ABSTRACT

LANKFORD, GEORGE BERNARD. Optimization, Modeling, and Control: Applications to Klystron Designing and Hepatitis C Virus Dynamics. (Under the direction of Dr. Hien Tran.)

In this dissertation, we address applying mathematical and numerical techniques in the fields of high energy physics and biomedical sciences. The first portion of this thesis presents a method for optimizing the design of klystron circuits. A klystron is an electron beam tube lined with cavities that emit resonant frequencies to velocity modulate electrons that pass through the tube. Radio frequencies (RF) inserted in the klystron are amplified due to the velocity modulation of the electrons. The routine described in this work automates the selection of cavity positions, resonant frequencies, quality factors, and other circuit parameters to maximize the efficiency with required gain. The method is based on deterministic sampling methods. We will describe the procedure and give several examples for both narrow and wide band klystrons, using the klystron codes AJDISK (Java) and TESLA (Python).

The rest of the dissertation is dedicated to developing, calibrating and using a mathematical model for hepatitis C dynamics with triple drug combination therapy. Groundbreaking new drugs, called direct acting antivirals, have been introduced recently to fight off chronic hepatitis C virus infection. The model we introduce is for hepatitis C dynamics treated with the direct acting antiviral drug, telaprevir, along with traditional interferon and ribavirin treatments to understand how this therapy affects the viral load of patients exhibiting different types of response. We use sensitivity and identifiability techniques to determine which parameters can be best estimated from viral load data. We use these estimations to give patient-specific fits of the model to partial viral response, end-of-treatment response, and breakthrough patients. We will then revise the model to incorporate an immune response dynamic to more accurately describe the dynamics. Finally, we will implement a suboptimal control to acquire a drug treatment regimen that will alleviate the systemic cost associated with constant drug treatment.

Optimization, Modeling, and Control: Applications to Klystron Designing and Hepatitis C Virus Dynamics

by
George Bernard Lankford

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2016

APPROVED BY:

Dr. Mansoor Haider	Dr. Ralph Smith
Dr. Marina Evans	Dr. Hien Tran Chair of Advisory Committee

DEDICATION

I dedicate this to Sharonda Hill-Stevens Porter. She isn't perfect, but I know the large amount of love in her heart. She has endured so much and still will take anyone off the street into her home whether she likes you or not. She took me in, gave me a roof over my head and food in my mouth. Even though we may not always see eye to eye, I love you more than you'll ever know.

BIOGRAPHY

George Lankford was born and raised in Huntsville, AL in August 10, 1989. He started out living with his grandmother and mother, Lillian Lankford and Angela Lankford. Angela and George moved out of his grandmother's house and and not soon after, his sister Essence Lankford was born. Her father, James Porter, is a strong influence for who he is today. After a turbulent time that included living at the Salvation Army, Angela, George, and Essence moved into the projects of Binford Gardens. He stayed here from 3rd to around 8th grade. He then was taken in by James' brother and sister-in-law, Juan and Sharonda Porter. Though not the best situation, George appreciates the kindness and love this family gave to him despite not being kin by blood. After graduating from Huntsville High School, George moved to LaGrange, GA to play football at LaGrange College. During the breaks in holidays, he began living with the Johnson family that consisted of one of his best childhood friends, Erin Johnson, and her parents, Wylie and Mishele Johnson. George loved visiting during breaks because there was so much love and relaxation. After sleeping through a good portion of his first math class, a serious talk with his Calculus I teacher and soon to be advisor, Dr. William Yin, brought a sense of urgency for George to focus on his education. Four years later, he received his B.S. in mathematics. After meeting his future mentor, Terrance Pendleton, he made his way to Raleigh, NC to become a PhD student in applied mathematics at North Carolina State University. Graduate school proved to be the most difficult, but rewarding experiences of his life. His maturity level exponentially grew during his 5 years at NC State and he wouldn't trade it for the world. He looks forward to beginning his journey without school by serving his country as a data scientist.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my advisor, Dr. Hien Tran. It is tough to really know how many times I popped in your office and went "You got a second Dr. T?" I am so grateful for all of the knowledge and assistance you have imparted to me these past five years. You made this process enjoyable and I am thankful to be your student.

I would like to also thank all of my committee members for their assistance in my journey. Dr. Evans and Dr. Haider, thank you for being supportive of me during this process. Dr. Smith, I am so appreciative for your kindness over the years. You were one of the first professors that I met when I got here. Countless times we've met in the hall at SAS and warmheartedly you would say, "George, how's it going?" Thank you for being a welcoming smile.

There are too many coaches, teachers, and professors that I want to put here, however, I want to point out a couple. I want to acknowledge my wrestling coach Bengie Balentine and my advisor, Dr. William Yin and his family. Thank you for your endless support and encouraging words through this process.

To President Linda Buchanan, Tom and Pandora Economy, and Charles and Jeannie Smith, I love you all so very much. You and your families have been there for me when I needed you during my journey and I can't thank you enough.

I acknowledge all of my family, the Lankfords, the Johnsons, the Porters, and whomever else has been like family to me. I'm grateful for your support during my journey. Special thanks to all my friends like Bill Simons and Kaska Adoteye who have been there for me since the beginning of graduate school. Also, shoutout to Emaan Abdul-Majid, Joshua Richardson, and Zach Hough for your encouraging words and enduring my stories of weekly crying at the end of my tenure as a graduate student.

Last, but certainly not least, I would like to thank from the bottom of my heart, my mentor/guardian angel, Terrance Pendleton aka T aka Tebow. Your guidance has brought me the farthest in my journey in becoming a good human being. As I reminisce about all the nights we stayed up doing work, all the nights I stayed up talking while you were doing work, and all the lessons you tried and continue to teach me it makes my heart smile. I am a better mathematician because of what you have taught me, but more importantly, I will be a better husband and father because of you. Thank you for being my best friend.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

The groundwork for mathematics is founded upon solving problems. Around 200 B.C., an ancient Chinese book called *Chiu-chang Suan-shu(Nine Chapters on Arithmetic)* was written. At the beginning of Chapter VIII, the following problem is given:

> *Three sheafs of a good crop, two sheafs of a mediocre crop, and one sheaf of a bad crop are sold for 39 dou. Two sheafs of good, three mediocre, and one bad are sold for 34 dou; and one good, two mediocre, and three bad are sold for 26 dou. What is the price received for each sheaf of a good crop, each sheaf of a mediocre crop, and each sheaf of a bad crop?*

This problem can subsequently be formulated as the system of linear equations given in (1.1):

$$\begin{aligned} 3x+2y+z &= 39, \\ 2x+3y+z &= 34, \\ x+2y+3z &= 26, \end{aligned} \tag{1.1}$$

where x is the price of one sheaf of the good crop, y is the price of one sheaf of the mediocre crop, and z is the price of one sheaf of the bad crop. There are many tools that have been developed to

solve such a problem. One approach involves methodically eliminating variables by adding and/or subtracting multiples of the equations from each other. Another is by putting the coefficients into an augmented matrix and using Gaussian elimination. Indeed, solving problems is a catalyst for many branches of mathematics. However, certain problems may not always be understood in the physical sense. To this extent, people with a combination of mathematics and specialized knowledge are needed to bridge the gap between theory and applications. The foundation of applied mathematics involves studying methodology in mathematics, and using those methods to quantify and/or solve practical problems. This thesis concerns itself with furthering the field of applied mathematics by exploring how different mathematical techniques can be used to investigate real-world applications concerning klystron design and viral kinetic modeling.

1.1 Thesis Outline

The remainder of this chapter is dedicated to developing certain techniques in optimization and modeling that will be relevant to our study. The implementation of the aforementioned techniques with applications to beam tube design optimization and biological modeling is described in the remaining chapters. In Chapter 2, an automated algorithm is presented to improve klystron design. Chapter 3 is dedicated to developing and validating a new model for hepatitis C dynamics with triple drug therapy. In Chapter 4, the model in Chapter 3 is further developed to include an immune response dynamic and a control is used to investigate potential optimal drug treatment protocols. A summary and conclusion is given in Chapter 5.

1.2 Optimization Techniques

Most students initially learn about a special relation between two sets–usually denoted as an *input* set and *output* set–called a function. Formally, a function $f : X \to Y$ is a rule that assigns to each element x of a set X a unique element y of a set Y. It is well known that functions can be used to characterize a myriad of applications. These applications may range from determining how much revenue a company will earn for a product given a certain price to giving the position of an object on a surface at some time. These (and many more) are standard examples that are usually explored in depth in an introductory algebra class. It is often advantageous to find a particular set of values which may help reveal additional information regarding the behavior of the function (and hence the behavior of the physical phenomenon that the function models, if applicable). These aforementioned values are typically defined to be the *critical values* of a function. Typically, a critical value occurs when the function's derivative, f', is zero or undefined. For example, if we consider a

price-revenue function, the maximum revenue may be a critical value. In the latter example, the point for which the object is at its lowest height on the surface may be at a critical value. Finding these values is such an important concept, that an entire branch of mathematics was built around solving such a problem. Indeed, the process of finding the minimum or maximum value of a function is called *optimization*. In what follows, we detail some of the optimization tools that will be used to help solve the problems developed in this thesis.

1.2.1 Unconstrained and Constrained Minimization

Given an objective function $J : X \subset \mathbb{R}^n \to \mathbb{R}$, unconstrained minimization is the process of finding the element ω^* such that

$$J(\omega^*) \leq J(\omega) \; \forall \; \omega \in X,$$

and is commonly denoted by

$$\min_{\omega} J(\omega).$$

Here ω^* is called the global minimizer of the function J. However, as noted in [61], this problem is considerably more difficult than finding the local minimizer ω^*

$$J(\omega^*) \leq J(\omega) \; \forall \; \omega \text{ near } \omega^*.$$

To quantify the meaning of near, we require the existence of an $\epsilon > 0$ so that $|\omega^* - \omega| < \epsilon$. The main difference between a constrained minimization problem and an unconstrained minimization problem is that the former problem involves conditions that must be enforced on the function's output or input. Thus, instead of using X ,the entire domain of J, a set $\Omega \subset X$ such that the constraints hold is considered. Similarly, the global and local minimizers over Ω are given as the ω^* such that

$$J(\omega^*) \leq J(\omega) \; \forall \; \omega \in \Omega,$$

and

$$J(\omega^*) \leq J(\omega) \; \forall \; \omega \text{ near } \omega^*,$$

respectively. Depending on what is known about J, there are many different methods that can be used to find ω^* for both constrained and unconstrained minimization problems.

In what follows, we consider a broad class of problems known as *nonlinear least squares problems* as the models that will be introduced in the subsequent chapters will not be linear in the states or parameters. Thus, during the model fitting to data, we will be minimizing a weighted sum of

squared errors between our data and model output. To this end, one of our goals is to minimize objective functions of the form

$$J(t,p) = w_i \sum_{i=1}^{N} (y_i - \hat{y}(t_i,p))^2,$$

where $w_i \in \mathbb{R}$ are the weights, $p \in \mathbb{R}^m$ is the vector of m parameters, y_i is the measured data at time, t_i, and $\hat{y}(t_i;p)$ is the model output. The upcoming sections will introduce the methods used in this work following similar notation and theory given in [57, 61] and references therein.

1.2.2 Gradient Based Methods

One popular strategy for locating the local and/or absolute extrema of a function, $f(x)$ for $x \in \mathbb{R}$, involves using derivatives. The derivative of a function, f', exists if and only if f is sufficiently smooth. Recalling that f' can be interpreted as the slope of the tangent line to the curve of f at a given point, we observe that local extrema can only occur at x when $f'(x) = 0$ for functions of this type. After finding these values, the second derivative, f'', is used to determine if that point is a local or absolute extrema. If $f'(x) = 0$ and $f''(x) > 0$ then f has a local minimum at x. Gradient based methods seek to extend this concept to functions of multiple variables so that $x \in \mathbb{R}^n$. The gradient of a function, ∇f, contains the partial derivatives with respect to x_i, the ith component of x. We assume that f is twice continuously differentiable (i.e. $\nabla^2 f(x)$ exist) in order to consider gradient based methods. This is because, as in the one dimensional case, $\nabla f(x) = 0$ and $||\nabla^2 f(x)|| > 0$ is needed to guarantee a local minimum at x. Specific details about many other iterative gradient based optimization methods such as the conjugate gradient method, Newton's method and the method of steepest descent can be found in [23, 40, 57, 61]. In what follows, we provide a comprehensive description of the method that was used in this thesis.

1.2.2.1 Levenberg-Marquardt Method

An important class of gradient based iterative strategies for optimization is the trust region approach. Let $f : \mathbb{R}^n \to \mathbb{R}$ be the considered function that is to be minimized. Given that f can be represented by the quadratic model in (1.2)

$$m_c(x) = f(x_c) + \nabla f(x_c)^T (x - x_c) + \frac{(x - x_c)^T H_c (x - x_c)}{2}, \tag{1.2}$$

in some region, the idea is that (1.2) can be iteratively minimized in varying regions to ultimately determine the minimum of f. Here H_c is the matrix of partial derivatives

$$(\nabla^2 f)_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j},$$

called the function hessian and x_c is the center of a ball, B, known as the *trust region* such that

$$B(r) = \{x \mid \|x - x_c\| \leq r\},$$

with *trust radius*, r. The difference between x_c and the minimum of m_c is s_t and the *trust region problem* is given by

$$\min_{\|s\|<r} m_c(x_c + s). \tag{1.3}$$

At each iteration, the trial step, s_t, or the trial solution, $x_t = x_c + s_t$, is accepted as the solution to (1.3) and determines if the step and/or r should be revised. Let the *actual reduction* in f be given by

$$ared = f(x_c) - f(x_t).$$

The decrease in the quadratic model is given by

$$\begin{aligned} pred = &\quad m_c(x_c) - m_c(x_t), \\ = &\quad f(x_c) + \nabla f(x_c)^T (x_c - x_c) + \frac{(x_c - x_c)^T H_c (x_c - x_c)}{2}, \\ &\quad -(f(x_c) + \nabla f(x_c)^T (x_t - x_c) + \frac{(x_t - x_c)^T H_c (x_t - x_c)}{2}), \\ = &\quad -\nabla f(x_c)^T (x_c + s_t - x_c) - \frac{(x_c + s_t - x_c)^T H_c (x_c + s_t - x_c)}{2}, \\ = &\quad -\nabla f(x_c)^T s_t - \frac{s_t^T H_c s_t}{2}, \end{aligned}$$

and is called the *predicted reduction* such that $pred > 0$ unless $\nabla f(x_c) = 0$. Typically, three control parameters

$$\mu_0 \leq \mu_1 < \mu_2,$$

determine if the trial step or trust region radius should be adjusted. If $\frac{ared}{pred} < \mu_0$, s_t is rejected. If $\frac{ared}{pred} > \mu_2$, then r increases to $\hat{r} = \omega_2 \cdot r$ where $\omega_2 > 1$. If $\frac{ared}{pred} < \mu_1$, r decreases to $\hat{r} = \omega_1 \cdot r$ where $0 < \omega_1 < 1$. Typical values are $\mu_0 = 10^{-4}$, $\mu_1 = .25$, $\mu_2 = .75$, $\omega_1 = .5$, and $\omega_2 = 2$. So that the region

doesn't expand infinitely, a bound for the lengthening of r is set so that

$$r \leq c||\nabla f(x_c)||,$$

for some $c > 0$ that may depend on x_c. Adjustment patterns such as a sufficient decrease in f being the factor that determines modifications to s_t or using $\frac{|pred-ared|}{||\nabla f||}$ instead of $\frac{ared}{pred}$ can be implemented depending on the algorithm. A significant advantage of using the trust region approach is that there is an exact solution to (1.3). In what follows we prove the existence of an exact solution similar to what is given in [61, 96].

