avoid overflow the Jacobian matrices, $DF(\mathbf{x}, n\Delta t) \quad n = 1, 2, \ldots$ are calculated by integrating the variational equations with $DF(\mathbf{x}, (n-1)\Delta t)$ equal to the identity matrix. If Δt is too small, of the order of the spiking period ($1ms$), the Lyapunov exponents exhibit spiking behavior. They can be made smoother by averaging over the time of voltage bursts, namely over $\Delta t = 1500ms$.

5.5 Estimation of Model States and Parameters from Sparse Data

The central question we address in this paper has to do with the number of measurements required to permit accurate estimation of the fixed parameters and unobserved state variables of the model using time series of the data. There is a direct connection between the ability to synchronize the data with the model output and the ability to accurately estimate unknown parameters and unobserved state variables in the model [4]. If this synchronization is absent, then the surface over which one searches in the estimation procedure has many local minima. When synchronization

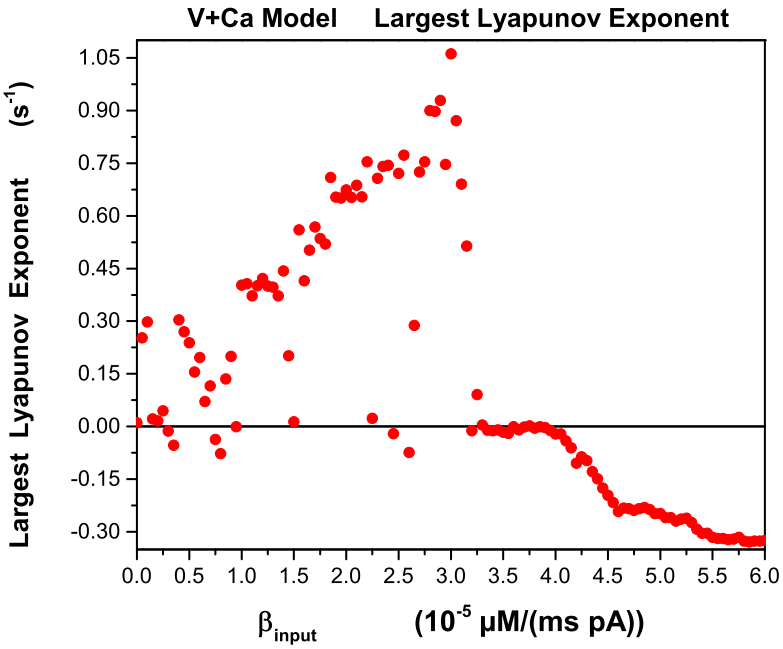

Figure 5.2: The largest Lyapunov exponent of the full V+Ca model as a function of the parameter β_{input}.

occurs, this surface becomes quite smooth and estimation proceeds with accuracy. This connection is present here as well, as we shall show.

We introduce information about the data into the model dynamics through terms in the dynamical equation for state variables $x_l(t)$ which are observed. These have the form $g_{x_l}\left(y_l(t) - x_l(t)\right); l = 1, 2, ..., L$, where the g_{x_l} are constant, nonzero, and positive coupling strengths for the measured components. We will consider two kinds of data within the V+Ca models: measurements of the membrane voltage $V(t)$ $y_1(t) = V_{data}(t)$ and measurements of the cytosolic calcium concentration $Z_{\mathrm{Ca}}(t)$

$y_2(t) = Ca_{data}(t)$. Accordingly, we modify the equations in the model to be

$$\frac{dV(t)}{dt} = F_V\left(V(t), Z_{\mathrm{Ca}}(t), \mathbf{Y}(t)\right) + g_V\left(y_1(t) - V(t)\right) \tag{5.7a}$$

$$\frac{dZ_{\mathrm{Ca}}(t)}{dt} = F_{Ca}\left(V(t), Z_{\mathrm{Ca}}(t), \mathbf{Y}(t)\right) + g_{Ca}\left(y_2(t) - Z_{\mathrm{Ca}}(t)\right), \tag{5.7b}$$

where $F_V(V(t), Z_{\mathrm{Ca}}(t), \mathbf{Y}(t))$ and $F_{Ca}(V(t), Z_{\mathrm{Ca}}(t), \mathbf{Y}(t))$ are the vector fields in the model as outlined in Section 5.3. The dependence on $V(t)$ and $Ca(t)$ is written explicitly for clarity. The rest of the dynamical variables, whose equations remain unchanged, are collected in $\mathbf{Y}(t)$.

Our overall goal is to provide a method for accurately estimating the parameters and state variables of a model of neuron dynamics when we observe a sparse subset of the state variables in the model. The general formulation of this is given in [4], and here we use the variational approach which minimizes a cost function comparing the model output with the data that has been collected, subject to the dynamical equations of the model. The model thus acts as a nonlinear filter which pass information in the data along to the unobserved state variables and constrains the parameters.

In the present case we have a model with many gating variables, fixed parameters, and dynamical variables, and we seek to determine all of these parameters and all states from observations over a temporal window $[0, T]$. If we have good estimations of the parameters from data in this window and we have accurate estimations of the states at $t = T$, we can use the model differential equations to predict observed behavior for $t > T$. We consider here the possibility of measuring two state variables, the membrane voltage $V(t)$ and the intracellular calcium concentration $Z_{\mathrm{Ca}}(t)$. From one or both of these time series observations during $[0, T]$, we want to estimate the

fixed parameters and all the gating variables and other state variables at T.

5.5.1 Estimation with V(t) Measurements Only

We first ask if we can estimate parameters and states of the V plus Ca system with voltage measurements alone using a coupling term as in Eq.(5.7)? To answer this we look at the synchronization error as a function of the selected value of the initial condition of calcium concentration for the model $Z_{\mathrm{Ca}}(t=0)_{\mathrm{Model}}$ minus the known initial value for the calcium concentration in the data $Z_{\mathrm{Ca}}(t=0)_{\mathrm{Data}}$ when all other parameters fixed to their known values in generating the data, this should have a zero when $Z_{\mathrm{Ca}}(t=0)_{\mathrm{Model}}$ is chosen correctly. We evaluate this synchronization error, or cost function, with g_V fixed at $g_V = 2.0\, ms^{-1}$,

$$\mathbf{C}_{\mathrm{SE}}\left(Z_{\mathrm{Ca}}(t=0)_{\mathrm{Model}} - Z_{\mathrm{Ca}}(t=0)_{\mathrm{Data}}, g_{Ca}\right) = \frac{2}{N}\sum_{n=N/2}^{N}\left(y_1(t_n) - V(t_n)\right)^2, \qquad (5.8)$$

and for various values of g_{Ca}. This is displayed in Fig.5.3. $V(t_n)$ is a time series of the model output obtained by forward integration of Eq.5.7 starting with a different initial condition than that used to generate the data. The first half of the integration data points containing the initial transient behavior is discarded. The synchronization error is evaluated using the model output voltage only, even though in this twin experiment we know all of the other state variables because we generated the data. In a laboratory experiment, we would only know the observed and model output voltages, and, perhaps, the observed and model output $Z_{\mathrm{Ca}}(t)$.

In generating the model trajectories, we fixed all parameters and initial state values at the values used in generating data except for the initial condition