Theorem 1.2.1 *Let $g \in \mathbb{R}^n$ and let A be a symmetric $N \times N$ matrix. Let*

$$m(s) = g^T s + \frac{s^T A s}{2}.$$

A vector s is a solution to

$$\min_{||s|| \leq r} m(s), \tag{1.4}$$

if and only if there is $v \geq 0$ such that

$$(A + vI)s = -g,$$

and either $v = 0$ or $||s|| = r$.

Proof 1.2.2 *Consider the equivalent problem to (1.4) given by (1.5):*

$$\min_{||s||^2 \leq r^2} m(s). \tag{1.5}$$

Using the theory of Lagrange multipliers, where

$$\begin{aligned} F(s) &= ||s||^2 - r^2, \\ &= s^T s - r^2, \end{aligned}$$

if s solves (1.5) then it must also solve

$$\begin{aligned} \nabla m(s) &= \bar{\lambda}(\nabla F(s)), \\ g + As &= \bar{\lambda}(2s), \\ (A + \lambda I)s &= -g, \end{aligned} \tag{1.6}$$

subject to

$$\lambda F(s) = 0.$$

Here $\lambda \geq 0$ is a multiple of the Lagrange multiplier associated with $||s||^2 \leq r^2$, $\hat{\lambda}$. If $s \neq 0$ and solves (1.4) then it also solves

$$\min_{||w|| \leq ||s||} m(w), \tag{1.7}$$

because $||w|| \leq ||s|| \leq r$. This means that for any w so that $||w|| = ||s||$, if (1.4) and (1.6) are combined then the following is obtained

$$\begin{aligned} -s^T(A+\lambda I)w + \frac{w^T A w}{2} &\geq -s^T(A+\lambda I)s + \frac{s^T A s}{2}, \\ \implies -s^T A w - \lambda s^T w + \frac{w^T A w}{2} &\geq -s^T A s - \lambda s^T s + \frac{s^T A s}{2}, \\ \implies \frac{w^T A w}{2} - s^T A w - \lambda s^T w + \frac{s^T A s}{2} + \frac{\lambda s^T s}{2} &\geq -\frac{\lambda s^T s}{2}. \end{aligned}$$

The following equalities

$$\begin{aligned} s^T A w &= \frac{s^T A w}{2} + \frac{w^T A s}{2}, \\ \lambda s^T w &= \frac{\lambda s^T w}{2} + \frac{\lambda w^T s}{2}, \end{aligned}$$

are used after adding $\frac{\lambda w^T w}{2}$ to both sides to give the following inequality

$$\frac{w^T A w}{2} - \frac{s^T A w}{2} - \frac{w^T A s}{2} - \frac{\lambda s^T w}{2} - \frac{\lambda w^T s}{2} + \frac{s^T A s}{2} + \frac{\lambda s^T s}{2} + \frac{\lambda w^T w}{2} \geq \frac{\lambda w^T w}{2} - \frac{\lambda s^T s}{2}.$$

Therefore,

$$\frac{1}{2}(w-s)^T(A+\lambda I)(w-s) \geq \frac{\lambda}{2}(w^T w - s^T s) = 0. \tag{1.8}$$

It follows from (1.8) that $A+\lambda I$ is positive semidefinite. If $s = 0$, then $g = 0$ and s solves

$$\min_{||s|| \leq r} \frac{s^T A s}{2},$$

which implies A is positive semidefinite. Therefore, $A+\lambda I$ is always at least positive semidefinite since $\lambda \geq 0$. If s solves (1.6) and (1.7) then for any w

$$\begin{aligned} g^T w + \frac{w^T(A+\lambda I)w}{2} &\geq g^T s + \frac{s^T(A+\lambda I)s}{2} \\ \implies m(w) &\geq m(s) + \frac{\lambda}{2}(s^T s - w^T w). \end{aligned} \tag{1.9}$$

The following are consequences acquired immediately from (1.9):

1. *If* $\lambda = 0$ *and* $||s|| \leq r$ *then* s *solves (1.4).*

2. *If* $||s|| = r$ *then* s *solves*

$$m(s) = \min_{||w||=r} m(w).$$

3. *If* $\lambda \geq 0$ *and* $||s|| = r$ *then* s *solves (1.4).*

This completes the proof. ∎

When positive definiteness is obtained for $(A+\lambda I)$ and $w \neq s$ then the inequality is strict in (1.9) and

$$s = -(H_c + \lambda I)^{-1} g,$$

is the exact solution to (1.3). Levenberg-Marquardt algorithm [65, 69] takes advantage of adjusting λ instead of r with respect to $\frac{ared}{pred}$ in their trust region based algorithm. Variations of this method can be found in [57, 96] but this work focuses on the algorithm as described in [61]. The Levenberg-Marquardt quadratic model with parameter λ_c at the point x_c is

$$m(x) = f(x_c) + (x - x_c)^T R'(x_c)^T R(x_c) + \frac{1}{2}(x - x_c)^T (R'(x_c)^T R'(x_c) + \lambda_c I)(x - x_c),$$

using the least squares objective function given by

$$f(x) = \frac{1}{2}\sum_{i=1}^{M} ||r_i(x)||_2^2 = \frac{1}{2} R(x)^T R(x).$$

The minimizer of $m(x)$ is given by

$$x_t = x_c - (R'(x_c)^T R'(x_c) + \lambda_c I)^{-1} R'(x_c)^T R(x_c), \tag{1.10}$$

with step $s = x_t - x_c$. The predicted reduction is then

$$\begin{aligned} pred &= m(x_c) - m(x_t), \\ &= -s^T R'(x_c)^T R(x_c) - \frac{1}{2} s^T (R'(x_c)^T R'(x_c) + \lambda_c I) s, \\ &= -s^T R'(x_c)^T R(x_c) + \frac{1}{2} s^T R'(x_c)^T R(x_c) \text{ using (1.10)}, \\ &= -\frac{1}{2} s^T \nabla f(x_c). \end{aligned}$$

This means we will accept/reject our trial point, x_t, and Levenberg-Marquardt parameter, λ_c, based on the ratio

$$\begin{aligned}\frac{ared}{pred} &= \frac{f(x_c)-f(x_t)}{m(x_c)-m(x_t)},\\ &= -2\frac{f(x_c)-f(x_t)}{s^T\nabla f(x_c)}.\end{aligned}$$

Adjusting λ is opposite to how we adjusted r. If $\frac{ared}{pred}$ is large, the actual reduction in the function could be large. Thus, s should be large in order to go further in this direction, so λ is decreased so that the term $(R'(x_c)^T R'(x_c)+\lambda_c I)^{-1}R'(x_c)^T R(x_c)$ is larger in (1.10). By similar reasoning, if $\frac{ared}{pred}$ is small, λ increases. This method converges q-quadratically [61] in the following sense

Definition Let $\{x_n\}\subset\mathbb{R}^n$ and $x^*\in\mathbb{R}^n$. $x_n\to x^*$ q-quadratically if $x_n\to x^*$ and there is $K>0$ such that

$$||x_{n+1}-x^*||\leq K||x_n-x^*||^2.$$

We use the MATLAB implementation in the package *nlinfit* of algorithm (1.1).

Algorithm 1.1 lmalg(x,R,kmax)

1. Set $\lambda=\lambda_0$.
2. For $k=1,...,kmax$
   (a) Let $x_c=x$.
   (b) Compute R, f, R', and ∇f; test for termination.
   (c) Compute x_t using (1.10).
   (d) Call *lmfunc*(x_c,x_t,x,f,λ) (see algorithm 1.2)

1.2.3 Non-Gradient Based Methods

If the function of interest is non-smooth, discontinuous, based on a stochastic model, or if the gradient is too difficult to compute, then the optimization methods discussed to this point may not be useful or practical. In this case there are methods that don't depend on the gradient or the Hessian of the objective function. Instead, the method evaluates the objective function at different points and predicts where to search next to find the minimum. Global convergence is difficult to guarantee

Algorithm 1.2 lmfunc$(x_c, x_t, x_+, f, \lambda)$

1. $z = x_c$
2. Do while $z = x_c$
   (a) $ared = f(x_c) - f(x_t), s_t = x_t - x_c, pred = \frac{-\nabla f(x_c)^T s_t}{2}$.
   (b) If $\frac{ared}{pred} < \mu_0$ then set $z = x_c$, $\lambda = \max(\omega_2\lambda, \lambda_0)$, and recompute the trial point with the new value of λ.
   (c) If $\mu_0 \leq \frac{ared}{pred} < \mu_1$, then set $z = x_t$ and $\lambda = \max(\omega_2\lambda, \lambda_0)$.
   (d) If $\mu_1 \leq \frac{ared}{pred}$, then set $z = x_t$.
   If $\mu_2 < \frac{ared}{pred}$, then set $\lambda = \omega_1\lambda$.
   If $\lambda < \lambda_0$, then set $\lambda = 0$.
3. $x_+ = z$

with no knowledge of the gradient; however, some convergence criterion have been proven for specific functions in lower dimensions [31, 64]. An interesting class of these methods are called genetic algorithms. They are adaptive heuristic search algorithms that at each iteration mimics the principles of natural selection and genetics. Popularly used for mixed integer optimization, they use the initial population as “parents" and use them to give “offspring" based on their fitness score to move towards a minimum. The chief operators for these types of algorithms use probabilistic rather than deterministic transitions and are given in the following:

1. **Reproduction** - determines which of the parents are going to survive to the next generation.
2. **Crossover** - determines how the parents will be combined to make offspring.
3. **Mutation** - determines how changes will be made to parents to create offspring.

We refer the reader to [38, 48] and references therein for a more detailed description. The class of non-gradient based methods that are employed in this work are called deterministic sampling methods.

1.2.3.1 Deterministic Sampling Methods

Sampling methods do not require knowledge of the objective function’s gradient. The method will sample points in the domain of the function, then use that information to determine where to

search next for the optimal value of the objective function. Deterministic sampling methods use patterns to optimally guide the search [103]. There are a myriad of algorithms that use deterministic sampling like the Hooke-Jeeves algorithm and the multidirectional search method described in [61]. We describe two specific methods that are used in this research.

1.2.3.1.1 Nelder-Mead Algorithm

The Nelder-Mead algorithm is a non-gradient based method that is simplex based [61, 80]. Let $J:\mathbb{R}^n \to \mathbb{R}$ and $S=\{\lambda_0,\lambda_1,\cdots,\lambda_n\}$ be a simplex of $n+1$ points with $\lambda_i \in \mathbb{R}^n$. We let

$$J(\lambda_l)=\min_{\lambda\in S} J(\lambda),$$

and

$$J(\lambda_u)=\max_{\lambda\in S} J(\lambda).$$

By adjusting the point in the simplex that gives the value farthest from the objective, we are able to find the minimum over the space. Figure 1.1 gives a sketch of the simplex points $(\lambda_1,\lambda_2,\lambda_3)$ in two dimensions as well as the points considered to revise the simplex which are given by

- extension point (e),
- reflection point (r),
- outer contraction point (oc), and
- inner contraction point (ic).

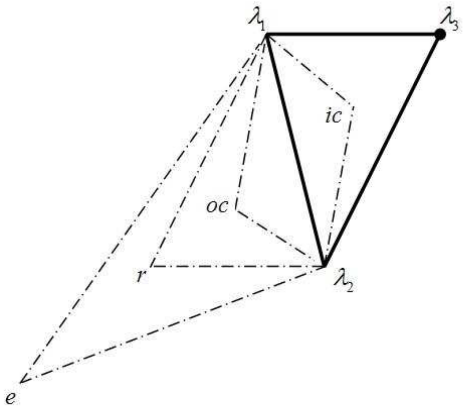

Figure 1.1 A simplex and points considered during the Nelder-Mead algorithm from [37].

Algorithm (1.3), as obtained from [37, 66, 74], gives a brief overview of how the Nelder-Mead algorithm is implemented. The tolerance between values of the functions and total number of iterations are among options for stopping criterion.

Figure 1.2 shows the first 11 iterations of the Nelder-Mead algorithm converging to the local minimum for a function of two variables.

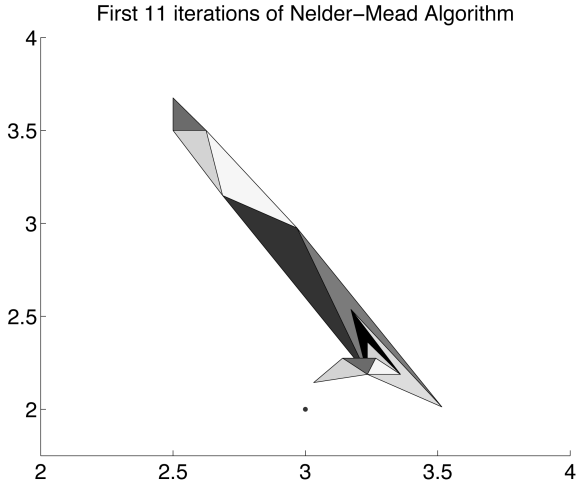

Figure 1.2 A graph of the different simplexes associated with the implementation of the Nelder-Mead algorithm where each triangle is a simplex and the minimum at $p = (3, 2)$.

Algorithm 1.3

1. Not including λ_u, compute the centroid of the simplex, c,

$$c = \frac{\sum_{i=0, i\neq u}^{n} \lambda_i}{n}.$$

2. Using $\alpha = 1$ as the step size in the direction (in relevance to the centroid) that is opposite to the direction of λ_u, calculate the reflection point $r = c + \alpha(c - \lambda_u)$. If $J(\lambda_l) \leq J(r) < J(\lambda_k)$ where $J(\lambda_k)$ is the second lowest objective function value then λ_u is replaced with r and restart.

3. Compute the extension point $e = c + \alpha(c - s_u)$ where $\alpha = 2$ if $J(r) < J(\lambda_l)$ to expand the search further than r. If $J(e) < J(r)$, λ_u is replaced with e and the algorithm is restarted. If not, then the simplex is not expanded and λ_u is replaced with r and restart.

4. If the above is not true then check if $J(r) \geq J(\lambda_i)$ for $i \neq u$. For this case, one of the following points are considered:

   (a) If $J(r) < J(\lambda_u)$ then the outer contraction point, $oc = c + \alpha(r - c)$, is computed where $\alpha = \frac{1}{2}$. If $J(oc) < J(r)$ then replace λ_u with oc and restart.

   (b) If $J(r) \geq J(\lambda_u)$ then the inner contraction point, $ic = c + \alpha(\lambda_u - c)$, is computed where $\alpha = \frac{1}{2}$. If $J(ic) < J(\lambda_u)$ then replace λ_u with ic and restart.

5. If the above points are not used, then let $\lambda_i = \frac{\lambda_i + \lambda_l}{2}$ for each i and restart from step 1.

Convergence is only guaranteed in one dimension for strictly convex functions with bounded level sets [64]. In MATLAB, the optimization package *fminsearch* provides a numerical implementation of the algorithm.

1.2.3.1.2 Implicit Filtering

Implicit filtering is a non-gradient based optimization routine that estimates the gradient using difference approximations [61]. The size of the increment in the difference varies as iterations progress. This is to filter out initial oscillations and circumvent local minima. Algorithm (1.4) gives a simplified version of the implicit filtering routine as described in [66]. The implicit filtering iterative

Algorithm 1.4

1. We start with the current minimum of J being at s_c and initial step size h_c. A $2n$ dimensional stencil is created about s_c and is given by

$$S(s_c, h_c) = \{s_c \pm h_c e_i\},$$

   where e_i are unit vectors.

2. We then evaluate J at all points in S and let

$$J(s_u) = \min\{J(z) | z \in S(s_c, h_c)\}.$$

3. Set $s_0 = s_c$ and let

$$s_c = s_0 - \lambda \nabla_{h_c} J(s_0),$$

   if $J(s_u) < J(s_c)$ where λ assures enough decrease such that

$$J(s_c - \lambda d) < J(s_c) \text{ where } d = -\nabla J(s_c),$$

   and

$$(\nabla_{h_c} J(x))_i = \frac{J(x + h_c e_i) - J(x - h_c e_i)}{2h_c}.$$

4. If the above is not true and $J(s_u) \geq J(s_c)$ then h_c is reduced and restart from step 1.

process will stop when the step size h_c falls below a user specified value.

Figure 1.3 depicts a function where there are many local minima close together and the first 7 iterations of the implicit filtering algorithm trying to find the global minimum. It is noted that the algorithm initially uses a large step size h_c to avoid the many local minima.

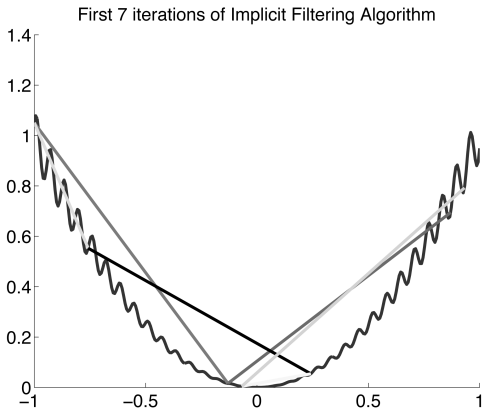

Figure 1.3 The implicit filtering iterations are able to skip many local minima.

In [31], the authors proved that if the objective function is smooth with low amplitude noise and the noise decays rapidly near the minimizer sufficiently, then superlinear convergence is established in the following sense

Definition Let $\{x_n\} \subset \mathbb{R}^n$ and $x^* \in \mathbb{R}^n$. $x_n \to x^*$ q-superlinearly if

$$\lim_{n\to\infty} \frac{||x_{n+1} - x^*||}{||x_n - x^*||} = 0.$$

1.3 Modeling Techniques and Validation

In Chapter 3, a system of nonlinear ordinary differential equations(ODEs) is used to model a biological phenomena. This means there are parameters in the model that are used to describe particular aspects of the system. Thus, emphasis should be placed in obtaining accurate estimates of these parameters while fitting model output to clinical data. These estimates can help determine the robustness and capabilities of the model in solving the *forward problem*. The *forward problem* refers to using a model to predict the future behavior of a system given a set of parameters. The *inverse problem* is the parameterization of a model from empirical data [5, 34, 110]. Accuracy of

the parameters directly determines the accuracy of the model output. However, depending on the complexity of the model, providing accurate parameters can be challenging or impossible. There has been extensive study about parameter selection while solving the inverse problem for biological models and other applications that can be found in [5, 9, 11, 13, 14] and references therein. In what follows, we provide background for the techniques used to calibrate the parameters for the models used in this thesis.

1.3.1 Sensitivity Analysis

A sensitivity analysis is the process of understanding how the model output is affected by changes in the parameters. Sensitivity analyses are used in many branches of mathematics such as statistics, PDEs (partial differential equations), and control design [10, 106]. The parameters that give the most change in the output are said to be sensitive parameters. This is important in the *forward problem* because it allows an understanding of which parameters will give useful information. Once the parameters have been identified, a sensitivity analysis for the *inverse problem* is usually performed to determine the sensitive parameters. Parameters with minimal impact are fixed from literature. There are two different types of sensitivity analysis: global and local. A global sensitivity analysis heavily depends on the structure of the model and quantifies how uncertainties in outputs can be apportioned to uncertainties in inputs. We refer the reader to [94] for more information. Our work uses a local sensitivity analysis which depends on the prescribed value of the parameters.

1.3.1.1 Sensitivity Equations

The sensitivity analysis presented in this section uses a derivative-based approach. Consider the general form of an ODE model and a function z of its output

$$\begin{aligned} \frac{dy}{dt} &= f(t, y; q), \\ z &= g(t, y; q), \end{aligned} \tag{1.11}$$

whereby the vectors y and q contain the variables and parameters of the model, respectively. Since we are concerned with how our model output, z, is influenced by changes to our parameters, q, then we consider the partial derivative of z, $\frac{\partial z}{\partial q}$, with respect to q. One approach to computing this

partial derivative is by solving the associated sensitivity equations. Consider the formulation

$$\begin{aligned}\frac{\partial z}{\partial q} &= \frac{\partial g}{\partial t}\frac{\partial t}{\partial q} + \frac{\partial g}{\partial y}\frac{\partial y}{\partial q} + \frac{\partial g}{\partial q}, \frac{\partial q}{\partial q} \\ &= \frac{\partial g}{\partial y}\frac{\partial y}{\partial q} + \frac{\partial g}{\partial q},\end{aligned} \tag{1.12}$$

since $\frac{\partial t}{\partial q} = 0$ and $\frac{\partial q}{\partial q} = 1$. The two components $\frac{\partial g}{\partial y}$ and $\frac{\partial g}{\partial q}$ can be directly calculated from g, but can be cumbersome to do by hand depending on the complexity of the function. Thus, one can employ automatic differentiation to evaluate these derivatives. Since any mathematical function can be decomposed into elementary functions, automatic differentiation numerically implements the chain rule and basic arithmetic equations repeatedly to compute the total derivative of a function with accuracy to working machine precision [25]. This is achieved with table lookups and tabulating all of the functional compositions [51, 73]. An automatic differentiation(AD) code developed by Martin Fink in MATLAB is employed. Finally, to calculate $\frac{\partial y}{\partial q}$, it is noted that y is continuous in t and q. Since $\frac{\partial y}{\partial q}$ exists, by computing the partial derivative with respect to q of the state equations and reversing the order of differentiation [100] the following is obtained

$$\begin{aligned}\frac{\partial}{\partial q}\left(\frac{dy}{dt}\right) = \frac{d}{dt}\left(\frac{\partial y}{\partial q}\right) &= \frac{\partial f}{\partial t}\frac{\partial t}{\partial q} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial q} + \frac{\partial f}{\partial q}\frac{\partial q}{\partial q} \\ &= \frac{\partial f}{\partial y}\frac{\partial y}{\partial q} + \frac{\partial f}{\partial q}.\end{aligned} \tag{1.13}$$

Similar to $\frac{\partial g}{\partial y}$ and $\frac{\partial g}{\partial q}$, $\frac{\partial f}{\partial y}$ and $\frac{\partial f}{\partial q}$ is calculated using automatic differentiation. From (1.13), the sensitivity equations are given as follows

$$\begin{aligned}\frac{dy}{dt} &= f(t, y; q), \\ \frac{d}{dt}\left(\frac{\partial y}{\partial q}\right) &= \frac{\partial f}{\partial q}\frac{\partial y}{\partial q} + \frac{\partial f}{\partial q}.\end{aligned} \tag{1.14}$$

Solving the sensitivity equations calculates $\frac{\partial y}{\partial q}$ to determine $\frac{\partial z}{\partial q}$.

1.3.1.2 Finite Difference Approximation

A direct approach to finding $\frac{dz}{dq_i}$ is by a finite difference approximation where q_i is the ith component of q. Recall by the definition of a derivative,

$$\frac{dz}{dq_i} = \lim_{h\to 0} \frac{z(q_i+h)-z(q_i)}{h}.$$

Thus, an approximation for the first derivative is the forward-difference formula given by

$$\frac{dz}{dq_i} \approx \frac{z(q_i+h)-z(q_i)}{h},$$

with step size, h, and $\mathcal{O}(h)$ truncation error so it is accurate to first order. Another approximation for the first derivative is the backwards-difference formula given by

$$\frac{dz}{dq_i} \approx \frac{z(q_i)-z(q_i-h)}{h},$$

that also has $\mathcal{O}(h)$ truncation error. Adding the previous two formulas together yields the centered difference formula

$$\begin{aligned} 2\frac{dz}{dq_i} &\approx \frac{z(q_i+h)-z(q_i)+z(q_i)-z(q_i-h)}{h}, \\ \implies \frac{dz}{dq_i} &\approx \frac{z(q_i+h)-z(q_i-h)}{2h}, \end{aligned} \tag{1.15}$$

and has $\mathcal{O}(h^2)$ truncation error so it is accurate to second order. The step size should be chosen to minimize truncation error and subtractive cancellation error due to finite precision arithmetic. In [57], it is shown that a good candidate for the step size in the backwards and forward differences is $h = \sqrt{\text{macheps}} \cdot q_i$ and $h = (\text{macheps})^{\frac{1}{3}} \cdot q_i$ for central difference where macheps is accuracy at which the function z is evaluated. Therefore, to obtain an accurate approximation to the finite difference of the derivative of z, z must be evaluated at high precision. A routine where subtractive cancellation error does not occur is preferable and is presented in the next section.

1.3.1.3 Complex Step Method

This method utilizes a procedure that uses complex functions to calculate first derivatives as presented in [70]. The technique takes advantage of how complex functions are extensions of their real counterparts, the analyticity of complex functions and the definition of a derivative to devise an algorithm to determine the first derivative of a function. Assume that $z = x + iy$ where $\text{Re}(z) = x$

and $\mathrm{Im}(z) = y$ such that

$$f(z) = u(x + iy) + iv(x + iy)$$

is an analytic complex function where u and v are the real and imaginary parts of f, respectively. Since f is analytic, it satisfies the Cauchy-Riemann equations. That is, the components satisfy

$$u_x = v_y,\ u_y = -v_x. \tag{1.16}$$

The forward difference approximation is used to rewrite the first equation in (1.16) as

$$u_x \approx \frac{v(x + i(y + h)) - v(x + iy)}{h}.$$

If the function is restricted to the real axis then the following are true:

$$\begin{aligned} y &= 0, \\ v(x) &= 0, \\ f(x) &= u(x). \end{aligned} \tag{1.17}$$

This implies that

$$\begin{aligned} \frac{df}{dx} &= u_x, \\ &\approx \frac{v(x + ih) - v(x)}{h}, \\ &= \frac{v(x + ih)}{h}. \end{aligned} \tag{1.18}$$

Therefore, an approximation of the first derivative of f at a given parameter x is

$$\frac{df}{dx} \approx \frac{\mathrm{Im}(f(x + ih))}{h},$$

and is called the complex step derivative approximation. This has an obvious advantage over the finite difference approximation because there is no subtraction operation and thus has no subtractive cancellation error. The benefit over automatic differentiation is that it is usually quicker and uses less memory since it doesn't take multiple function evaluations. We implement this method in MATLAB by evaluating $f(x + ih)$ and recovering the imaginary component of the output over h.

The error for the complex step method can be determined using a Taylor series expansion with a pure imaginary step, ih. Given that f is analytic and a real function in real variables, the following

Taylor series expansion is obtained about a real point x,

$$f(x+ih)=f(x)+ihf'(x)-h^2\frac{f''(x)}{2!}-ih^3\frac{f'''(x)}{3!}+...$$

By considering the imaginary parts of both sides and dividing by h we obtain

$$f'(x)=\frac{\text{Im}(f(x+ih))}{h}+h^2\frac{f'''(x)}{3!}+...,$$

and therefore have $\mathcal{O}(h^2)$ error. Since there is no subtractive cancellation, h can be reduced to very small values to achieve higher accuracy in the derivative.

As an exercise, a comparison of the sensitivities given by the complex step method, sensitivity equations with automatic differentiation, and finite difference is executed at different values of h. To this end, consider the logistic growth model with the Verhulst-Pearl logistic equation studied in [10, 12]

$$\begin{aligned}\frac{dx}{dt}&=rx(1-\frac{x}{K}),\\ x(0)&=x_0,\end{aligned}\tag{1.19}$$

where K is the carrying capacity and r is the intrinsic growth rate. The exact solution for (1.19) is given by

$$x(t)=\frac{K}{1+(\frac{K}{x_0}-1)e^{-rt}}.$$

The graph in Figure 1.4 shows the solution to (1.19) with specific parameters. The parameters in this model are r, K, and x_0 and their impact on the output x is of interest. Let $x_r(t)=\frac{dx}{dr}$, $x_K(t)=\frac{dx}{dK}$, and $x_{x_0}(t)=\frac{dx}{dx_0}$. We can directly calculate the sensitivity equations using (1.14) as a guide to obtain

$$\begin{aligned}\frac{dx}{dt}&=rx(1-\frac{x}{K}),\\ \frac{dx_r}{dt}&=(r-\frac{2r}{K}x(t))x_r+x(t)-\frac{1}{K}x^2(t),\\ \frac{dx_K}{dt}&=(r-\frac{2r}{K}x(t))x_K+\frac{r}{K^2}x^2(t),\\ \frac{dx_{x_0}}{dt}&=(r-\frac{2r}{K}x(t))x_{x_0},\end{aligned}\tag{1.20}$$

and initial conditions

$$x(0) = x_0,$$
$$x_r(0) = 0,$$
$$x_K(0) = 0,$$
$$x_{x_0}(0) = 1.$$

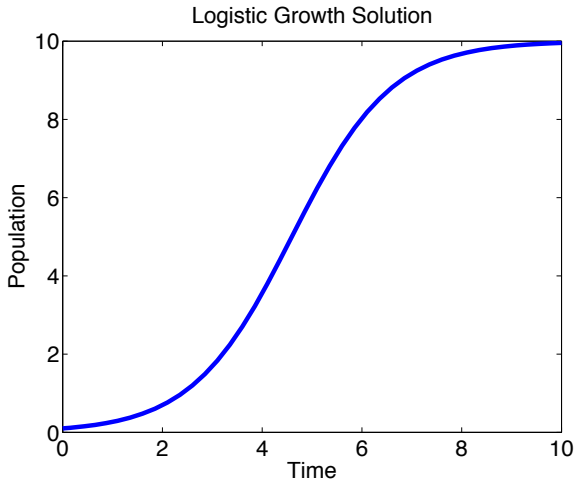

Figure 1.4 Solution to (1.19) with $K = 10$, $r = 1$ and $x_0 = .1$.

The graphs on the left in Figure 1.5 and Figure 1.6 have the sensitivities using (1.4) and automatic differentiation (blue solid line), central difference (red dotted line), and complex step method (black dashed line). The graphs on the right are the errors between the sensitivity equations (SE) and finite difference (blue solid line), finite difference (FD) and complex step method (red dotted line), and complex step method (CM) and sensitivity equations (black dashed line). It is observed in Figure 1.5 that at $h = 1 \times 10^{-10}$ all 3 methods are nearly the same with small error. However, as h gets smaller in Figure 1.6, the central difference method becomes inaccurate, but the complex step method continues to be precise for $h = 1 \times 10^{-200}$. In fact, with machine accuracy 1×10^{-323}, the complex step method is still accurate at $h = 1 \times 10^{-320}$.