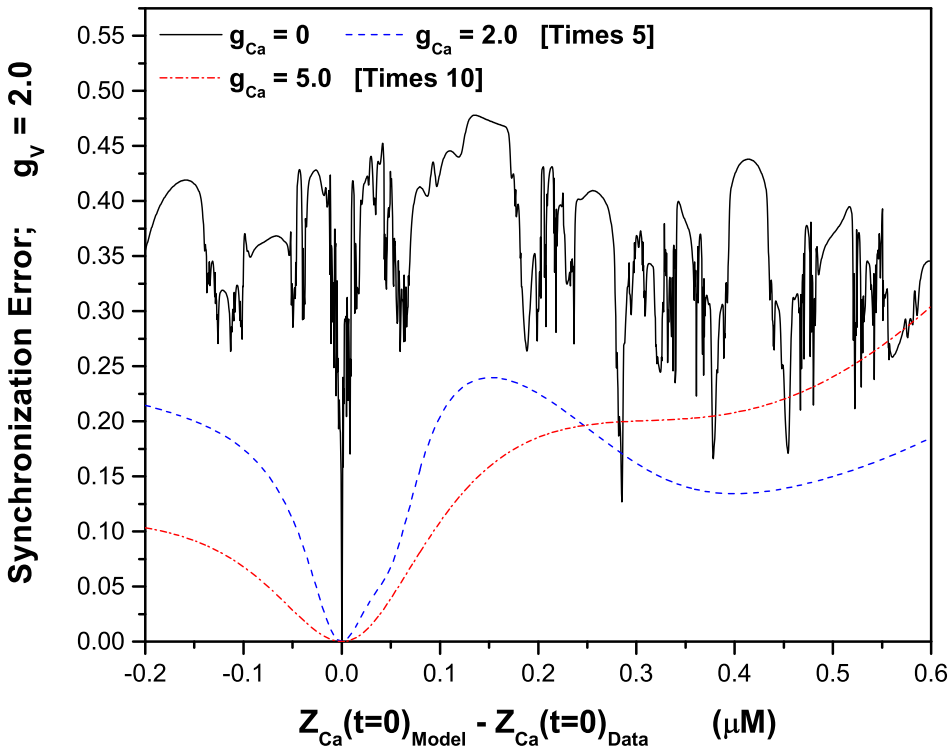

Figure 5.3: Synchronization error as a function of the deviation of the $Z_{\mathrm{Ca}}(t)$ initial condition from its actual value: $Z_{\mathrm{Ca}}(t=0)_{\mathrm{Model}} - Z_{\mathrm{Ca}}(t=0)_{\mathrm{Data}}$. The coupling of voltage data is fixed at $g_V = 2.0\,\mathrm{s}^{-1}$, and a selection of values of g_{Ca} are listed in the legends: when $g_{Ca} = 0\,\mathrm{ms}^{-1}$, the synchronization manifold exhibits surface with many local minima. This is the case when no measurements of $Z_{\mathrm{Ca}}(t)_{\mathrm{Data}}$ are made. Next we select positive values of g_{Ca}: $g_{Ca} = 2.0\,\mathrm{ms}^{-1}$ (the synchronization error is shown multiplied by a factor of 5) and $g_{Ca} = 5.0\,\mathrm{ms}^{-1}$ (the synchronization error is shown multiplied by a factor of 10) display the smoothing process of the synchronization manifold as the g_{Ca} coupling strength increases.

$Z_{\mathrm{Ca}}(t = 0)$. To estimate this initial condition, we should minimize the synchronization error over a range of values of $Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}}$. In Fig.5.3 we display $\mathbf{C}_{\mathrm{SE}}\left(Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}} - Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Data}}, g_{Ca}\right)$ versus the difference $Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}} - Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Data}}$ for various values of the calcium coupling g_{Ca} and a fixed value of the voltage coupling g_V. To estimate $Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}}$, we could search over the surface shown by some form of iterative procedure. However, with $g_{Ca} = 0\ s^{-1}$, which means no calcium data are presented, the surface is peppered by local minima, and local optimization procedures will fail to find the correct $Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}}$, in general.

With $g_{Ca} = 0$, then, namely no information about $Z_{\mathrm{Ca}}(t)_{\mathrm{Data}}$ passed to the model, the surface over which we must search for $Z_{\mathrm{Ca}}(t = 0)_{\mathrm{Model}}$ has multiple local minima, and this impedes accurate estimation.

5.5.2 Estimation with V(t) and $Z_{\mathrm{Ca}}(t)$ Measurements

We next increased g_{Ca} to 2.0 s^{-1} and again computed the synchronization error. Note that in the figure, this synchronization error is multiplied by a factor of 5 so that it is clearly seen on the same scale as the surface for $g_{Ca} = 0\ s^{-1}$. While the surface is smoother, there remains a visible local minimum away from zero, and again local optimization procedures are not suitable, especially if there are more local minima not shown in the plot.

The final curve in Fig.5.3 was computed with $g_{Ca} = 5.0\ s^{-1}$. The search surface is smoothed out enough so that essentially any search algorithm will result in the correct answer. This display has the synchronization error multiplied by a factor of 10 so that it is clearly seen on the same scale as the surface for $g_{Ca} = 0\ s^{-1}$. Thus, increasing g_{Ca} with g_V fixed and large enough results in the synchronization of the

data and the model output, both of which are chaotic, as well as smoothing the surface over which a search must proceed.

We can examine the effect of varying both g_V and g_{Ca} using the joint synchronization error in $V(t)$ and $Z_{\mathrm{Ca}}(t)$ measurements:

$$\mathbf{C}_{\mathrm{SE}}\left(g_V, g_{Ca}\right)_{\mathrm{V\&Ca}} = \frac{2}{N}\sum_{n=N/2}^{N}\left[\frac{\left(V(t_n)_{\mathrm{data}} - V(t_n)\right)^2}{S_V^2} + \frac{\left(Z_{\mathrm{Ca}}(t_n)_{\mathrm{data}} - Z_{\mathrm{Ca}}(t_n)\right)^2}{S_{Ca}^2}\right]. \tag{5.9}$$

The constants S_V and S_{Ca} are selected to scale the $V(t)$ and $Z_{\mathrm{Ca}}(t)$ dynamical ranges to be nearly equal. For this purpose we set $S_k = |\max\, x_k - \min\, x_k|$ for the k^{th} contribution to SE, where the minimum and the maximum are taken over the observation window. The result is shown in Fig.5.4 with $S_V = 82.5\,\mathrm{mV}$ and $S_{Ca} = 0.236\,\mu\mathrm{M}$.

This figure demonstrate that when either g_V or g_{Ca} is zero, the synchronization error does not go to zero leading to inaccurate estimations of the parameters and state values determining the model output.

5.6 State and Parameter Estimation; Prediction after Estimation

So far we have established that in coupling measurements of $V(t)$ and $Z_{\mathrm{Ca}}(t)$ into the overall cellular dynamics via Eq.(5.7), we are able to transmit enough information to the model to smooth out the search surface of the synchronization error. The couplings g_V and g_{Ca} are not part of the biophysics of the model, however. After they are used to synchronize the model output and the data, they should be disposed of when predicting data beyond the observation window.

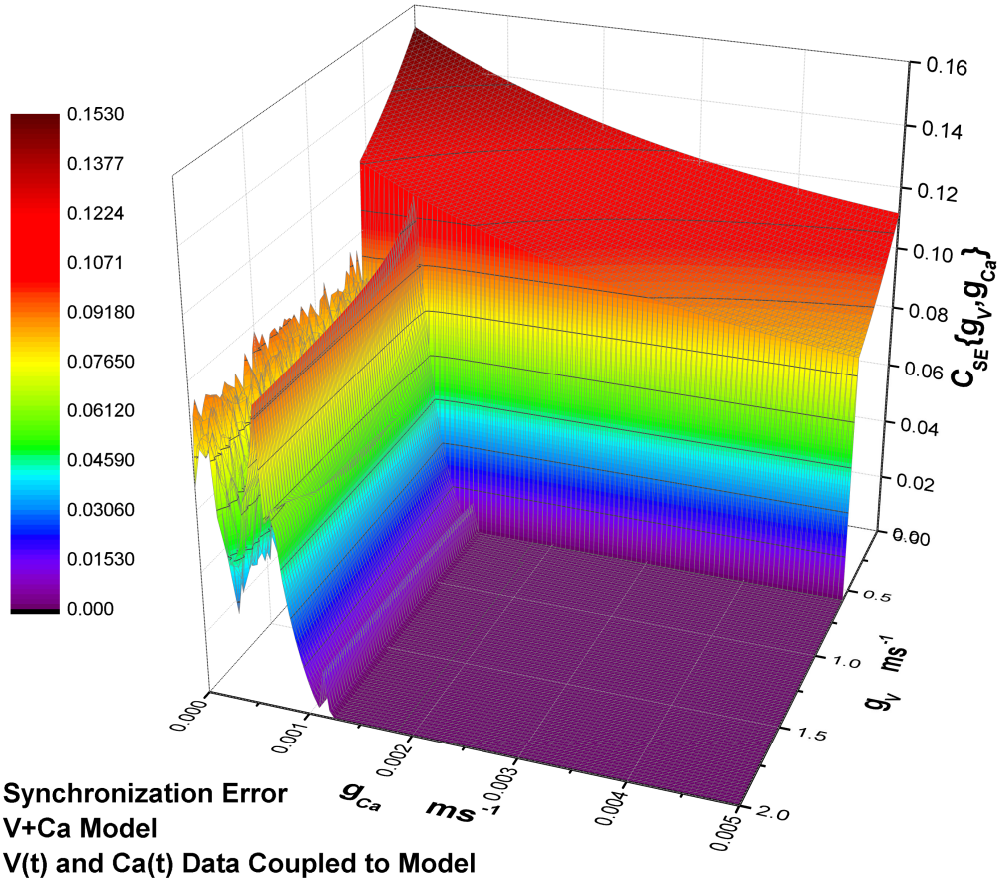

Figure 5.4: Synchronization error as a function of two coupling strengths g_{Ca} and g_V: $\mathbf{C}_{\mathrm{SE}}$ (Eq.(5.9)) remains at nonzero values when either g_{Ca} or g_V is equal to zero. This indicates that when only $V(t)$ or only $Z_{\mathrm{Ca}}(t)$ are observed, it does not allow synchronization of the model output with the data. However, this Figure shows that the synchronization error does go to zero when g_{Ca} and g_V are both large enough. This indicates that two measurements will allow accurate estimation of the parameters and unobserved states in the V+Ca model. At the end of the observation window, $t = T$, we are then able to use the model with estimated parameters and initial conditions at $t = T$ to predict the response of the neuron for $t > T$. This last step is the validation criterion for the model itself.