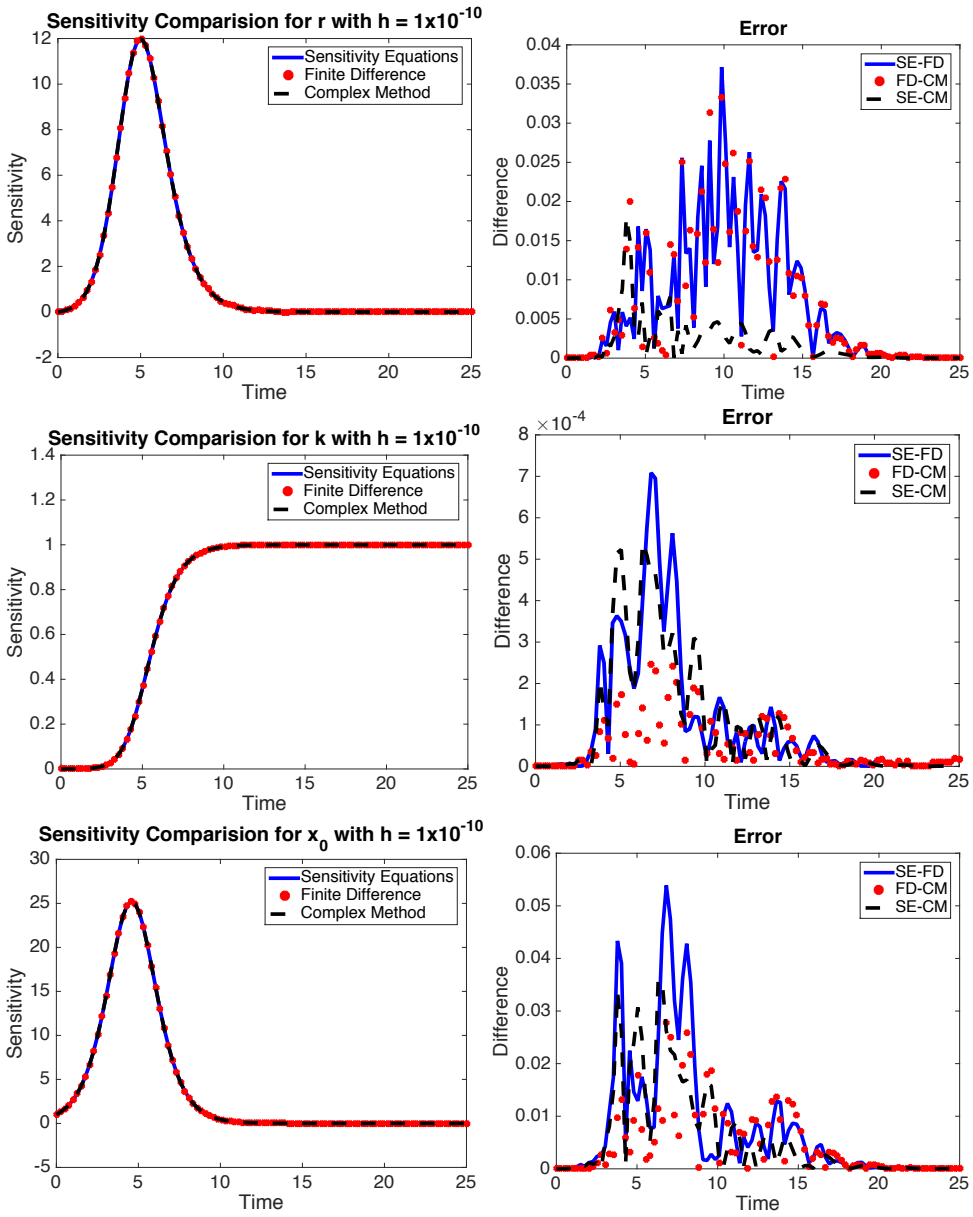

Figure 1.5 Graphs show a comparison of sensitivities using the sensitivity equations with automatic differentiation, central difference, and complex step methods with with $h = 1 \times 10^{-10}$.

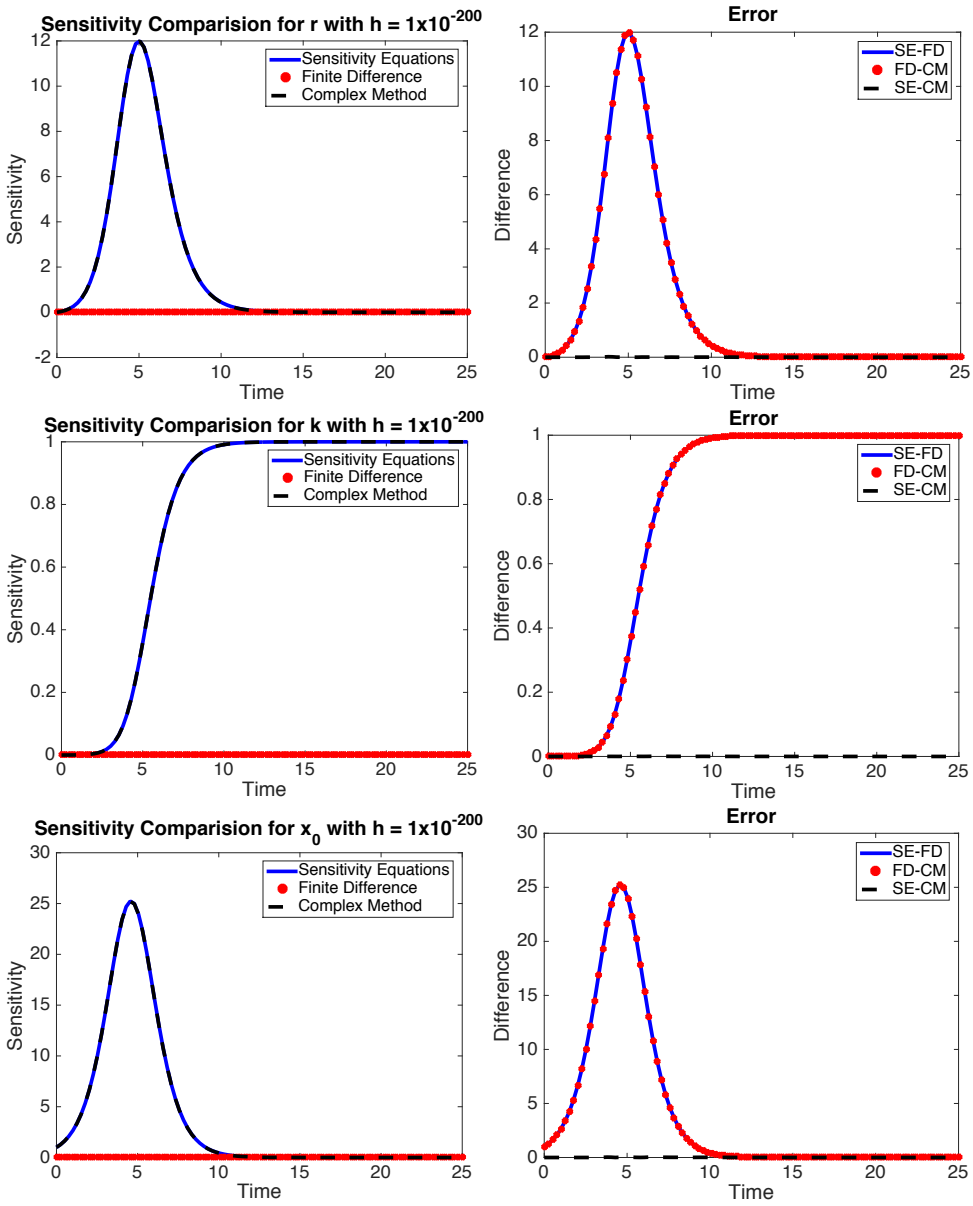

Figure 1.6 Graphs show a comparison of sensitivities using the sensitivity equations with automatic differentiation, central difference, and complex step methods with with $h = 1 \times 10^{-200}$.

1.3.2 Identifiability Analysis

After deciding which parameters are sensitive, consideration is given to understanding which sensitive parameters can uniquely be identified from the data. The structure of the model as well as amount of data can affect whether a parameter is identifiable. A parameter q is locally identifiable if for an open neighborhood about q in the parameter space, $y(q_1) = y(q_2)$ is true implies that $q_1 = q_2$ [71]. We illustrate how model structure can affect identifiability by considering the parameters a and b within the simple ODE:

$$\frac{dx}{dt} = abx. \tag{1.21}$$

We observe that a and b are unidentifiable as there are many possible values for a and b that result in the same product ab. $a = 2$ and $b = 1$ results in the same solution to (1.21) as $a = 1$ and $b = 2$. Thus, estimating the parameters in this model is futile because of the lack of uniqueness. An identifiability analysis will aid us in deciding which parameters can be uniquely estimated from the experimental data as it is desirable to estimate parameters that are both sensitive and identifiable. Consider the parameters contained in q which minimize the cost function

$$J(q) = \frac{1}{N}\sum_{i=1}^{N}(V_d^i - V(t_i;q))^2,$$

with $V(t_i;q)$ denoting the model output and V_d^i denoting the corresponding data value at time point t_i for $i = 1,\dots N$, where N is the number of data values. Assume that q^* is the minimum of this cost function. Then by using a Taylor series expansion around q^*, we obtain

$$V(t_i,q) = V(t_i;q^*) + \frac{dV(t_i;q^*)}{dq}(q-q^*) + \dots$$

If we only consider the first two elements of $V(t_i,q)$ under the assumption that $q \approx q^*$ and substitute this expression into the cost function we find that

$$\begin{aligned} J(q) &= \frac{1}{N}\sum_{i=1}^{N}\left(V_d^i - V(t_i;q^*) - \frac{dV(t_i;q^*)}{dq}(q-q^*)\right)^2, \\ &= \frac{1}{N}\sum_{i=1}^{N}\left(\frac{dV(t_i;q^*)}{dq}(q-q^*)\right)^2, \end{aligned} \tag{1.22}$$

where we used the fact that q^* is the minimum of the cost function so that $V_d^i \approx V(t_i;q^*)$. Let

$$S=\frac{dV}{dq}=\begin{bmatrix} \frac{dV}{dq_1}(t_1) & \frac{dV}{dq_2}(t_1) & \cdots & \frac{dV}{dq_l}(t_1) \\ \frac{dV}{dq_1}(t_2) & \frac{dV}{dq_2}(t_2) & \cdots & \frac{dV}{dq_l}(t_2) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{dV}{dq_1}(t_N) & \frac{dV}{dq_2}(t_N) & \cdots & \frac{dV}{dq_l}(t_N) \end{bmatrix}, \tag{1.23}$$

be a $(N\times l)$ sensitivity matrix relating to the sensitivities $\frac{dV}{dq_j}(t_i)$ of the output with $i=1,\ldots,N$ and $j=1,\ldots,l$, where l denotes the number of parameters. The cost function of (1.22) is rewritten in terms of this sensitivity matrix

$$\begin{aligned} J(q)&=\frac{1}{N}(S(q-q^*))^T(S(q-q^*)),\\ &=\frac{1}{N}(S\Delta q)^T(S\Delta q), \end{aligned}$$

where $\Delta q=q-q^*$. Rearranging $\Delta q=q-q^*$, we formulate the cost function in terms of $q^*+\Delta q$:

$$J(q^*+\Delta q)=\frac{1}{N}\Delta q^T S^T S\Delta q. \tag{1.24}$$

If we suppose that Δq is an eigenvector of S^TS with $S^TS\Delta q=\lambda\Delta q$, then we have

$$\begin{aligned} J(q^*+\Delta q)&=\frac{1}{N}\Delta q^T(\lambda\Delta q),\\ &=\frac{1}{N}\lambda||\Delta q||_2^2. \end{aligned}$$

We note that if Δq is an eigenvector with eigenvalue $\lambda=0$, then the cost function to second order approximation is $J(q^*+\Delta q)=0$. The least squares cost function does not change values when moving from q^* to $q^*+g\Delta q$, with g arbitrary. Thus, the parameters are locally unidentifiable at q^*. If S^TS has very small eigenvalues, this can also be a problem for parameter identification. There has been studies about how the Fisher Matrix(S^TS) information can be used for parameter identification [21, 24, 33, 81]. In [33], they search all possible parameter combinations and choose them based on the rank of the sensitivity matrix, S, and asymptotic standard error uncertainty. We use algorithm (1.5) as described in [81] to help us decide which of the parameters in our model will be unidentifiable. After performing this procedure, we now have a set of sensitive and locally identifiable parameters to estimate. The rest of the parameters are set to "typical values" found in literature.

Algorithm 1.5

1. Create the matrix $S^T S$, compute its eigenvalues, and order them such that

$$|\lambda_1| \leq |\lambda_2| \leq \cdots \leq |\lambda_n|.$$

2. If $|\lambda_1|$ is less than some threshold ϵ (typically taken to be 10^{-4}), we say that there is a parameter that is unidentifiable.

3. The largest magnitude component of the eigenvector Δq_1 associated with the eigenvalue λ_1 corresponds to the least identifiable parameter. Remove the corresponding column from S and repeat step 1.

1.3.3 Data Analysis

Occasionally, there is data that one cannot observe. For instance in infectious diseases, due to different patient responsiveness to drug treatment, there is viral load data that we are not able to examine with the technology used. Specifically, the exact data measurement is below what is called the lower limit of quantification(LLOQ). This data is considered to be left-censored. Censored data is sometimes excluded in model calibration which can result in inaccurate parameter estimates. The importance of understanding the data below LLOQ can be seen in an example of a Breakthrough patient's viral load shown in Figure 1.7.

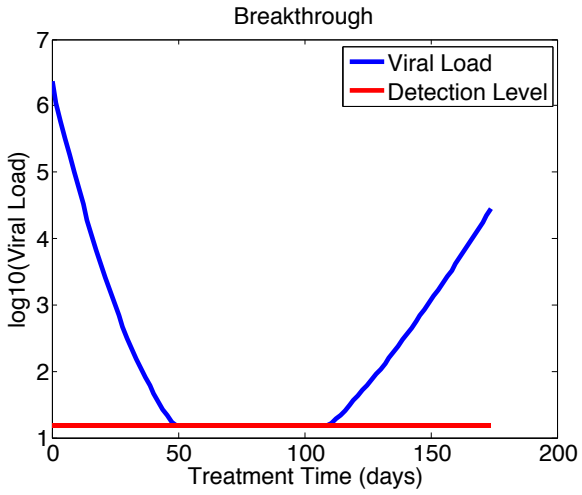

Figure 1.7 Viral load pattern for a Breakthrough patient.

The viral load in Breakthrough patients goes below LLOQ (detection level) and rises back up during treatment. This patient behavior will not be accurately modeled without robust predictions for what is occurring beneath the censoring line. Thus, when there is censored data, expectation maximization(EM) [39] is used to compute maximum likelihood estimates for the parameters, q. A derivation of the EM algorithm that is employed is provided and uses similar notation and techniques as in [7]. Consider the general objective function

$$q^* = \arg\min J(q) = \frac{1}{N}\sum_{i=1}^{N} |y_d^i - y(t_i, q)|^2, \tag{1.25}$$

where N is the number of data points, y_d^i is the data at time t_i, and y is the model output over an admissible parameter space $\omega \subset \mathbb{R}^p$ such that p is the number of parameters being estimated. Since, in general, the model output, data, and parameters have a large range of values, they are $\log_{10}$ scaled. Assume for true parameter q^0 and variance σ^2 that the log scaled data is normally distributed such that

$$y_d^i \sim \mathcal{N}(y(t_i, q^0), \sigma^2).$$

Let $y_i = y(t_i, q)$ and $L = \log(LLOQ)$. The data can be written as the piecewise function

$$d^i = \begin{cases} y_d^i & \text{if} y_d^i > L \\ L & \text{if} y_d^i \leq L \end{cases},$$

with

$$X^i = I_{\{y_d^i > L\}},$$

where $I_{\{y_d^i > L\}}$ is an indicator function such that if $y_d^i > L$ at t_i then $X^i = 1$, otherwise $X^i = 0$. Before presenting the likelihood function to be maximized, standard probability tools are introduced. The standard probability density function(pdf) for mean 0 and unit variance is given by

$$\varphi(\xi) = \frac{1}{\sqrt{2\pi}} e^{\frac{-\xi^2}{2}},$$

which gives the probability that a continuous random variable has the value ξ. The pdf has corresponding cumulative distribution function(cdf),

$$\Phi(\xi) = \int_{-\infty}^{\xi} \varphi(s) ds,$$

which gives the probability of a continuous random variable being less than or equal to ξ. Since the censored data is bounded above, it is assumed to follow a truncated normal distribution so that the likelihood function for a sample point d_i is given by

$$\overline{M}(q,\sigma)=\prod_{i=1}^{N}\Big[\frac{1}{\sigma}\varphi\Big(\frac{d^i-y_i}{\sigma}\Big)\Big]^{X^i}\Big[\Phi\Big(\frac{d^i-y_i}{\sigma}\Big)\Big]^{1-X^i}, \tag{1.26}$$

where the first term accounts for the probability of observing d^i if it is uncensored and the second term accounts for the probability of observing d^i if it is censored. Computing $\log(\overline{M})$ gives the log-likelihood function

$$M(q,\sigma)=\sum_{i=1}^{N}\Big[X^i\Big[\log\varphi\Big(\frac{d^i-y_i}{\sigma}\Big)-\log\sigma\Big]+\big(1-X^i\big)\Big[\Phi\Big(\frac{d^i-y_i}{\sigma}\Big)\Big]. \tag{1.27}$$

Since in the first term of (1.26), $d_i=y_d^i$, and in the second term, $d_i=L$, (1.27) is rewritten as

$$M(q,\sigma)=\sum_{i=1}^{N}\Big[X^i\Big[\log\varphi\Big(\frac{y_d^i-y_i}{\sigma}\Big)-\log\sigma\Big]+\big(1-X^i\big)\Big[\Phi\Big(\frac{L-y_i}{\sigma}\Big)\Big]. \tag{1.28}$$

EM is used to maximize M by iteratively improving q and σ until the maximum is achieved. A lower truncated normal distribution is considered so the following is true:

$$E[y_d^i|y_d^i\le L]=y_i-\sigma\Lambda(\xi^i),$$

where $\xi^i=\frac{L-y_i}{\sigma}$ and $\Lambda(\xi^i)=\frac{\varphi(\xi^i)}{\Phi(\xi^i)}$. The following theorem from [50] gives an important result about truncated normal distributions.