To allow these couplings to regulate the instabilities on the synchronization manifold, and then send them to zero when the estimation procedure is completed, we now explicitly make the coupling strengths g_V and g_{Ca} functions of time and treat $g_V(t_n)$ and $g_{Ca}(t_n)$ as additional parameters to be estimated according to our estimation protocol. In principle this means they will be small at times along the orbit when the synchronization manifold is nearly stable and larger in unstable regions. More precisely, we introduce the coupling strengths as additional penalties into the cost function Eq.(5.9) as

$$\mathbf{C}_{\mathrm{SE}} = \frac{2}{N}\sum_{N/2}^{N}\left[\frac{(V(t_n)_{\mathrm{data}} - V(t_n))^2}{S_V^2} + \frac{(Z_{\mathrm{Ca}}(t_n)_{\mathrm{data}} - Z_{\mathrm{Ca}}(t_n))^2}{S_{Ca}^2} + g_V(t_n)^2 + g_{Ca}(t_n)^2\right], \tag{5.10}$$

which is to be minimized subject to Eq.(5.7). The minimization was done using the publicly available nonlinear optimization software IPOPT [95, 94, 60]. This estimates all the model parameters, all the model state variables at time within the observation window, and estimates $g_{Ca}(t_n)$ and $g_V(t_n)$.

5.6.1 Prediction with only $V(t)$ Measurements

First, we performed the state and parameter estimation for the full model but with **voltage measurements alone**. This was done to verify the conclusion made above that it is not feasible to estimate unobserved states with voltage data only. We performed the optimization with 80,000 voltage data points spaced in model time by $\Delta t = 0.02\,\mathrm{ms}$, with all parameters fixed and an external stimulus $I_{\mathrm{app}} = 600\,\mathrm{pA}$. The estimated states were then used to continue integrating the model forward with the same I_{app} but with $g_V = 0ms^{-1}$, $g_{Ca} = 0ms^{-1}$.

Since, when only $V(t)$ is observed, the SE surface should be riddled with local minima, any "optimal" solution found by the estimation should make the model predict future solutions poorly. Indeed, this was the case, and an example inaccurate prediction is shown in Fig.5.5.

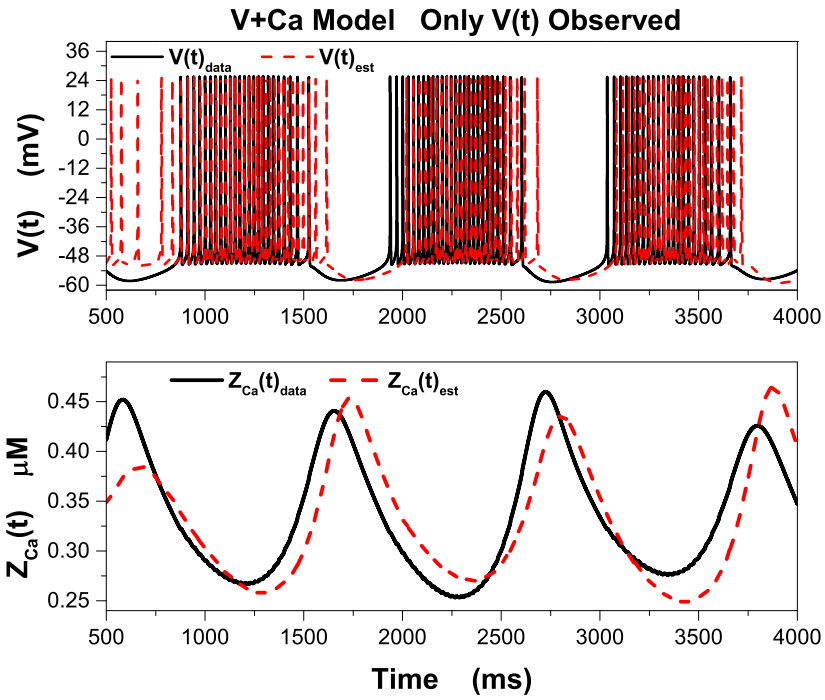

Figure 5.5: Results from dynamical state estimation for the V+Ca model with only voltage data presented ($g_{Ca} = 0\ s^{-1}$). 80,000 voltage data points are used for estimation; this is an assimilation window of 1600ms. All parameters model are fixed, and all the states are to be estimated. The result shown is the integration based on using the estimated state variables at $t = 0\ ms$ as initial conditions then integrating the dynamical equations with $g_V = 0\ ms^{-1}$ using the estimated initial conditions. The disagreement between estimate and data shows that the information from voltage observations alone is not enough to accurately estimate all the unobserved states. The known data are in black. Estimates are shown in red.

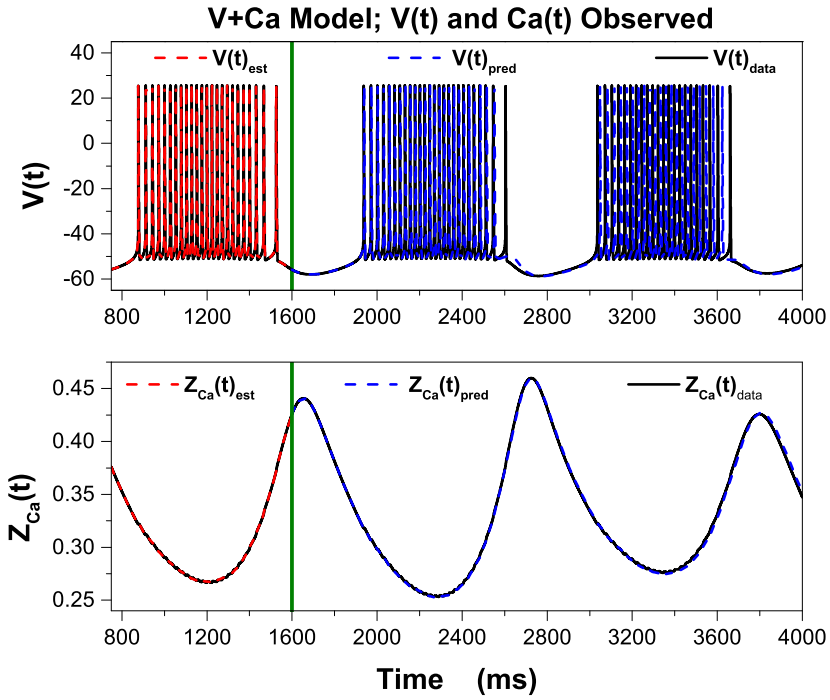

Figure 5.6: Results from dynamical state estimation for the full model with both $V(t)$ and $Z_{\mathrm{Ca}}(t)$ presented to the model. 80,000 voltage data points and 80,000 $Z_{\mathrm{Ca}}(t)$ data points are used for estimation; this is an assimilation window of 1600ms. All the states except $V(t)$ and $Z_{\mathrm{Ca}}(t)$ are to be estimated. In this case we also estimated the parameters listed in Table 5.4. Other parameters in the model are fixed. Prediction is made by integrating the model forward using the estimated parameters and estimated state variables at T = 1600ms as initial conditions. A vertical line indicates where estimations terminate and prediction begins. The known data are in black. Estimates are in red, and predictions are shown in blue.

Table 5.4: Estimated parameters and comparison with their true values. Both $V(t)$ and $Z_{\mathrm{Ca}}(t)$ data were presented to the model neuron.

Parameter	Value in Data	Estimate
g_{Na} (nS)	450	450.02610
g_K (nS)	50	50.00015
g_L (nS)	2	2.00124
g_{CaT} (nS)	3	2.40044
g_{CaL} (nS)	10	10.00011
g_h (nS)	4	3.37253
g_{CaK} (nS)	2	2.00110
ν_0 (μM s^{-1})	3.22	3.22316
β_{input} (μM s^{-1}pA^{-1})	2×10^{-5}	1.99973×10^{-5}

5.6.2 Prediction with both $V(t)$ and $Z_{\mathbf{Ca}}(t)$ Measurements

When we presented **both voltage and calcium measurements** to the model, the estimation using IPOPT returned accurate state and parameter values which gave the model strong predictive behavior. The resulting estimates are listed in Table 5.4. Note that all of the conductances were very accurately estimated except g_{CaT} and g_h. This is probably because I_{CaT} and I_{h} are triggered by large hyperpolarizing currents [22], but we stimulated the neuron with a contant depolarizing current. In our model, however, $I_{K/Ca}$ can hyperpolarize the neuron below threshold and thus activate I_{CaT} and I_{h} to some extent, so inaccurate estimates for these two maximal conductances appear not to effect the value of SE very much. If the neuron were driven by a large hyperpolarized step current, the estimates of these quantities might be improved significantly.

5.7 Discussion and Summary

We have shown that the use of a neuron model which couples membrane voltage dynamics to the relatively slow dynamics of intracellular calcium uptake and release requires more than a measurement of the membrane voltage trace alone to estimate model parameters and initial conditions when the parameter values give the model chaotic behavior. Specifically, we have shown that simultaneous measurements of voltage $V(t)$ and cytosolic calcium concentration $Z_{\mathrm{Ca}}(t)$ are sufficient to regularize this system and estimate the maximal conductances of the membrane currents.

The calcium components in our V+Ca model are intrinsically chaotic. Complex forms of calcium oscillations are typically observed in biological phenomena, oscillations express periodic as well as nonperiodic behavior. [14, 98, 38, 8, 20, 40, 75, 29, 87, 85, 61] The erratic behavior may often be thought as to be the consequence of stochastic effects, but it can also be generated by a deterministic system operating in the chaotic regime. The model we used here that was found exhibit both aperodic oscillation and bursting behaviors in its chaotic regime. (Fig.7 in [25]) Similar calcium bursting behavior characterized by irregular number of secondary spikes and an irregular spacing between main spike and aperodic oscillation with varying amplitudes are observed in hepatocytes under different stimuli. [23, 83, 41] Hazledine et al. found the largest Lyapunov exponent is positive for calcium oscillation data obtained from certain legume plants, which directly identifies that calcium oscillation is chaotic. [43, 61]

The non-periodic nature of calcium oscillations is related to the concentration of extracellular agonists such as hormones and neurotransmitters, thereby dictating that the external signal is encoded in terms of the temporal pattern of calcium oscillations, the so-called frequency encoding calcium signals. [98, 38] It is not yet

known to what extent this chaotic nature of oscillations is essential to the cell function (muscle/neurons/etc); however, it had been hypothesized that chaotic oscillations is likely a common feature of calcium signal transductions where chaos permits greater flexibility in the regulation of protein activity than either stochastic or stable systems, allowing differential responses in multifunctional signaling pathways. [8, 61]

Our focus here is on properties of coupled voltage and calcium dynamics as it relates to our ability to use observed time courses in individual cells, as well as in networks, to determine the biophysical cellular processes and the network connectivity in networks of neurons.

We used "twin experiments" in which model output trajectories with known initial conditions and parameter values were used as sources of experimental 'data' in the dynamical synchronization procedure. This is to be compared to a realistic experimental setting in which only a limited number of physical quantities can be measured and in general with a non-negligible level of noise [4]. The advantage of using twin experiments is that they provide a controlled setting in which we have exact knowledge of the *entire* state of the system. While the true metric of success is the estimation and accurate prediction of the measurable quantities, twin experiments give one the opportunity to investigate the efficacy of the data assimilation methods on the entire model space, and to possibly eliminate unnecessary or degenerate degrees of freedom.

When such a V+Ca model is applied to analyze experimental data, knowledge of the neuron's membrane potentials and the intracellular Ca^{2+} concentration is needed to interpret Ca^{2+} signalling. Loading calcium indicators into neurons has been widely used in recent years. Fluorescent Ca^{2+} indicators can be loaded into biological neurons

through the same microelectrodes used for recordings the intracellular voltage thereby allowing simultaneous electrophysiological recordings and ratiometric calcium imaging. In the early years, calcium indicators were delivered through sharp microelectrodes both in vitro [48] and in vivo [91]. In more recent years, calcium dyes are delivered through whole-cell patch-clamp micropipettes [64, 101, 28, 74] and the whole-cell recordings are generally performed under visual guidance using two-photon imaging [49, 57, 92].

When using the kind of model explored in this paper to analyze data from indivudal neurons [4] it may suffice to use the single compartment approach described here. However, we recognize that when neurons with voltage and Ca dynamics are to be used in biophysical realizations of networks, we will need to introduce spatial representations (compartments) for dendritic process where the Ca dynamics resides, and axonal processes to facilitate communication of neural activity within the network.

Chapter 5, in full, is a reprint of the material as it appears in Jingxin Ye, Paul J. Rozdeba, Uriel I. Morone, Arij Daou, Henry D. I. Abarbanel, Estimating the Biophysical Properties of Neurons with Intracellular Calcium Dynamics, *Physical Review E*, 89(6), 062714 (2014). The dissertation author was the primary investigator and author of this paper.

Appendix A

Annealing code `minAone` User Guide

A.1 About `minAone`

The annealing code `minAone` described in this document is used for calculating action levels of dynamical systems. The code is developed as an extention of `minAzero` written by Bryan Toth and Chris Knowlton. That is where the name `minAone` comes from. In another aspect, following the lowest action level A_0, A_1 represents the second lowest one, which is an interesting quantity we care about and has significant application in statistical data assimilation.

A.2 Problem Statement

Given a dynamical system modeled by D-dimensional discrete map

$$x_a(n+1) = f_a(\mathbf{x}(n)), \quad a = 1, \ldots, D$$

the probability distribution of its states can be expressed as $P(X|Y) = \exp(-A_0)$, when L-dimensional obersations Y are present. If one assumes both measurement noises and model error are independent and gaussian, the action A_0 has the format of

$$A_0(X) = \sum_{n=0}^{m} \frac{R_m(n)}{2} \sum_{l=1}^{L} [x_l(n) - y_l(n)]^2 + \frac{R_f}{2} \sum_{n=0}^{m-1} \sum_{a=1}^{D} [x_a(n+1) - f_a(\mathbf{x}(n))]^2. \quad \text{(A.1)}$$

where R_m and R_f are the inverse of variances.

The annealing method is based on the observation that the minima solution X^q of A_0 at $R_f = 0$ is $x_l(n) = y_l(n)$, the other $D - L$ components of the model state vector are undetermined, and the solution is degenerate. As we increase R_f, the action levels split, and depending on R_m, R_f, L and the precise form of the dynamical vector field $\mathbf{f}(\mathbf{x})$, there will be 1,2,... minima of A_0.

A.3 Annealing Procedure

The annealing process proceeds as follows: with very small initial R_f, we call it R_{f0}, solve the $(m+1)D$-dimensional search problem with an optimization algorithm that seeks minima of $A_0(X)$. Start the search with a set of trial paths whose components are selected from a uniform distribution within limits suggested by examining the times series generated by the model $\mathbf{x} \to \mathbf{f}(\mathbf{x})$ (or any other selection process for the initial guess). This will generate a collection of approximate paths X^q. Increase R_f by a small increment (we choose $R_f = \{R_{f0}\alpha^\beta\}$, where $\alpha = 2, \beta = 0, 1, \ldots$ in our examples), and using the paths found for the smaller R_f as initial guesses, find a new set of approximate X^q. Continue this process until the lowest action level path X^0 produces a $A_0(X^0)$ near expected value, which can be identified from our knowledge

of measurement noises. In our example, as the values $[y_l(n) - x_l(n)] \sim \mathcal{N}(0, \sigma^2)$ by our choice, the measurement error term $\sum_{n=0}^{m} \sum_{l=1}^{L} [(x_l(n) - y_l(n))/\sigma]^2/2$ has a χ^2 distribution with $L(m+1)$ degrees of freedom. The mean and uncertainty of this distribution over different choices of noise waveforms are $(m+1)L/2$ and $\sqrt{(m+1)L/2}$, respectively.