Theorem 1.3.1 *If $x\sim\mathcal{N}[\mu,\sigma^2]$ and a is a constant, then*

$$E[x|truncation]=\mu+\sigma\lambda(\alpha),$$
$$Var[x|truncation]=\sigma^2[1-\delta(\alpha)],$$

where $\alpha=\frac{a-\mu}{\sigma}$ such that

$$\lambda(\alpha)=\frac{\varphi(\alpha)}{1-\Phi(\alpha)} \quad \text{if truncation is } x>a,$$
$$\lambda(\alpha)=\frac{\varphi(\alpha)}{\Phi(\alpha)} \quad \text{if truncation is } x<a.$$

Here $\varphi(\alpha)$ is the standard normal density with $\Phi(\alpha)$ cumulative distribution function, and

$$\delta(\alpha) = \lambda(\alpha)[\lambda(\alpha) - \alpha].$$

Theorem 1.3.1 along with the relationship between the expected value, $E(X)$, and the variance, $Var(X)$,

$$\begin{aligned} Var(X) &= E(X^2) - E^2(X), \\ \implies E(X^2) &= Var(X) + E^2(X), \end{aligned}$$

gives the following

$$\begin{aligned} E[(y_d^i)^2 | y_d^i \leq L] &= Var[y_d^i | y_d^i \leq L] + E^2[y_d^i | y_d^i \leq L], \\ &= \sigma^2(1 - \Lambda(\xi^i)[\Lambda(\xi^i) - \xi^i]) + (y_i - \sigma\Lambda(\xi^i))^2, \\ &= \sigma^2 - \sigma^2\Lambda(\xi^i)^2 + \sigma^2\Lambda(\xi^i)\xi^i + y_i^2 - 2\sigma\Lambda(\xi^i)y_i + \sigma^2\Lambda(\xi^i)^2, \\ &= y_i^2 - 2\sigma\Lambda(\xi^i)y_i - \sigma^2\xi^i\Lambda(\xi^i) + \sigma^2. \end{aligned}$$

The censored data is updated with

$$\begin{aligned} \bar{y}^i &= X^i y_d^i + (1 - X^i)E[y_d^i | y_d^i \leq L], \\ &= X^i y_d^i + (1 - X^i)[y_i - \sigma\Lambda(\xi^i)]. \end{aligned}$$

The squared residuals are updated through

$$\begin{aligned} \bar{r}^i &= X^i E[(y_d^i - y_i)^2] + (1 - X^i)E[(y_d^i - y_i)^2 | y_d^i \leq L]. \\ &= X^i(y_d^i - y_i)^2 + (1 - X^i)[E[(y_d^i)^2 | y_d^i \leq L] - 2y_i E[y_d^i | y_d^i \leq L] + y_i^2], \\ &= X^i(y_d^i - y_i)^2 + (1 - X^i)\sigma^2[1 - \xi^i\Lambda(\xi^i)]. \end{aligned}$$

The EM Algorithm is presented in algorithm (1.6). A stopping criterion for this algorithm is the relative change between $\hat{q}, \hat{\sigma}$. EM results in estimates of the expected value and variance for the censored data.

1.3.4 Confidence and Prediction Intervals

An error-free model with noiseless data is an unreasonable expectation. In practice, one should expect noise in the data and model due to human or technical imperfection. Thus, there will always be some degree of uncertainty in the model that needs to be taken into consideration. Confidence and prediction intervals are used to understand the extent of uncertainty involved in estimating our

Algorithm 1.6

1. Estimate $\hat{q}^0$ using y_d and ordinary least squares where the censored data is adjusted by half to $\frac{L}{2}$. Set $k=0$ and compute an initial estimate for σ^2 from

$$(\hat{\sigma}^{(0)})^2 = \frac{1}{N}\sum_{i=1}^{N} |\bar{y}^i - y(t_i;\hat{q}^{(0)})|^2.$$

2. Let $\hat{y}_i^{(k)} = y(t_i,\hat{q}^{(k)})$ and $\hat{\xi}^{i(k)} = \frac{L-y_i^{(k)}}{\hat{\sigma}^{(k)}}$ and update the data and residuals by

$$\bar{y}^{i(k)} = X^i y_d^i + (1-X^i)[\hat{y}_i^{(k)} - \hat{\sigma}^{(k)}\Lambda(\hat{\xi}^{i(k)})],$$
$$\bar{r}^{i(k)} = X^i(y_d^i - \hat{y}_i^{(k)})^2 + (1-X^i)(\hat{\sigma}^{(k)})^2[1-\hat{\xi}^{i(k)}\Lambda(\hat{\xi}^{i(k)})].$$

3. Compute $\hat{q}^{(k+1)}, \hat{\sigma}^{(k+1)}$ using ordinary least squares by solving

$$\hat{q}^{(k+1)} = \arg\min \frac{1}{N}\sum_{i=1}^{N} |\bar{y}^{i(k)} - y(t_i,q)|^2,$$

and updating $\hat{\sigma}$ with

$$(\hat{\sigma}^{(k+1)})^2 = \frac{1}{N}\sum_{i=1}^{N} \bar{r}^{i(k)}.$$

parameters. In calculating these intervals, standard errors are computed from the model predictions using the parameters that have been estimated. Techniques and notation as in [5, 7, 13, 88, 94] are used.

1.3.4.1 Parameter Confidence Intervals

Consider the statistical model

$$Y_j \equiv f(t_j, \vec{q}_0) + \epsilon_j, \ j = 1, 2, ..., N, \tag{1.29}$$

for N observations where f is the model in terms of the theoretical true parameter values, $\vec{q}_0 \in \mathbb{R}^p$. The errors, ϵ_j, are assumed to be independent and identically distributed (i.i.d.) random variables with mean $E[\epsilon_j] = 0$ and variance, $Var(\epsilon_j) = \sigma_0^2$ where σ_0^2 is unknown. Thus, Y_j are i.i.d. with mean $f(t_j, \vec{q}_0)$ and variance σ_0^2. The parameters, q, are estimated using the ordinary least squares approach

$$q^* = \arg\min J(q) = \sum_{j=1}^{N} |y_j - f(t_i, \vec{q})|^2, \tag{1.30}$$

where $\{y_j\}$ is a realization of the observation process $\{Y_j\}$ and q^* is an estimator that depends on the sampling size. Since Y_j is a random variable, so is q^* with a distribution called the sampling distribution. A sampling distribution characterizes the distribution of all the values an estimator $\{q^*\}$ could have across all realizations $\{y_j\}$ with data size, N, that could be collected. Thus, the standard errors provide a measure of the extent of uncertainty involved in estimating q using the estimator q^* with sample size N. Here p-multivariate Gaussian distributions with asymptotic convergence in distribution, mean $E[q^*(\vec{Y})] \approx \vec{q}_0$, and covariance matrix $\Sigma_0 \approx \sigma_0^2 (S^T(q_0)S(q_0))^{-1}$ are used to approximate the sampling distribution. Asymptotic convergence in distribution means that the cumulative distribution functions converge as $N \to \infty$. Here $S(q_0)$ is the sensitivity matrix similar to (1.23). Consequently, the sampling distribution approximates satisfy

$$q^*(Y) \sim \mathcal{N}_p(q_0, \Sigma_0) \approx \mathcal{N}_p(q_0, \sigma_0^2 (S^T(q_0)S(q_0))^{-1}), \tag{1.31}$$

for large N. Note that σ_0^2 is approximated by

$$\sigma_0^2 \approx \hat{\sigma}^2 = \frac{1}{N-p} \sum_{j=1}^{N} (y_k - f(t_k, q^*))^2. \tag{1.32}$$

The standard errors that will be used in the half-widths of the confidence intervals are given by

$$SE_k(q) = \sqrt{\Sigma_{kk}(q)},\ k = 1, 2, ..., p.$$

Thus, a $100(1-\alpha)\%$ confidence interval for parameter q_k is

$$\hat{q}_k \pm \tau^{N-p}_{1-\frac{\alpha}{2}} \hat{\sigma}_k SE_k(q)$$

where $\tau^{N-p}_{1-\frac{\alpha}{2}}$ is the $1-\frac{\alpha}{2}$ quantile of a student's t-distribution with $N-p$ degrees of freedom. $\alpha = .05$ since 95% confidence intervals are used. [49, 88] Given the parameter estimates, the next step is quantifying the accuracy in the model predictions.

1.3.4.2 Predictive Confidence Intervals

An understanding of the uncertainty in the model predictions is important for making conclusions. This is determined by calculating predictive confidence intervals. Consider an estimation of the true mean response, $\bar{y}_j$, of the output

$$\hat{y}_j = f(t_j, \hat{q}),$$

where $\hat{q}$ is an estimate of the solution to (1.30). Note that $\hat{q}$ is close to the true value, q_0, for large N so

$$\nabla f(t_j, q_0) \approx \nabla f(t_j, \hat{q}).$$

This implies that

$$E[\hat{y}_j] = \bar{y}_j,$$

and by using Taylor series, the following is observed

$$\begin{aligned}
Var[\hat{y}_j] &= Var[f(t_j, \hat{q})], \\
&\approx Var[f(t_j, q_0) + \nabla f(t_j, q_0)(\hat{q} - q_0)], \\
&= Var[f(t_j, q_0)] + Var[\nabla f(t_j, q_0)(\hat{q} - q_0)], \\
&= \nabla f(t_j, q_0)^T Var[(\hat{q} - q_0)] \nabla f(t_j, q_0), \\
&= \sigma_0^2 \nabla f(t_j, q_0)^T (S(q_0)^T S(q_0))^{-1} \nabla f(t_j, q_0).
\end{aligned}$$

Thus,

$$\frac{\hat{y}_j - \bar{y}_j}{\hat{\sigma}\sqrt{v_0}} \sim \tau^{N-p}_{1-\frac{\alpha}{2}},$$

such that

$$v_0 = \nabla f(t_j, q_0)^T (S^T S)^{-1} \nabla f(t_j, q_0). \tag{1.33}$$

Therefore, the confidence interval for $\hat{y}_j$ is given by

$$\hat{y}_j \pm \tau_{1-\frac{\alpha}{2}}^{N-p} \hat{\sigma} \sqrt{v_0}.$$

The variance estimator $\hat{\sigma}^2$ specified in (1.32) is employed here. A followup question to consider is given an output, where will the next prediction be given the uncertainty in the parameter and prediction values. The answer to the aforementioned question lies in prediction intervals. [88]

1.3.4.3 Prediction Intervals

Prediction intervals provide a range of values for a future prediction. Consider (1.29) and recall that

$$\epsilon_j \sim \mathcal{N}(0, \sigma^2).$$

If $\hat{q}$ is the estimated solution to (1.30), a reasonable estimate for Y_j at time t_j is

$$\hat{y}_j = f(t_j, \hat{q}).$$

Taylor series expansion is used to obtain

$$\begin{aligned} f(t_j, \hat{q}) &\approx f(t_j, q_0) + \nabla f(t_j, q_0)^T (\hat{q} - q_0), \\ \implies Y_j - \hat{y}_j &\approx Y_j - f(t_j, q_0) - \nabla f(t_j, q_0)^T (\hat{q} - q_0), \\ &= \epsilon_j - \nabla f(t_j, q_0)^T (\hat{q} - q_0). \end{aligned}$$

This implies that

$$\begin{aligned} E[Y_j - \hat{y}_j] &\approx E[\epsilon_j] - \nabla f(t_j, q_0)^T E[\hat{q} - q_0], \\ &\approx 0, \end{aligned}$$

and

$$\begin{aligned} Var[Y_j - \hat{y}_j] &\approx Var[\epsilon_j] + Var[\nabla f(t_j, q_0)^T (\hat{q} - q_0)], \\ &\approx \sigma_0^2 + \sigma_0^2 \nabla f(t_j, q_0)^T (S(q_0)^T S(q_0))^{-1} \nabla f(t_j, q_0), \\ &= \sigma_0^2 (1 + v_0). \end{aligned}$$

Here v_0 is the expression in 1.33. This means that $Y_j - \hat{y}_j$ is asymptotically $\mathcal{N}(0, \sigma^2(1+v_0))$. Since $\hat{\sigma}^2$ is independent of Y_j and asymptotically independent of $\hat{q}$ because we noted earlier that $\hat{q} \approx q_0$ for

large N, then $\hat{\sigma}^2$ is asymptotically independent of $Y_j - \hat{y}_j$. Thus,

$$\frac{Y_j - \hat{y}_j}{\hat{\sigma}\sqrt{1+v_0}} \sim \tau^{N-p}_{1-\frac{\alpha}{2}},$$

asymptotically. Therefore, the prediction interval is given by

$$\hat{y}_j \pm \tau^{N-p}_{1-\frac{\alpha}{2}} \hat{\sigma}\sqrt{1+v_0}.$$

Note that the difference between the predictive confidence interval and prediction interval is that there is $\sqrt{v_0}$ in the former and $\sqrt{1+v_0}$ in the latter. Thus, the prediction intervals will be wider than the prediction confidence intervals.

CHAPTER

2

OPTIMIZATION OF KLYSTRON DESIGNS USING DETERMINISTIC SAMPLING METHODS

2.1 Introduction

This chapter introduces a process for optimizing the design of klystron circuits. Developed in the 1930s, the klystron (shown in Figure 2.1) is categorized as a linear-beam tube [6, 56, 77]. There are several different kinds of klystrons including multiple-beam, multiple-cavity, reflex, and clustered-cavity klystrons. It uses an electron beam to amplify signals such as radio-frequency(RF) waves and microwaves using interactions between several cavities that line the inner walls of the klystron. The motivation for designing these devices was a desire for higher RF powers at higher RF frequencies. There are many applications for klystrons in linear colliders, communications, plasma heating, and radar systems [6, 30, 54, 56, 77, 92, 93, 101].

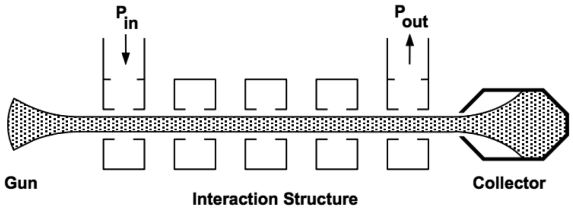

Figure 2.1 Diagram of a Klystron from [99].

Klystrons have the ability to operate under different frequency ranges or bandwidths in order to be a part of these different applications. An electron gun (*Gun* in Figure 2.1) emits an electron beam into the klystron. The beam travels through a floating drift tube to a collector and interacts with the cavities that line the tube. There are four types of cavities that can be found in a klystron. The first cavity is called the input cavity (first cavity in Figure 2.1 with P_{in}). This is where RF input power is inserted via an external circuit. The last cavity is called the output cavity where RF output power is extracted (fifth cavity in Figure 2.1 with P_{out}). The cavities in between are called idler cavities (middle three cavities in Figure 2.1) with the cavity next to the output cavity (fourth cavity in Figure 2.1) called the penultimate cavity. These cavities serve to further enhance the velocity modulation of the electrons causing electron bunching since some of the electrons are decelerated, accelerated or left alone from the resonant frequencies of the cavities. The cavity frequencies depend on the cavity shape. The penultimate cavity's resonant frequency is tuned higher than the other cavities to maximize efficiency by sharpening the bunches. The interaction velocity between the cavities and beam that modulates the velocity of the electrons with a modulating electric field is given by

$$u = u_0(1 + \frac{\alpha M}{2} \sin \omega t),$$

where u_0 is the initial velocity, α is the depth of modulation, M is the modulating coefficient and ω is the electron plasma frequency. The electron beam carries RF current with it as it moves through the tube given by

$$i(z) = \frac{i_1}{|1 - X \cos \omega t|},$$

where z is the position in the klystron, i_1 is approximately the DC current for small α, and

$$X = \frac{\omega z}{u_0} \frac{\alpha M}{2},$$

is the bunching parameter. As the beam traverses the gaps of the idler cavities, current is induced in the cavities at an instantaneous rate of

$$i = E_v q v.$$

This results in an electric field across the gap that further modulates the velocities of the electrons and results in more bunching. Finally, after reaching the output cavity, RF output power is extracted that is substantially larger than the input power via an external circuit [6, 56]. The positions and frequencies of the cavities play a vital role in RF power amplification.

A klystron is typically used as part of another device. A satellite, for instance, uses a klystron to amplify the RF signal to reach long distances. Specific values for the gain, efficiency, and output power are usually the most important desires for klystron users. The gain gives the magnitude in which the klystron is amplifying and is given by

$$\text{gain} = 10 \log_{10} \frac{P_{out}}{P_{in}}.$$

The efficiency of a klystron describes how much power is being lost in the tube and is given by

$$\eta = \frac{P_{out}}{V_b \cdot I},$$

where P_{out} is the RF output power, V_b is the beam voltage, and I is the beam current [6]. Efficiencies exceeding 80% have been demonstrated, with gains over 43 dB [105]. However, high efficiency klystrons have a number of cavities, each of which has several parameters that must be adjusted to produce high efficiency and gain. The parameters are interdependent, so the design process can be an extended, iterative exercise. Even with a skilled klystron engineer, this can take considerable time.

Recently in [67], the authors introduced an automatic optimization technique for the design of a klystron interaction structure using evolutionary algorithms. The methodology was applied to optimally design the interaction structure of the B-factory klystron [46], where all cavity frequencies, drift lengths from previous cavities, and input coupling were considered as design or free parameters. As noted by the authors, the evolutionary algorithm can not be applied to the optimal design of the klystron interaction structure in its generic form but several modifications need to be considered for it to converge to a useful set of klystron parameters.

This chapter describes how to automate the klystron design process using a variety of deterministic sampling methods conveniently bundled in MATLAB. Since the optimization methods are considered as black boxes, they are good candidates for the automated design of a klystron. In Section 2, we introduce the design software to simulate klystron output given a set of input parameters. We also describe in detail the construction of the automated process used for klystron design. Section 3 gives the results from klystrons in which we implement our numerical scheme and shows that we achieve significant improvements even with a design that was obtained using conventional methods by a skilled engineer. We also show that a very rough design can be improved dramatically. Section 4 discusses the advantages and disadvantages of our optimizer and why our scheme works well. Lastly, Section 5 gives concluding remarks concerning our optimizer and the automation of the klystron design process.

2.2 Methodology

As discussed before, a klystron designer will take specifications and attempt to find optimal parameters to achieve a desired output. For this optimal design problem, we formulate an objective function that will penalize the method the farther it is from the goal. Let $J(p)$ be the objective function to be minimized and $p = [p_1 \cdots p_n]$ be the set of inputs or free parameters. First, the deterministic sampling methods that are used to strategically find the p^* that minimizes J are given. Then we will give a brief description of the klystron simulation software. Finally, we will give the algorithm for the optimization scheme.

2.2.1 Deterministic Sampling Methods

The packages in MATLAB pertinent to our study include the deterministic sampling methods described earlier in Chapter 1 in Section 1.2.3.1.1 and Section 1.2.3.1.2. That is the Nelder-Mead algorithm (1.3) and implicit filtering algorithm (1.4). We do not use gradient based methods, since there is no analytic gradient available for the klystron design problem.

2.2.2 Simulation Programs

For numerical proof of concept, we use the two, large signal, klystron simulation codes AJDISK and TESLA. This provides a good demonstration of our approach because the analysis and design are independent of the code used.

2.2.2.1 AJDISK

AJDISK is a 1-D(one dimensional), large signal, klystron simulator. Given a set of parameters, such as cavity frequencies and positions, AJDISK will predict klystron performance characteristics, such as output power and gain. The algorithm, as described in [59], breaks the electron beam into thin rectangular plates, then calculates the electric field in each cavity using the space charge field from the other plates to determine the force on each plate. The dynamics of the charges (plates) are determined by the Lorentz force (conservation of momentum) equations given by

$$\frac{d(\gamma m \bar{v})}{dt} = q(\bar{E} + \bar{v} \times \bar{B}),$$

where γ is the relativistic correction factor. If it is assumed that the beam is confined by the infinite magnetic field and the direction of propagation of the charge is in the z direction, then the Lorentz force equation is simplified to

$$\begin{aligned} \gamma^3 m \frac{d^2 z}{dt^2} &= q E_z, \\ &= q(E_{cav} + E_{spch}), \end{aligned}$$

where the electric field is split between the electric field from the klystron cavity E_{cav} and the space charge field E_{spch} from the other plates. The above equation is integrated to find the charge position as a function of the fields E_{cav} and E_{spch}. For the calculations of E_{cav} and E_{spch}, we refer the reader to [59].

2.2.2.2 TESLA

TESLA(Telagraphist's Equations Solution for Linear beam Amplifiers) is a 2.5-D, klystron simulator [87] that evolved from the gyroton code MAGY [19] and was developed [104] at the University of Maryland and Naval Research Laboratory. The capabilities include the ability to simulate linear beam, vacuum electronic devices with cavities, such as klystrons, extended interaction klystrons, twystrons, and coupled cavity amplifiers. The model includes a self-consistent, nonlinear solution of the three-dimensional, relativistic, electron equations of motion and the solution of time-dependent electromagnetic field equations [29, 87]. For more information about TESLA, we refer the reader to [19, 29, 87, 104] and references therein. Similar to AJDISK, TESLA uses input parameters and calculates klystron performance.

2.2.3 Optimizer Scheme

The implemented optimizer scheme unifies the simulation software for a klystron and optimization algorithm in MATLAB. Both klystron simulators read the input parameters from a formatted file, and the flowchart in Figure 2.2 shows the combined capabilities of the simulators. Deterministic sampling methods in MATLAB automate the klystron design process. Results in the next section demonstrate the feasibility of this scheme for the automation of klystron design.

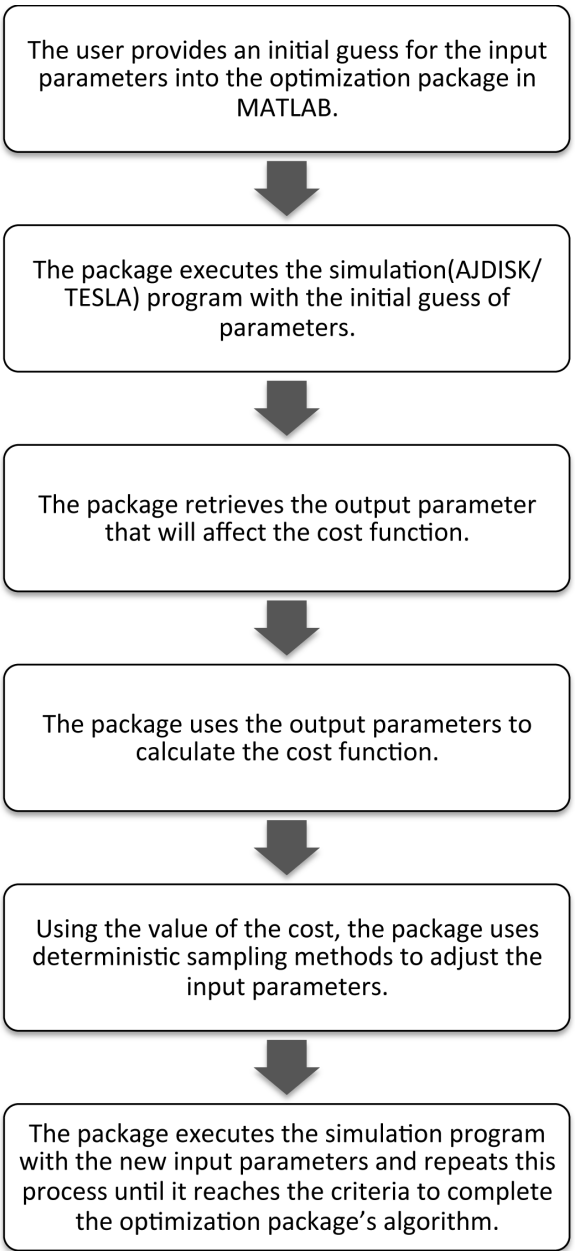

Figure 2.2 Schematic diagram for the local optimizer routine.

2.3 Examples

In this section, we use the optimization algorithm in Figure 2.2 to design three example klystrons. The examples include both narrow and wide band klystrons. The results will illustrate how our optimization algorithms take user provided initial data and not only makes the design process more efficient, but also significantly improves the klystron design.

2.3.1 CBAND

CBAND is a nine cavity, wideband klystron. This is a multi-beam device with a voltage of 25 kV and a current per beamlet of 1.2 A. While there were 18 beams, all of the analysis here is for a single beam device with R/Q equal to 1/18 times the actual value. Our goal was to achieve 40db gain and 6940 watts output power over a bandwidth of 5400-5770 GHz. This is a 6.6% bandwidth. As shown in Figure 2.3 (dashed curves), we were unable to achieve this bandwidth with a conventional design process.

The parameters that we chose to optimize and their initial values are given in Table 2.1. We acquired this initial data from engineers with extensive experience in designing this type of klystron. For more information on these parameters, we refer the reader to [6].

Achieving at least 40db gain across the required band was critical; however, it was also desirable to have the gain curve as flat as possible. Achieving 40db gain across the band simultaneously achieved at least 6940 W power out. Thus, we can choose our objective functional without incorporating power out and focusing solely on achieving our goal gain. Consequently, we chose our objective functional as

$$c(x) = \begin{cases} 1000(40-x) & \text{if } x < 40 \\ 0 & \text{if } x \geq 40, \end{cases}$$

with the weight(1000) to sufficiently penalize the optimizer for not achieving the goal. We sampled across the bandwidth by uniformly picking 20 frequencies between 5400-5770 MHz. We then picked the smallest value of those evaluations and used that in the objective functional. Now that we have our initial data and objective functional, we can apply the steps described in Figure 2.2 to obtain the results in Figure 2.3. We note that AJDISK is the simulator used along with the Nelder-Mead Algorithm described before as the deterministic sampling method.

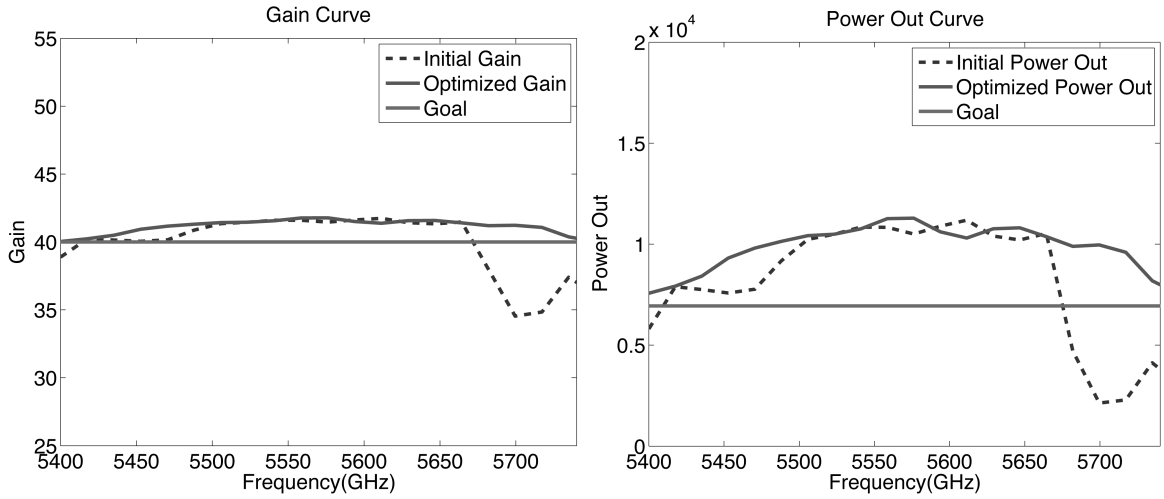

Figure 2.3 The Gain (left) and The Power Out (right) for CBAND with a run time of approximately 38 hours.

We achieved the desired specifications, in both gain and power out, across the bandwidth. The optimized parameters, shown in Table 2.1, were obtained on a laptop with a 2.70GHz Intel(R) Core(TM) i7-2620M CPU processor in less than two days of computation.

Table 2.1 Initial and optimized parameter values for the CBAND klystron.

Initial and Optimized parameters for CBAND

Parameters	Initial Data	Optimized Values
Cavity Position of Idler Cavity 1-3	.07112 m	.07112 m
	.12954 m	.13344 m
	.18415 m	.1853 m
Cavity Position of Idler Cavity 4-6	.22860 m	.23053 m
	.26543 m	.26626 m
	.29972 m	.30041 m
Cavity Position of Penultimate Cavity	.32893 m	.32958 m
Cavity Position of Output Cavity	.35535 m	.35024 m
Cavity Frequency of Input Cavity	5399 MHz	5393.924 MHz
Cavity Frequency of Idler Cavity 1-3	5473 MHz	5481.732 MHz
	5595 MHz	5597.977 MHz
	5692 MHz	5705.562 MHz
Cavity Frequency of Idler Cavity 4-6	5718 MHz	5713.674 MHz
	5749 MHz	5765.532 MHz
	5773 MHz	5784.384 MHz
Cavity Frequency of Penultimate Cavity	5855 MHz	5870.885 MHz
Cavity Frequency of Output Cavity	5602 MHz	5583.489 MHz
Power In	.75 W	.75 W
Q_e of Output Cavity	8	8.038

2.3.2 KSB

KSB is a five cavity, PPM (Periodic Permanent Magnet) focused narrowband klystron with operating frequency 2.856 GHz. TESLA is the simulator used with a combination of implicit filtering and Nelder-Mead algorithms as the deterministic sampling methods. We chose TESLA because AJDISK is unable to handle PPM focused klystrons. We first use the Nelder-Mead algorithm to find a minimum of the cost. Then, since implicit filtering can leap over local minima, we use it after the Nelder-Mead algorithm to find a better local minimum. Once implicit filtering is unable to locate a better local minimum, we then implement the Nelder-Mead algorithm with the improved input parameter values to determine if we can improve our results. If the previous step is successful, we reimplement the implicit filtering algorithm. We continue this process until the local minimum does not improve for either method. We optimized for two different designs of this klystron on a laptop equipped with a 1.80 GHz Intel(R) Core(TM) i7-4500U CPU processor.

2.3.2.1 Design One

Our goal is to achieve 5.5 MW power out with a voltage of 127 kV and 90.5 A current. The optimized parameters and their initial values are given in Table 2.2. Because the cavities were inserted between the pole pieces of a PPM focusing structure, the positions of the cavities could only be varied by increments of $\frac{1}{2}$ of the PPM period. Thus, for the purposes of this optimization, the cavity positions were fixed. Since this was a narrow band klystron, the Q's of the idler cavities were unimportant if they are above some minimum values. Thus, only the resonant frequencies of all cavities and the Q of the output cavity were varied in the optimization. The initial data in Table 2.2 were generated by trial and error to obtain a klystron with 4 MW power out to start.