After identifying the global minima and other local minima of A_0, we can employ laplace method to approximate the expected value $\langle G(X) \rangle$ of a function $G(X)$ is

$$\langle G(X) \rangle = \frac{\int dX\, G(X) \exp[-A_0(X)]}{\int dX\, \exp[-A_0(X)]} \approx G(X^0). \tag{A.2}$$

plus exponentially small corrections. If the action level $A_0(X^0)$ is substantially less than the action level on the next path $A_0(X^0) \ll A_0(X^1)$, all statistical data assimilation expected values $\langle G(X) \rangle$ are given by X^0 and fluctuations about that path with exponential accuracy of order $\exp[-(A_0(X^1) - A_0(X^0))]$.

More details can be found in Ye, J., Kadakia, N., Rozdeba, P. J., Abarbanel, H. D. I., and Quinn, J. C.: Improved variational methods in statistical data assimilation, Nonlin. Processes Geophys., 22, 205-213, doi:10.5194/npg-22-205-2015, 2015

A.4 Installing Required Programs and Packages

This document will assume that the user is using a Linux distribution and has basic compliers installed including gcc, gfortran and python.

A.4.1 Python Packages

These python scripts link to the sympy library. To install these, use `apt-get/yum install sympy` or download directly from sympy.org.

A.4.2 IPOPT

Download

Get it here: `https://projects.coin-or.org/Ipopt`

- Download and unzip latest version of IPOPT
- As of right now this is 3.11.7 - Efficacy of installation instructions may degrade over time as packages are updated.
- Go into ThirdParty folder in the IPOPT directory then do the following commands.

```
$ cd Blas
$ ./get.Blas
$ cd ../Lapack
$ ./get.Lapack
$ cd ../ASL
$ ./get.ASL
$ cd ../Metis
$ ./get.Metis
```

- Get the HSL subroutines from `http://hsl.rl.ac.uk/ipopt`
- Note that there are two releases for HSL - you will want the more complete one that contains ma57, ma77, and ma97.
- While the freely available ma27 will work for many problems, the newer packages are faster, work on larger problems, and can use multi-core architecture.

- This will require filling out a form stating essentially that you are in academia and waiting a couple hours for a link to download.
- Unpack the resulting library into the ThirdParty folder such that the path is (IPOPT Path)/ThirdParty/HSL/coinhsl

Install

- Go to the IPOPT directory

```
$ mkdir build
$ cd build
$ ../configure
```

- Note that if you have lapack or blas installed previously you can use –with-lapack and –with-blas to link to those packages
- If something goes wrong refer here

  `http://www.coin-or.org/Ipopt/documentation/node19.html`

- Assuming everything worked:

```
$ make
$ make test
$ make install
```

A.5 minAone.py Description

minAone is a python script used to write C++ code and compiler instructions using the IPOPT (Interior Point OPTimization) libraries to estimate unmeasured states and parameters in dynamical systems with limited measurements. The scripts take a set of differential equations and state and parameter names provided by a text

file "equations.txt" and returns a set of C++ files consisting of a set of constraints based on a discretized version of those differential equations. A second text file 'specs.txt' allows for changes in run specific quantities state and parameter bounds, as well as input files without the need to recompile.

List of Files

- discAone.py

  -Discretizes equations and creates strings for Jacobian and Hessian Elements.

- makecppAone.py

  -Writes C++ file linking to IPOPT libraries using strings from discAone.py

- makehppAone.py

  -Writes header file for above

- makemakeAone.py

  -Writes makefile for problem. Will need to be changed based on install location of IPOPT

- makeoptAone.py

  -Writes settings file for IPOPT

These files can be put in /usr/local/sbin for ease of use

Modify makemakeAone.py

The Makefile compiles C ++ object files and links them with the installed IPOPT libraries, in order to create an executable. Since the location of the IPOPT

libraries, as well as the flags used to compile them, differ between installations, this file will be unique to a given machine. Modification of the makemake.py script to give correct Makefiles for a given machine consists of:

- Ensure that the IPOPT installation proceeded correctly, as evidenced by zero errors for the make install step.
- In the IPOPT build directory, try to compile (make) one of the examples, for instance at /build/Ipopt/examples/hs071 cpp.
- If this compiles and runs correctly, open the Makefile in this directory.
- Make note of the entries in the following fields of this Makefile: CXX, CXXFLAGS, CXXLINK- FLAGS, INCL, LIBS.
- In makemake.py, replace the default entries for these fields with those given in the example Makefile.
  - makemakeAone.py is formatted differently than a Makefile, since it is a python code generation script.
  - Lines that begin with the # sign will be comments in the Makefile - leave these alone.
  - All lines must end with \n\in order for the Makefile to be generated correctly.
  - The best way to ensure that all the compile flags are correct is to copy and paste from the example Makefile, ensuring that the end line characters are in place.

- The modification of makemake.py must only be done once for a given machine, unless IPOPT is reinstalled for whatever reason.

A.6 Running the Code

minAone uses two text documents (along with any needed data files) as input, equations.txt and specs.txt. Once these are filled

equations.txt contains information on the model and is used once for generating the needed cpp and hpp files for the run. The file should be written as described below in this order.

- The first line is the problem name, this name will be used to name the resulting executable.
- The second line tells minAzero how many dynamical variables, parameters, coupling terms, stimuli, functions, and measurements there are, in that order as a comma delimited list. It is essential that these numbers are accurate as minAzero uses this to know how many lines to read for each component of the code.
- A list of every differential equation.
- The measurement term of the cost function. A penalty term for coupling terms is suggested as any coupling to measurements is not present in physical systems.
- The names of all the variables. These must be the same as used in the differential equations and should be multiple letters/and or numbers such that variable name is contained in any other name or common function.

- The names of parameters, names of couplings, names of data, and names of stimuli, in that order. Again use fully unique names.
- Function names and number of arguments of that function separated by a comma. Use a function if there is some component of the dynamics with a removable singularity or other difficult numerical object that requires an alternative local definition.
- Functions will require an additional file 'myfunctions.cpp' containing the function definition along with its jacobian and hessian (an example of this is included)

specs.txt contains run specific information such as file names, variable bounds, and problem length. This file can be edited without recompiling the code.

- First line is the number of full steps the code will use. Because the code is compiled using a midpoint method, the actual problem length will double this plus one.
- Second line is the number of lines in each input file to skip. This allows for the code to start at any point in a long data set.
- Third line is double the time step of the data. Again since a midpoint method is used, the time step is for a whole step - which includes two points.
- If you wish to start at a non constant guess, you can put a 1 followed by a line with an initial condition file. This file should have one column for each state. If you do not want to include an initial condition file, use 0
- One line for each of the measured data file names. Each file should be a single column.

- One line for each of the stimulus data file names. Each file should be a single column.
- For each variable, the lower bound, upper bound, and RF0 value separated by commas.
- For each parameter a lower bound, upper bound
- One line for annealing settings, alpha, incresement of beta and maximum beta separated by commas

Once everything is filled out and all data files are present, you can run the python scripts:

```
$ minAone.py
$ make
$ ./(problem_name)_cpp
```

If data files are missing or too short, the code will segfault. The output file contains annealing result for one path named like `D5_M1_PATH0.dat`. Each line of `D5_M1_PATH0.dat` contains the optimal path at different values of `beta`. The first three numbers are beta exitflag and action value, respectively. Exitflag can be 0 or 1. 1 means IPopt routines find the optimal path and 0 means it fails. The rest numbers represent the optimal path.

```
beta exitflag action_value
optimal_path[x1(0) x2(0) x3(0) x4(0) x5(0)  x1(1) x2(1) x3(1)
x4(1) x5(1) ... x1(NT) x2(NT) x3(NT) x4(NT) x5(NT)
 p(1) p(2) ... p(NP)]
```

A.7 Run in Parallel

One excute `(problem_name)_cpp` can obtain the result for only one random initial path. To explore the landscape of action A_0, we need to start from different random paths and each of them will converge to different local minima. Since all those paths are indenpendent from each other, it is easy to implement the calculation in parallel using array job.

Here we give a example submission scripts on ccom-boom cluster

```
#!/bin/bash
#$ -t 1-100
#$ -N job_name
#$ -cwd
#$ -j y
#$ -M your@email.com
#$ -S /bin/bash
#$ -m beas
#$ -o ./output
#$ -e ./error
#$ -q batch.q
./problem_name_cpp $SGE_TASK_ID
```

Each path will be stored in individual file with the name like `D5_M1_PATH0.dat`, `D5_M1_PATH2.dat`,...,`D5_M1_PATH100.dat`.

A.8 Examples

Two examples are provided: the first one is Lorenz96 D=5 to show the basic settings of `equations.txt` and `specs.txt`. And the other example is NaKL to show how to include external stimuli in `equations.txt` and `specs.txt`.