Our objective functional is given by

$$c(x)=\begin{cases} 5,500,000-x & \text{if } x<5,500,000 \\ 0 & \text{if } x\geq 5,500,000. \end{cases}$$

We used this objective functional because the difference between the target and result was large enough to penalize the optimizer. We achieved 5.501 MW power out with the optimized parameters given in Table 2.2 in about four and a half days of computer time.

Table 2.2 Initial and optimized parameter values for design one of the KSB klystron.

Initial and Optimized parameters for Design One of KSB

Parameters	Initial Data	Optimized Values
Cavity Frequency of Idler Cavity 1-2	2864 MHz	2856 MHz
	2897 MHz	2908 MHz
Cavity Frequency of Penultimate Cavity	2934 MHz	2931 MHz
Power In	104.7 W	175.1 W
Q_e of Output Cavity	21.4	35.3

2.3.2.2 Design Two

This illustrates how the optimizer could improve a design which is far from optimal. It was motivated by the desire to run the klystron described in Section 2.3.2.1 at lower voltage. At the lower voltage 97.6 kV, the efficiency was reduced significantly, resulting in an output power of 1.283 MW. The optimizer was employed to increase this power to the desired level of 3 MW. The objective functional is given by

$$c(x) = \begin{cases} 3,000,000 - x & \text{if } x < 3,000,000 \\ 0 & \text{if } x \geq 3,000,000. \end{cases}$$

A weighting coefficient in the objective functional was not required because the difference between the target and result was large enough to penalize the optimizer. We achieved 2.732 MW of power out, a 48% increase over the original output power, with optimized parameters given in Table 2.3 in approximately four days of computing time.

Table 2.3 Initial and optimized parameter values for design two of the KSB klystron.

Initial and Optimized parameters for Design Two of KSB

Parameters	Initial Data	Optimized Values
Cavity Frequency of Idler Cavity 1-2	2858 MHz	2859 MHz
	2901 MHz	2898 MHz
Cavity Frequency of Penultimate Cavity	2936 MHz	2884 MHz
Power In	126.4 W	138.2 W
Q_e of Output Cavity	21.4	46.1

2.4 Discussion

There are many parameters that we could have optimized, but we chose cavity frequency and cavity position because other parameters did not have as much of an impact on the output we desired. Because the deterministic sampling algorithms we used find a local minimum for the cost function, the initial guess for the parameters played a crucial role in achieving the output specifications. The number of parameters involved in klystron simulations can be very high so local minimum techniques are usually more desirable than global techniques. Even though they may not yield the best parameters, Nelder-Mead and Implicit Filtering efficiently achieve the desired output. Our new approach provides a highly efficient, automated technique to design klystrons that meet customer specifications.

2.5 Conclusion

This chapter describes how we used deterministic sampling methods to automate the design process of klystrons. This was motivated by the copious amount of time it takes to design klystrons by trial and error. It is shown that optimization can reduce the duration of the design process from weeks to days. Not only was design time reduced, but the performance achieved was better than previously achieved using manual, trial and error methods.

CHAPTER

3

MATHEMATICAL MODEL OF HEPATITIS C VIRUS

3.1 Introduction

Over 200-300 million people worldwide are infected with a virus that affects the liver called Hepatitis C (HCV) that was discovered in 1989 [89, 97]. It is usually spread by blood-to-blood contact via intravenous drug use, poorly sterilized medical equipment, and transfusions. Scarring of the liver and ultimately cirrhosis are just a few of the more severe complications associated with HCV [15, 84].

Six different genotypes of HCV exist due to the highly error prone RNA polymerase with the most common being genotype 1 that has the lowest levels of response to standard treatment [14, 22, 28, 75]. Genotype 1 patients have about a 50% chance for sustained viral response (SVR) while non-genotype 1 patients have about an 80% chance for SVR [53, 95]. The data available for the research was provided by the University of Sao Paulo, School of Medicine in Sao Paulo, Brazil and consist of genotype 1 patients.

One of the first treatments for HCV was 6-12 months monotherapy with interferon glycoproteins as

the only medication used. Interferon is naturally secreted from our bodies to fight off infection and monotherapy treatment with them is associated with around 10% SVR [62]. The addition of ribavirin (RBV), a drug believed to render some of the virus non-infectious, increased SVR to around 30% [62]. RBV monotherapy is not recommended because it does not give a significant benefit to SVR [20]. Until recently, the most common therapy was a combination of pegylated Interferon (IFN) and RBV for 24-48 weeks which yielded about a 45% SVR [41, 42, 44, 62, 79, 95]. One of the major differences between IFN and standard inteferon glycoproteins is that the pegylation allows the drugs to stay in the body longer [102]. There has also been clinical trials with RBV monotherapy before and after IFN+RBV therapy described in [82, 91]. Recently, new drugs called direct-acting antiviral agents (DAAs) have raised the chance for SVR for HCV patients.

DAAs give an increase to about an 80% chance for SVR for genotype 1 [98]. According to the FDA, DAAs are drugs that interfere with specific steps in the HCV replication cycle by taking advantage of the biological makeup of HCV [45]. HCV is a single stranded RNA molecule that is several nucleotides in length. During HCV's life cycle it is translated into a polyprotein that is composed into structural and nonstructural proteins that aid in replication. During post-translational processing, DAAs called protease inhibitors block a key protease from the replication process and hinders further infection [58, 63, 98]. Among the protease inhibitors available are boceprevir, telaprevir and simeprevir. Simeprevir is recommended over telaprevir and boceprevir because of both improved efficacy and less side effects, but telaprevir continues to be used because of its cost efficiency in other countries such as where we received our data [17, 55, 83, 109]. Telaprevir, a substrate of the efflux transporter p-glycoprotein, is an inhibitor of the CYP3A4 enzyme [107].

Mathematically modeling viral dynamics with data has led to further understanding of how treatment strategies dictate viral load patterns and how they compare to each other. Many nonlinear ODE models have been proposed to model different treatment strategies for HCV. One of the first models was given by Neumann et. al which attempted to describe HCV dynamics with interferon monotherapy [75]. Improvements were made to Neumann's model to better describe different mechanisms in the liver during treatment like the regeneration of liver cells. Adjustments were also made to include the standard of care, IFN and RBV. Some of these modifications can be found in [8, 35, 36, 95]. [95] had data after the end of the treatment phase so that the model can give a more accurate representation of it's prediction of SVR. The introduction of DAAs has ushered in more mathematical models that include this type of therapy [27]. For example, mathematical models have been proposed using telaprevir monotherapy [3, 4, 52, 85, 86] and in combination with IFN and RBV [4] that uses Bayesian Feedback to estimate the parameters in the model. The challenges

that come with models with DAAs is that since they are relatively new, there is not as much data available [3]. It can be difficult to predict SVR because of lack of data after the treatment phase ends due to how recent the drugs have been approved.

This chapter introduces a novel approach for the development of a mathematical model that describes how HCV propagates through the liver given the triple-drug combination treatment of IFN, RBV and the DAA telaprevir. We use a sensitivity and identifiability analysis to determine which parameters can be best estimated from the data. In Section 2, we describe how we adapted a previously known HCV model to include telaprevir. Section 3 discusses the analysis and results used for model validation. Section 4 gives results from parameter estimation. Lastly, Section 5 provides concluding remarks.

3.2 Model

3.2.1 Motivation

The original model for HCV dynamics in Neumann et. al. [75] was frequently used to assess viral-load profiles after short-term treatment and is given as

$$\begin{aligned} \frac{dT}{dt} &= s - dT - (1-\eta)\beta VT, \\ \frac{dI}{dt} &= (1-\eta)\beta VT - \delta I, \\ \frac{dV}{dt} &= (1-\epsilon)pI - cV. \end{aligned} \tag{3.1}$$

One of the key contributions of the model was the understanding of the mechanism of IFN. It was unknown whether it acted through $\eta > 0$ or $\epsilon > 0$. In [75], it is determined that it is through ϵ which inhibits production of the virus. The drawback to (3.1) is that it cannot describe patients exhibiting Breakthrough, relapse, and most importantly SVR. These responses are reasons that early viral response does not uniformly predict responses in the long term. Another important aspect is the handling of viral load measurements below the LLOQ. Previous analysis omitted the data below LLOQ, but it can contain critical information regarding long-term treatment outcome. Snoeck, et al., [95], presents a mathematical model used for HCV with the drug treatment combination of IFN and RBV that attempts to address both the long term responses and the use of the LLOQ.

3.2.2 Assumptions

Snoeck, et al. uses a maximum likelihood estimation (MLE) of the parameters, using the extended stochastic expectation-maximization(SAEM) algorithm which was implemented in the MONOLIX software. However, some of the parameters are fixed to values that make sense biologically. The fixed values come from previous literature that provides discussions about the parameters in question. In addition, the proliferation rates are the same in both the infected and uninfected hepatocytes. The parameters for the maximum number of hepatocytes, death rate of uninfected hepatocytes, and the production of new hepatocytes are all fixed to values that can be found in [35, 68, 90]. [95] also accounts for interindividual variability between patients. The estimated parameters that account for interindividual variability are the basic reproductive number, the natural clearance of the virus, the death rate of the infected hepatocytes and the estimated weekly dose of IFN that results in a 50% inhibition of the virion production using a nonlinear mixed effects model. If at any point there is less than one infected hepatocyte, the virion production rate is set to zero. This model does not take into account mutations due to drug resistant strands of the virus.

3.2.3 Model from Snoeck et. al.

The mathematical model given in [95] is described by the system of nonlinear ODEs

$$\begin{aligned}
\frac{dT}{dt} &= s + rT(1-\frac{T+I}{T_{max}}) - dT - \beta V_I T, \\
\frac{dI}{dt} &= \beta V_I T + rI(1-\frac{T+I}{T_{max}}) - \delta I, \\
\frac{dV_I}{dt} &= (1-\bar{\rho})(1-\bar{\epsilon})pI - cV_I, \\
\frac{dV_{NI}}{dt} &= \bar{\rho}(1-\bar{\epsilon})pI - cV_{NI},
\end{aligned} \tag{3.2}$$

where T (uninfected hepatocytes), I (infected hepatocytes), V_I (infectious virions) and V_{NI} (non-infectious virions) are natural states (international units IU/mL) from a standard model of viral infection [75]. The number of uninfected hepatocytes increases each day with reproduction rate s and regeneration rate r. That number decreases each day as those hepatocytes die naturally at a rate d or infected at a rate β. The maximum number of hepatocytes per mL is T_{max}. The number of infected hepatocytes increases when the healthy liver cells are infected and when the infected cells regenerate themselves. That number decreases when they die off naturally at a rate δ. Infected hepatocytes produce both infectious and noninfectious virions at a rate p. Virions are naturally cleared at a rate c. IFN inhibits virion production while RBV renders some of the virus noninfectious.

The drug efficacies of IFN and RBV are represented by ϵ and ρ, respectively. The bounds for IFN and RBV are $0 < \epsilon \leq 1$ and $0 < \rho \leq 1$ where the more effective the drug is, the closer the efficacy of the drug will be to 1. Snoeck uses data that extends beyond treatment for patients so $\bar{\epsilon}$ and $\bar{\rho}$ account for the exponential decays of the efficacies of the drugs after treatment has ceased. The exponential decay of the drug efficacies are given by

$$\bar{\epsilon} = \epsilon e^{-k(t-t_{end})_+},$$

and

$$\bar{\rho} = \rho e^{-k(t-t_{end})_+},$$

where k is the efficacy decay rate, t_{end} marks the end of treatment, and

$$(a)_+ = \begin{cases} a & \text{if } a \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

The parameters ϵ and ρ are given by the following expressions

$$\epsilon = \frac{\text{Dose}_{\text{PEG}}}{\text{ED}_{50_{\text{PEG}}} + \text{Dose}_{\text{PEG}}}, \tag{3.3}$$

and

$$\rho = \frac{\text{Dose}_{\text{RBV}}}{\text{ED}_{50_{\text{RBV}}} + \text{Dose}_{\text{RBV}}}, \tag{3.4}$$

where Dose_{PEG} is the weekly subcutaneous dose of IFN and $\text{ED}_{50_{\text{PEG}}}$ is the estimated weekly dose that causes 50% inhibition of virion production. Dose_{RBV} represents the daily dose of RBV/kg body weight, and $\text{ED}_{50_{\text{RBV}}}$ represents the estimated daily dose in mg/kg that makes 50% of the virions noninfectious. Biologically, all state variables and parameters are non-negative.

Parameter	Value
s	$6.17\times10^{4}\ \frac{\text{hepatocyte}}{\text{mL}\cdot\text{day}}$
r	$.00562\ \text{day}^{-1}$
β	$8.7\times10^{-9}\ \frac{\text{mL}}{\text{virion}\cdot\text{day}}$
δ	$.139\ \text{day}^{-1}$
c	$4.53\ \text{day}^{-1}$
$T_{\max}$	$1.85\times10^{7}\ \frac{\text{hepatocytes}}{\text{mL}}$
d	$.003\ \text{day}^{-1}$
p	$25.1\ \frac{\text{virions}}{\text{hepatocyte}\cdot\text{day}}$
ϵ	.896
ρ	.4-.6
k	$.0238\ \text{day}^{-1}$

Table 3.1 Typical values from [95].

3.2.4 Model with DAA

Snoeck's model is adapted to incorporate the DAA, telaprevir. Recall that a DAA targets specific parts of the genome of the virus to inhibit both replication and infection. The hindrance of replication of the virus in the infected hepatocytes results in the virus not being produced by those cells. This means that the DAA should be implemented as part of the infection term, $\beta T V_I$, for inhibiting infection and viral production terms, pV_I and pV_{NI}, for inhibiting replication of the virus in (3.2). However, after simulations and analysis, it is concluded in this study that the obstruction of the infection and replication of the virus by telaprevir can be described solely as an amplifier for mitigating the production of virions alongside IFN. With this assumption, the model in [95] is modified to include the triple drug combination of IFN, RBV and teleprevir as follows:

$$\begin{aligned}
\dot{T} &= s + rT\left(1-\frac{T+I}{T_{\max}}\right) - dT - \beta V_I T \\
\dot{I} &= \beta V_I T + rI\left(1-\frac{T+I}{T_{\max}}\right) - \delta I \\
\dot{V}_I &= (1-\bar{\rho})(1-\bar{\epsilon})(1-\bar{\gamma})pI - cV_I \\
\dot{V}_{NI} &= \bar{\rho}(1-\bar{\epsilon})(1-\bar{\gamma})pI - cV_{NI},
\end{aligned} \tag{3.5}$$

where $0 < \gamma \leq 1$ represents the efficacy of telaprevir. Existence and uniqueness of solutions are proven and a steady state stability analysis is performed to validate the model.

3.2.5 Existence and Uniqueness

Typically, when showing that a solution exists for a general differential equation given by

$$\dot{x} = \hat{f}(t, x, q), \tag{3.6}$$

it is sufficient to use the following theorem 3.2.1 (Cauchy-Lipschitz theorem) from [32],

Theorem 3.2.1 *Let $||\cdot||$ denote any norm in $\mathbb{R}^n$. Given $T > 0$ and $D = \mathbb{R}^n$, let $f \in C([0,T]\times\mathbb{R}^n;\mathbb{R}^n)$ be a mapping with the property that there exists a constant $L > 0$ that satisfies the Lipschitz condition*

$$||\hat{f}(t, x_1) - \hat{f}(t, x_2)|| \leq L||x_1 - x_2||, \ \forall\, (x_1, t),\, (x_2, t) \in D \times [0, T],$$

Then the initial value problem, or Cauchy problem,

$$\dot{x}(t) = f(t, x(t)),\ 0 \leq t \leq T, \text{ and } x(0) = x_0,$$

has a unique solution $x \in C^1([0,T];\mathbb{R}^n)$.