A.8.1 Lorenz96 D=5

Lorenz96 D=5 Vector Field

$$\frac{dx_1}{dt} = x_5(x_2 - x_4) - x_1 + f$$
$$\frac{dx_2}{dt} = x_1(x_3 - x_5) - x_2 + f$$
$$\frac{dx_3}{dt} = x_2(x_4 - x_1) - x_3 + f$$
$$\frac{dx_4}{dt} = x_3(x_5 - x_2) - x_4 + f$$
$$\frac{dx_5}{dt} = x_4(x_1 - x_3) - x_5 + f$$

Lorenz96 D=5 equations.txt

```
# Problem Name
lorenz96
# nY,nP,nU,nI,nF,nM
5,1,0,0,0,1
# equations
yy5*(yy2-yy4)-yy1+FF1
yy1*(yy3-yy5)-yy2+FF1
yy2*(yy4-yy1)-yy3+FF1
yy3*(yy5-yy2)-yy4+FF1
yy4*(yy1-yy3)-yy5+FF1
# Objective/Cost function
4*(data1-yy1)*(data1-yy1)
# variable names
yy1
yy2
yy3
yy4
yy5
# parameter names
FF1
# data names
data1
# stimuli names
```

Lorenz96 D=5 specs.txt

```
# Includes the problem length
80
# How much data to skip
# In case you do not want to start at the beginning of the data file
100
# Time step - this is twice the time step of the data,
# since the data includes time and midpoints.
0.02
# Data File names - input
x1.dat
# Data File name - stimuli
# No stimuli for this problem
# Boundary & initial conditions
# 0 for no initial data file, 1 for data file
# A data file must include values for all state variables
# at each time point.
0
# If above is 1, list name of data file next.  If 0, no entry needed.
# State Variables:
# These are in the formats: lower bound, upper bound, Rf0
# y1
-15, 15, 0.01
# y2
-15, 15, 0.01
# y3
-15, 15, 0.01
# y4
-15, 15, 0.01
# y5
-15, 15, 0.01
# Parameters:
0, 20, 8.17
#annealing setting: R_f = R_f0*alpha^beta
#There are in the formats: alpha, incresement of beta, maximum beta
# here we have alpha=2, beta = 0, 1, 2, 3, ..., 29
2,1,30
```

A.8.2 NaKL

NaKL Vector Field

$$\frac{dV}{dt} = CI_{inj}(t) + g_{Na}m^3h(E_{Na} - V) + g_K n^4(E_K - V) + g_L(E_L - V)$$
$$\frac{da}{dt} = \frac{a_\infty - a}{\tau_a}, \quad a = \{m, h, n\}$$
$$a_\infty = \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{V - V_a}{\Delta V_a}\right)$$
$$\tau_a = \tau_{a0} + \tau_{a0}\left(1 - \tanh^2\left(\frac{V - V_a}{\Delta V_a}\right)\right)$$

NaKL equations.txt

```
simple_nakl
# nY,nP,nU,nI,nF,nM
4,19,0,1,0,1
#vector field
gNa*(m0*m0*m0*h0)*(ENa-V0)+gK*n0*n0*n0*n0*(EK-V0)+gL*(EL-V0)+Area*Iinj
(0.5*(1+tanh((V0-Vmo)*dVm)) - m0)/(Cm1+Cm2*(1.0-tanh((V0-Vmo)*dVm)
*tanh((V0-Vmo)*dVm)))
(0.5*(1+tanh((V0-Vho)*dVh)) - h0)/(Ch1+Ch2*(1.0-tanh((V0-Vho)*dVh)
*tanh((V0-Vho)*dVh)))
(0.5*(1+tanh((V0-Vno)*dVn)) - n0)/(Cn1+Cn2*(1.0-tanh((V0-Vno)*dVn)
*tanh((V0-Vno)*dVn)))
#obj function
(VDATA0 - V0)*(VDATA0 - V0)
#states
V0
m0
h0
n0
#parameters
gNa
ENa
gK
EK
gL
EL
Area
```

```
Vmo
dVm
Cm1
Cm2
Vho
dVh
Ch1
Ch2
Vno
dVn
Cn1
Cn2
#data names
VDATA0
#stimuli
Iinj
```

NaKL specs.txt

```
3000
0
0.04
#data
./noise_measured.dat
#stimuli
./current.dat
0
#./allstates.dat
# state bounds and Rf0
-150,70,1e-3
0, 1,1e1
0, 1,1e1
0, 1,1e1
# parameter bounds
#gna
50,200,100,120
#Ena
0,100,50,50
#gki
5,40,30,20
#Ek
-100,-50,-70,-77
#gl
```

```
0.1,1,.2,.3
#El
-60,-50,-52,-54
#Area
0.5,1.5,1,0.8
#mv1
-60,-30,-45,-40
#mv2
.01,0.1,.075,0.06667
#cm1
0.05,.25,.15,.1
#cm2
.1,1,.4,.4
#hv1
-70,-40,-50,-60
#hv2
-0.1,-.01,-.05,-.06667
#ch1
.1,5,1.2,1
#ch2
1,15,6,7
#nv1
-70,-40,-52,-55
#nv2
.01,0.1,.03,.03333
#cn1
.1,5,.8,1
#cn2
2,12,5,5
#anneal settings
2,1,30
```

A.9 Troubleshooting

I have tested these scripts over a wide range of problems, so I believe that the algorithms are correct. However, there are a few common errors that may crop up.

- Variable and parameter naming is very important. At few common problems can crop up. Never use a variable name that includes the name of another variable.

For instance p1 and p11 would be bad, since p11 includes p1. In this case, p01 and p11 would be adequate. Along this vein, all variable names should be at least 2 characters long, just in case.

Bibliography

[1] Henry D. I. Abarbanel. *Analysis of observed chaotic data*. Springer, New York, 1996. ISBN 0-387-94523-7.

[2] Henry D. I. Abarbanel, Mark Kostuk, and William Whartenby. Data assimilation with regularized nonlinear instabilities. *Quarterly Journal of the Royal Meteorological Society*, 136(648):769–783, 2010.

[3] Henry DI Abarbanel. *Predicting the Future: Completing Models of Observed Complex Systems*. Springer, 2013.

[4] Henry DI Abarbanel. *Predicting the Future: Completing Models of Observed Complex Systems*. Springer, 2013.

[5] Henry DI Abarbanel, Daniel R Creveling, Reza Farsian, and Mark Kostuk. Dynamical state and parameter estimation. *SIAM Journal on Applied Dynamical Systems*, 8(4):1341–1381, 2009.

[6] Uri M. Ascher, Robert M. M. Mattheij, and Robert D. Russell. *Numerical solution of boundary value problems for ordinary differential equations*, volume 13. SIAM, 1995.

[7] A. F. Bennett. *Inverse Methods in Physical Oceanography*. Cambridge University Press, 1992.

[8] Michael Berridge, Peter Lipp, and Martin Bootman. Calcium signalling. *Current biology*, 9(5):R157–R159, 1999.

[9] Michael J Berridge. Elementary and global aspects of calcium signalling. *Journal of experimental biology*, 200(2):315–319, 1997.

[10] Michael J Berridge. Neuronal calcium signaling. *Neuron*, 21(1):13–26, 1998.

[11] Michael J Berridge, Peter Lipp, and Martin D Bootman. The versatility and universality of calcium signalling. *Nature reviews Molecular cell biology*, 1(1): 11–21, 2000.

[12] Michael J Berridge, Martin D Bootman, and H Llewelyn Roderick. Calcium signalling: dynamics, homeostasis and remodelling. *Nature Reviews Molecular Cell Biology*, 4(7):517–529, 2003.

[13] A. Beskos, N. Pillai, G.O. Roberts, J.-M. Sanz-Serna, and A.M. Stuart. Optimal tuning of hybrid monte carlo algorithm. *Bernoulli*, 19:1501–1534, 2013. doi: 10.3150/12-BEJ414.

[14] JoséA M Borghans, Geneviève Dupont, and Albert Goldbeter. Complex intracellular calcium oscillations a theoretical exploration of possible mechanisms. *Biophysical chemistry*, 66(1):25–41, 1997.

[15] Jochen Bröcker. On variational data assimilation in continuous time. *Quarterly Journal of the Royal Meteorological Society*, 136(652):1906–1919, 2010.

[16] Jochen Bröcker. My view on weak constraint four dimensional variational assimilation. *Summer School/Creative Workshop: Data Assimilation & Inverse Problems*, 2013. URL http://www.inverseproblems.info/reading:summer$_$school$_$2013.

[17] Jochen Bröcker and Ivan G Szendro. Sensitivity and out-of-sample error in continuous time data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 138(664):785–801, 2012.

[18] Teresa R Chay and John Rinzel. Bursting, beating, and chaos in an excitable membrane model. *Biophysical Journal*, 47(3):357–366, 1985.

[19] Teresa Ree Chay. Modelling for nonlinear dynamical processes in biology. *Patterns, Information and Chaos in Neuronal Systems*, pages 73–122, 1993.

[20] Teresa Ree Chay. Electrical bursting and luminal calcium oscillation in excitable cell models. *Biological cybernetics*, 75(5):419–431, 1996.

[21] B. S. Chua and A. F. Bennett. An inverse ocean modeling system. *Ocean Modelling*, 3:137–165, 2001.

[22] Arij Daou, Matthew Ross, Frank Johnson, Richard L Hyson, and Richard Bertram. Electrophysiological characterization and computational models of hvc neurons in the zebra finch. *Journal of Neurophysiology*, 2013.

[23] C Jane Dixon, Peter H Cobbold, and Anne K Green. Oscillations in cytosolic free ca2+ induced by adp and atp in single rat hepatocytes display differential sensitivity to application of phorbol ester. *Biochem. J*, 309:145–149, 1995.

[24] Michael R Duchen. Contributions of mitochondria to animal physiology: from homeostatic sensor to calcium signalling and cell death. *The Journal of physiology*, 516(1):1–17, 1999.

[25] G Dupont, G Houart, and A Goldbeter. From simple to complex ca 2+ oscillations: regulatory mechanisms and theoretical models. In *Understanding Calcium Dynamics*, pages 131–152. Springer, 2003.

[26] Geneviève Dupont and Albert Goldbeter. One-pool model for Ca^{2+} oscillations involving Ca^{2+} and inositol 1, 4, 5-trisphosphate as co-agonists for Ca^{2+} release. *Cell calcium*, 14(4):311–322, 1993.

[27] Hendrik Eibern and Hauke Schmidt. A four-dimensional variational chemistry data assimilation scheme for eulerian chemistry transport modeling. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 104(D15):18583–18598, 1999.

[28] Jens Eilers and Arthur Konnerth. Dye loading with patch pipettes. *Cold Spring Harbor Protocols*, 2009(4):pdb–prot5201, 2009.

[29] Thomas Elbert, William J Ray, Zbigniew J Kowalik, James E Skinner, Karl Eugen Graf, and Niels Birbaumer. Chaos and physiology: deterministic chaos in excitable cell assemblies. *Physiological Reviews*, 74(1):1–48, 1994.

[30] Geir Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer, 2009.

[31] Martin Falcke, Ramón Huerta, Mikhail I Rabinovich, Henry DI Abarbanel, Robert C Elson, and Allen I Selverston. Modeling observed chaotic oscillations in bursting neurons: the role of calcium dynamics and IP3. *Biological Cybernetics*, 82(6):517–527, 2000.

[32] Willliam Feller. *An introduction to probability theory and its applications*, volume 2. John Wiley & Sons, 2008.

[33] David D Friel. Calcium oscillations in neurons. *Calcium waves, gradients and oscillations*, 188:210, 1995.

[34] Sergio Fucile. Ca^{2+} permeability of nicotinic acetylcholine receptors. *Cell calcium*, 35(1):1–8, 2004.

[35] I. M. Gelfand and S. V. Fomin. *Calculus of Variations*. Dover Publications, Inc., 1963.

[36] I. M. Gelfand and A. M. Yaglom. Integration in functional spaces and its applications in quantum physics. *Journal of Mathematical Physics*, 1(1):48–69, 1960.

[37] Andrew Gelman, John B. Carlin, Hal S. Stern, and Donald B. Rubin. *Baysian Data Analysis: Second Edition*. Chapman & Hall, CRC Press, 2004.

[38] Albert Goldbeter et al. Biochemical oscillations and cellular rhythms. *Biochemical Oscillations and Cellular Rhythms, by Albert Goldbeter, Foreword by MJ Berridge, Cambridge, UK: Cambridge University Press, 1997*, 1, 1997.

[39] Michael H Graber, Fritjof Helmchen, and Richard HR Hahnloser. Activity in a premotor cortical nucleus of zebra finches is locally organized and exhibits auditory selectivity in neurons but not in glia. *PloS one*, 8(12):e81177, 2013.

[40] Anne K Green, C Jane Dixon, Alexander G McLennan, Peter H Cobbold, and Michael J Fisher. Adenine dinucleotide-mediated cytosolic free ca¡ sup¿ 2+¡/sup¿ oscillations in single hepatocytes. *FEBS letters*, 322(2):197–200, 1993.

[41] Anne K Green, Peter H Cobbold, and C Jane Dixon. Effects on the hepatocyte [ca¡ sup¿ 2+¡/sup¿]¡ sub¿ i¡/sub¿ oscillator of inhibition of the plasma membrane ca¡ sup¿ 2+¡/sup¿ pump by carboxyeosin or glucagon-(19–29). *Cell calcium*, 22 (2):99–109, 1997.

[42] Christine Grienberger and Arthur Konnerth. Imaging calcium in neurons. *Neuron*, 73(5):862–885, 2012.

[43] Saul Hazledine, Jongho Sun, Derin Wysham, J Allan Downie, Giles ED Oldroyd, and Richard J Morris. Nonlinear time series analysis of nodulation factor induced calcium oscillations: evidence for deterministic chaos? *PloS one*, 4(8):e6637, 2009.

[44] Michael J Higley and Bernardo L Sabatini. Calcium signaling in dendrites and spines: practical and functional considerations. *Neuron*, 59(6):902–913, 2008.

[45] David Hochberg, Carmen Molina-Paris, Juan Perez-Mercader, and Matt Visser. Effective action for stochastic partial differential equations. *Physical Review E*, 60(6):6343, 1999.

[46] Alan L Hodgkin and Andrew 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.

[47] Gérald Houart, Geneviève Dupont, and Albert Goldbeter. Bursting, chaos and birhythmicity originating from self-modulation of the inositol 1, 4, 5-trisphosphate signal in a model for intracellular Ca^{2+} oscillations. *Bulletin of mathematical biology*, 61(3):507–530, 1999.

[48] David B Jaffe, Daniel Johnston, Nechama Lasser-Ross, John E Lisman, Hiroyoshi Miyakawa, and William N Ross. The spread of Na^{+} spikes determines the pattern of dendritic Ca^{2+} entry into hippocampal neurons. *Nature*, 357(6375):244–246, 1992.

[49] Hongbo Jia, Nathalie L Rochefort, Xiaowei Chen, and Arthur Konnerth. In vivo two-photon imaging of sensory-evoked dendritic calcium signals in cortical neurons. *Nature protocols*, 6(1):28–35, 2010.

[50] Daniel Johnston and Samuel Miao-Sin Wu. *Foundations of Cellular Neurophysiology*. Bradford Books, MIT Press, 1995.

[51] Daniel Johnston, Samuel Miao-Sin Wu, and Richard Gray. *Foundations of cellular neurophysiology*. MIT press Cambridge, 1995.

[52] B Jouvet and R Phythian. Quantum aspects of classical and statistical fields. *Physical Review A*, 19(3):1350, 1979.

[53] Masaki Kato and Kazuo Okanoya. Molecular characterization of the song control nucleus hvc in bengalese finch brain. *Brain research*, 1360:56–76, 2010.

[54] David Stewart Kendall Atkinson, Weimin Han. *Numerical Solution of Ordinary Differential Equations*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2009.

[55] Donald E. Kirk. *Optimal Control Theory: An Introduction*. Dover Publications, Inc., 1970.

[56] Scott Kirkpatrick, MP Vecchi, et al. Optimization by simmulated annealing. *science*, 220(4598):671–680, 1983.

[57] Kazuo Kitamura, Benjamin Judkewitz, Masanobu Kano, Winfried Denk, and Michael Häusser. Targeted patch-clamp recordings and single-cell electroporation of unlabeled neurons in vivo. *Nature methods*, 5(1):61–67, 2007.

[58] Mark Kostuk. Synchronization and statistical methods for the data assimilation of hvc neuron models. *PhD Dissertation in Physics, University of California, San Diego*, 2012. URL http://escholarship.org/uc/item/2fh4d086.

[59] Mark Kostuk, BryanA. Toth, C.Daniel Meliza, Daniel Margoliash, and HenryD.I. Abarbanel. Dynamical estimation of neuron and network properties ii: path integral monte carlo methods. *Biological Cybernetics*, 106(3):155–167, 2012. ISSN 0340-1200.

[60] Mark Kostuk, BryanA. Toth, C.Daniel Meliza, Daniel Margoliash, and HenryD.I. Abarbanel. Dynamical estimation of neuron and network properties ii: path integral monte carlo methods. *Biological Cybernetics*, 106(3):155–167, 2012.

[61] Sonja Kosuta, Saul Hazledine, Jongho Sun, Hiroki Miwa, Richard J Morris, J Allan Downie, and Giles ED Oldroyd. Differential and chaotic calcium signatures in the symbiosis signaling pathway of legumes. *Proceedings of the National Academy of Sciences*, 105(28):9823–9828, 2008.

[62] Mark Kot. *A First Course In the Calculus of Variations*. American Mathematical Society; Providence, Rhode Island., 2014.

[63] M Kubota and N Saito. Sodium-and calcium-dependent conductances of neurones in the zebra finch hyperstriatum ventrale pars caudale in vitro. *The Journal of physiology*, 440(1):131–142, 1991.

[64] Thomas Ladewig and Bernhard U Keller. Simultaneous patch-clamp recording and calcium imaging in a rhythmically active neuronal network in the brainstem slice preparation from mouse. *Pflügers Archiv*, 440(2):322–332, 2000.

[65] Rubin H. Landau, Manuel Jose' Paez, and Cristian C. Bordeianu. *A Survey of Computational Physics: Introductory Computational Science*. Princeton University Press, 2010.

[66] Pierre Simon Laplace. Memoir on the probability of causes of events. *Mémoires de Mathématique et de Physique, Tome Sixième*, pages 621–656, 1774.

[67] Yue-Xian Li, SS Stojilković, Joel Keizer, and John Rinzel. Sensing and refilling calcium stores in an excitable cell. *Biophysical journal*, 72(3):1080–1091, 1997.

[68] Daniel Liberzon. *Calculus of Variations and Optimal Control Theory*. Princeton University Press, 2012.

[69] Juliane Liepe, Paul Kirk, Sarah Filippi, Tina Toni, Chris P Barnes, and Michael PH Stumpf. A framework for parameter estimation and model selection from experimental data in systems biology using approximate bayesian computation. *Nature protocols*, 9(2):439–456, 2014.

[70] Michael A Long, Dezhe Z Jin, and Michale S Fee. Support for a synaptic chain model of neuronal sequence generation. *Nature*, 468(7322):394–399, 2010.

[71] Andrew C Lorenc and Tim Payne. 4d-var and the butterfly effect: Statistical four-dimensional data assimilation for a wide range of scales. *Quarterly Journal of the Royal Meteorological Society*, 133(624):607–614, 2007.

[72] Edward N Lorenz. Deterministic nonperiodic flow. *Journal of the atmospheric sciences*, 20(2):130–141, 1963.

[73] Edward N Lorenz. Predictability: A problem partly solved. In Tim Palmer and Renate Hagedorn, editors, *Predictability of weather and climate*. Cambridge, 2006.

[74] Troy W Margrie, Michael Brecht, and Bert Sakmann. In vivo, low-resistance, whole-cell recordings from neurons in the anaesthetized and awake mammalian brain. *Pflügers Archiv*, 444(4):491–498, 2002.

[75] I Marrero, A Sanchez-Bueno, PH Cobbold, and CJ Dixon. Taurolithocholate and taurolithocholate 3-sulphate exert different effects on cytosolic free ca2+ concentration in rat hepatocytes. *Biochem. J*, 300:383–386, 1994.

[76] Thomas Nowotny, Rafael Levi, and Allen I Selverston. Probing the dynamics of identified neurons with a data-driven modeling approach. *PloS one*, 3(7):e2627, 2008.

[77] Lars Onsager and S Machlup. Fluctuations and irreversible processes. *Physical Review*, 91(6):1505, 1953.

[78] Carlos Pires, Robert Vautard, and Olivier Talagrand. On extending the limits of variational assimilation in nonlinear chaotic systems. *Tellus A*, 48(1):96–121, 1996.

[79] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge Univ. Press, 1992.

[80] William H Press et al. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2012.

[81] John C Quinn. A path integral approach to data assimilation in stochastic nonlinear systems. *PhD Dissertation in Physics, University of California, San Diego*, 2010. URL http://escholarship.org/uc/item/0bm253qk.

[82] I Scott Ramsey, Markus Delling, and David E Clapham. An introduction to trp channels. *Annu. Rev. Physiol.*, 68:619–647, 2006.

[83] Thomas A Rooney, Suresh K Joseph, Christina Queen, and Andrew P Thomas. Cyclic gmp induces oscillatory calcium signals in rat hepatocytes. *Journal of Biological Chemistry*, 271(33):19817–19825, 1996.

[84] Pankaj Sah. Ca^{2+}-activated K^+ currents in neurones: types, physiological roles and modulation. *Trends in neurosciences*, 19(4):150–154, 1996.

[85] Steven J Schiff, Kristin Jerger, Duc H Duong, Taeun Chang, Mark L Spano, William L Ditto, et al. Controlling chaos in the brain. *Nature*, 370(6491): 615–620, 1994.

[86] Beat Schwaller. Cytosolic ca^{2+} buffers. *Cold Spring Harbor perspectives in biology*, 2(11), 2010.

[87] James E Skinner, Craig M Pratt, and Tomas Vybiral. A reduction in the correlation dimension of heartbeat intervals precedes imminent ventricular fibrillation in human subjects. *American heart journal*, 125(3):731–743, 1993.

[88] N Solovyova, N Veselovsky, EC Toescu, and A Verkhratsky. Ca^{2+} dynamics in the lumen of the endoplasmic reticulum in sensory neurons: direct visualization of Ca^{2+}-induced Ca^{2+} release triggered by physiological Ca^{2+} entry. *The EMBO journal*, 21(4):622–630, 2002.

[89] David Sterratt, Bruce Graham, Andrew Gillies, and David Willshaw. *Principles of Computational Modelling in Neuroscience*. Cambridge University Press, 2011.

[90] Endre Süli. Numerical solution of ordinary differential equations. *people.maths.ox.ac.uk/suli/nsodes.pdf*, pages 1–82, 2014.

[91] Karel Svoboda, Winfried Denk, David Kleinfeld, David W Tank, et al. In vivo dendritic calcium dynamics in neocortical pyramidal neurons. *Nature*, 385(6612): 161–165, 1997.

[92] Hajime Takechi, Jens Eilers, and Arthur Konnerth. A new class of synaptic response involving calcium release in dendritic spines. *Nature*, 396(6713):757–760, 1998.

[93] Kevin Thurley, Alexander Skupin, Rüdiger Thul, and Martin Falcke. Fundamental properties of Ca $^{2+}$ signals. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1820(8):1185–1194, 2012.

[94] BryanA. Toth, Mark Kostuk, C.Daniel Meliza, Daniel Margoliash, and HenryD.I. Abarbanel. Dynamical estimation of neuron and network properties i: variational methods. *Biological Cybernetics*, 105(3-4):217–237, 2011.

[95] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

[96] J Martin Wild, Matthew N Williams, Graham J Howie, and Richard Mooney. Calcium-binding proteins define interneurons in hvc of the zebra finch (taeniopygia guttata). *Journal of Comparative Neurology*, 483(1):76–90, 2005.

[97] Daniel S. Wilks. Effects of stochastic parametrizations in the lorenz'96 system. *Quarterly Journal of the Royal Meteorological Society*, 131(606):389–407, 2005.

[98] Niall M Woods, KS Roy Cuthbertson, and Peter H Cobbold. Repetitive transient rises in cytoplasmic free calcium in hormone-stimulated hepatocytes. 1986.

[99] J. Ye, N. Kadakia, P. J. Rozdeba, H. D. I. Abarbanel, and J. C. Quinn. Improved variational approximations in statistical data assimilation. *Nonlinear Processes in Geophysics Discussions*, 1(2):1603–1620, 2014.

[100] Jingxin Ye, Bryan Toth, and Chris Knowlton. Annealing code minaone. 2015. URL https://github.com/yejingxin/minAone.

[101] Steven R. Young, Robert K.S. Wong, and Riccardo Bianchi. Simultaneous intracellular recording and calcium imaging in single neurons of hippocampal slices. *Methods*, 21(4):373 – 383, 2000.

[102] Ciyou Zhu, Richard H Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997.

[103] Jean Zinn-Justin. *Quantum field theory and critical phenomena*. Oxford University Press, 2002.