Note that the right hand side of (3.5) corresponding to $\hat{f}$ does not satisfy a Lipschitz condition in the states due to product nonlinearities like $\beta V_I T$ and $r T \frac{T+I}{T_{\max}}$. However, since the model represents a biological system, it is impractical to expect the states to be boundless as the population increases. Thus, the non-linear terms are replaced with terms that saturate as the cell populations grow large as in [2, 7]. This causes $\hat{f}$ to be piecewise differentiable and satisfy a global Lipschitz condition to guarantee existence and uniqueness of a solution.

In a revised saturated model where each state has a saturation limit such that $x_i \leq x_i^M$ then (3.6) is rewritten as the following

$$\dot{x}^s = \hat{f}^s(t, x^s, q), \tag{3.7}$$

where any nonlinearity of the form $a_n x_i x_j$ with constant a_n is replaced with

$$a_n(x_i)a_n(x_j), \tag{3.8}$$

where the saturation term is given by

$$a_n(x_p) = \begin{cases} 0, & x_p < 0 \\ \sqrt{a_n}\, x_p, & 0 \le x_p \le x_p^M \\ \sqrt{a_n}\, x_p^M, & x_p^M < x_p. \end{cases}$$

It is observed that (3.7) is exactly (3.5) below the saturation limit and (3.8) is globally bounded and piecewise differentiable. (3.5) is rewritten to give the following

$$\dot{x} = A + B(t)x + c(t, x, q),$$

where $A = [s, 0, 0, 0]$ contains the source terms, $B(t)x$ contains the terms linear in x like $d\,T$, and $c(t, x.q)$ contains the nonlinearities. Immediately from (3.8) it is acquired that the derivative of the saturated nonlinear term is bounded and gives the following

$$||D_x\, c(t, x^s, q)|| < \infty.$$

It can now be concluded that for

$$\dot{x}^s = \hat{f}^s(t, x^s, q) = A + B(t)x^s + c(t, x^s, q),$$

and any x_1^s and x_2^s where $\hat{f}^s(t, x_i^s, q) = \hat{f}_i^s$, the following is obtained

$$\begin{aligned} ||f_1^s - f_2^s|| &= ||B(t)(x_1^s - x_2^s) + c(t, x_1^s, q) - c(t, x_2^s, q)||, \\ &= ||B(t)(x_1^s - x_2^s) + \int_0^1 D_x\, c(t, x_2^s + \bar{s}(x_1^s - x_2^s), q)(x_1^s - x_2^s) d\,\bar{s}|| \text{ by Mean Value Theorem}, \\ &\le ||B(t)||\, ||x_1^s - x_2^s|| + ||D_x\, c(t, x_2^s + \bar{s}(x_1^s - x_2^s), q)||\, ||x_1^s - x_2^s||, \\ &\le L||x_1^s - x_2^s||. \end{aligned}$$

This provides a global Lipschitz condition for (3.7). A last remark is that even though the above analysis is with the saturated system, the original system (3.5) is used during this work. The reasoning is that during simulations, no issues regarding states growing without bound were observed. Solutions being guaranteed allows the use of this system of ODEs. An analysis of the steady states will give more information about the dynamics of these solutions.

3.2.6 Steady States and Stability

The computation of steady states give the values for which the system is in stasis. The steady states are examined without outside interference so drugs are not introduced. Thus, the drug efficacies are fixed to $\epsilon = \rho = \gamma = 0$ to simulate no treatment being administered. Steady states occur at the equilibria of an ODE which are found by solving the following system of equations

$$\begin{aligned} \dot{T} &= 0, \\ \dot{I} &= 0, \\ \dot{V}_I &= 0, \\ \dot{V}_{NI} &= 0. \end{aligned} \tag{3.9}$$

(3.9) is solved to obtain two physically possible steady states. The uninfected steady state is given by

$$\begin{aligned} T &= \frac{T_{max}}{2r}\Big(r - d + \sqrt{(r-d)^2 + \frac{4rs}{T_{max}}}\Big), \\ I &= 0, \\ V_I &= 0, \\ V_{NI} &= 0, \end{aligned} \tag{3.10}$$

and the infected steady state is given by

$$\begin{aligned} T &= \frac{1}{2}\Big(-\frac{r^2 D}{A^2} + \sqrt{(\frac{r^2 D}{A^2})^2 + \frac{4rsT_{max}}{A^2}}\Big), \\ I &= T\big(\frac{A}{r} - 1\big) + T_{max} - \frac{\delta T_{max}}{r}, \\ V_I &= \frac{p}{c} I, \\ V_{NI} &= 0, \end{aligned} \tag{3.11}$$

where

$$\begin{aligned} A &= \frac{p\beta T_{max}}{c}, \\ D &= \frac{T_{max}}{r^2}[A(r-\delta) + r(d-\delta)]. \end{aligned}$$

The nominal parameters from [95] are used to acquire the uninfected and infected steady state values in Table 3.2. Since the uninfected state is unstable, introduction of the virus will likely lead to chronic infection.

State	Uninfected Steady State	Infected Steady State
T	1.92×10^7	2.788×10^6
I	0	4.928×10^5
V_I	0	2.7307×10^6
V_{NI}	0	0

Table 3.2 Steady state values for (3.5).

The local stability of each steady state is determined by analyzing the eigenvalues of the Jacobian matrix for (3.5) given by

$$J_{NI} = \begin{bmatrix} r(1-\frac{T+I}{T_{max}}) - \frac{rT}{T_{max}} - d - \beta V_I & -\frac{rT}{T_{max}} & -\beta T & 0 \\ \beta V_I - \frac{rI}{T_{max}} & r(1-\frac{T+I}{T_{max}}) - \frac{rI}{T_{max}} - \delta & \beta T & 0 \\ 0 & p & -c & 0 \\ 0 & p & 0 & -c \end{bmatrix}. \tag{3.12}$$

The uninfected steady state gives (3.12) at least one positive eigenvalue so it is locally unstable. The infected steady state gives (3.12) all negative eigenvalues so it is locally stable. Figure 3.1 shows that with the introduction of one $\frac{\text{Virion}}{\text{mL}}$, cell concentrations will rapidly propagate from the uninfected steady state to the infected steady state. This illustrates how instrumental the drugs are in viral clearance. Without treatment, V_{NI} is decoupled from the system and remains at zero. Next, information about when treatment will be administered is presented.

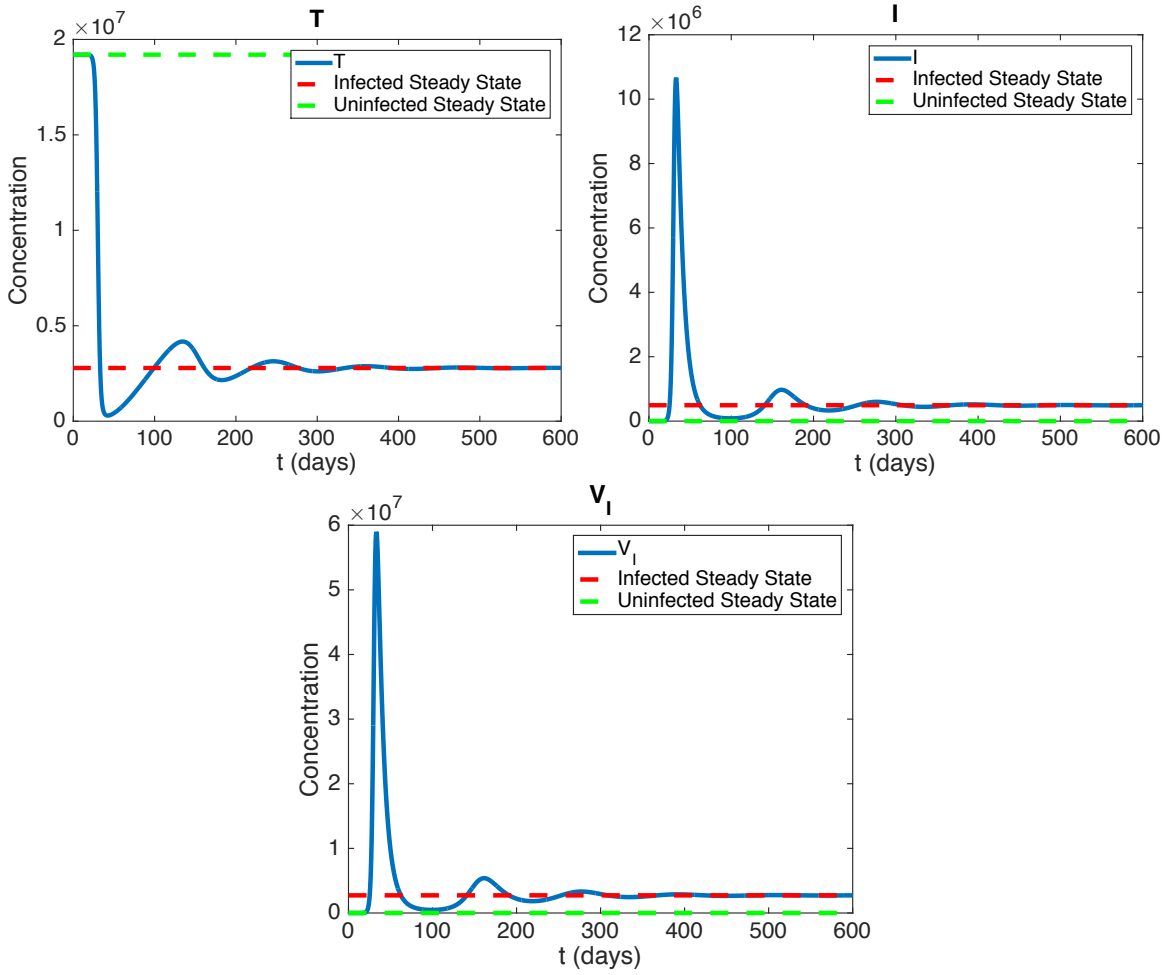

Figure 3.1 Simulation of (3.5) with initial conditions $[T_0, I_0, V_{I0}] = [1.92 \times 10^7, 0, 1]$.

3.2.7 Treatment Schedule

The data in this research uses the treatment schedule timeline as described in Figure 3.2.

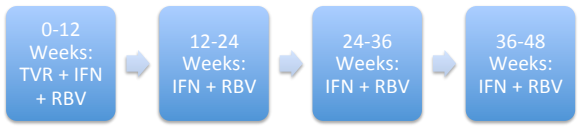

Figure 3.2 Treatment schedule for patients used for data received from patients treated at University of Sao Paulo, School of Medicine in Sao Paulo, Brazil.

(1) The patient is treated with IFN+RBV+telaprevir the first 12 weeks.

(2) If at 12 weeks, viral load > 1000ml, then discontinue treatment. Otherwise, continue 12 weeks IFN+RBV.

(3) If at 24 weeks, viral load > LLOQ(12-15ml), then discontinue treatment. Otherwise, continue 12 weeks IFN+RBV.

(4) If at 36 weeks, viral load > LLOQ, then discontinue treatment. Otherwise, continue 12 weeks IFN+RBV.

(5) End of treatment at 48 weeks.

3.3 Subset Selection

Many parameter sets in the parameter space are used to determine which parameters in (3.5) will be estimated. In particular, the subset of parameters that are locally sensitive and identifiable as described in Chapter 1 are estimated. Thus, algorithm (3.1) is implemented to perform subset selection on each set of parameters. Since these are local analyses, this procedure is repeated over a

Algorithm 3.1

1. Start with full parameter set Q.
2. Remove parameters that are not locally sensitive to attain $Q_S \subset Q$.
3. Remove parameters that are not locally identifiable from Q_S to obtain sensitive and identifiable parameter set Q_{SI}

large number of parameter sets and the parameters that appear most often in Q_{SI} are the parameters that are estimated. All other parameter values are fixed to values from literature. A biological and structural explanation for some of the fixed parameters is given in the next section.

3.3.1 Fixed Parameters

The assumptions for fixed parameters are the same as in [95]. Since the maximum number of hepatocytes in the liver is 2.50×10^{11} and HCV RNA is distributed in plasma and extracellular fluids

with a volume of $\sim 1.35\times10^4$ ml, then $T_{max}=\frac{2.50\times10^{11}}{1.35\times10^4}=1.85\times10^7$. d is obtained from hepatocyte turnover being every 300 days and $s=T_{max}\cdot d$ can be deduced in the absence of liver disease. p is always fixed because $p(1-\epsilon)$ appears in $\dot{V}$ and $\dot{V}_{NI}$ making p and ϵ, impossible to estimate uniquely. The rest of the parameters will be considered in the sensitivity analysis.

3.3.2 Sensitivity Analysis Model Considerations and Results

The procedure for the sensitivity analysis as outlined in Section 1.3.1 is implemented. The sensitivities of each parameter are ranked to obtain which parameters are most sensitive. Since there is a large range of parameter and viral load values, each parameter, q_j, is log scaled in association with the state variable, y; i.e.,

$$\frac{d\log_{10}(y)}{d\log_{10}(q_j)}=\frac{q_j}{y}\frac{dy}{dq_j}$$

is considered instead of $\frac{dy}{dq_j}$. This allows a comparison the sensitivities of each parameter using similar magnitudes. The l_2-norm is used to non-dimensionalize the sensitivities over time so the following sensitivity coefficient is considered for each parameter

$$S_{ij}=\left\|\frac{\partial y_i}{\partial q_j}\right\|_2=\left[\frac{1}{t_f-t_0}\int_{t_0}^{t_f}\left(\frac{\partial y_i}{\partial q_j}\left(\frac{q_j}{\max y_i}\right)\right)^2\partial t\right]^{\frac{1}{2}}. \tag{3.13}$$

(3.13) is defined to be the relative ranking sensitivity of each variable y_i in y with respect to each individual parameter q_j.

Since the local sensitivity analysis depends on values in q, independent sets of parameters that have a log-normal distribution are created according to the population-based model fit in Snoeck et al. That is, a sequence of independent parameter sets $\{q_k\}$ are generated from this distribution using the typical values from [95] as the mean. Initially, 5000 parameter sets are considered for this local sensitivity analysis to make it pseudo-global. It was observed that certain parameter sets in the parameter space obtained negative components in the initial conditions of (3.5). The reason for this phenomenon is that the initial conditions are parameter dependent. Also, it is desired to only examine parameter values that give measurements above the LLOQ to simulate measurements that can be observed. Therefore, only parameter sets that give all nonnegative components in the initial conditions and output above the LLOQ are analyzed. To determine pseudo-global sensitivities, a sensitivity coefficient, S_{ij}^k, is computed for each parameter in the kth parameter set. Then, if B

parameter sets are to be analyzed, then an average for all the parameter sets is computed by

$$\bar{S}_{ij} = \frac{1}{B}\sum_{k=1}^{B} S_{ij}^{k}. \tag{3.14}$$

A cutoff is determined based on the ranking of the averages attained in (3.14). Those parameters above the cutoff are further examined in the identifiability analysis. This method is a version of what is referred to as Morris Screening in [94]. Similar to the work done here, the Morris algorithm [72] averages local derivative approximations to provide more global sensitivity measures. The difference being that the variance in the parameter sets is also considered. Here that variance would be given by

$$\sigma_{ij}^{2} = \frac{1}{B-1}\sum_{k=1}^{B}(S_{ij}^{k} - \bar{S}_{ij})^{2}. \tag{3.15}$$

As explained in [94], while the mean (3.14) quantifies the individual effect of the input on the output, the variance (3.15) estimates the combined effects of the input due to nonlinearities or interactions with other inputs. The reader is referred to [72,94] and references therein for a more detailed analysis of Morris Screening. It is noted that only the marginal distributions are given in [95], so computations are ignorant of any covariances between parameters. The data that is used contains only the viral load observations. So the sensitivities of $V = V_I + V_{NI}$ are of interest. Therefore, (1.11) is considered where

$$y = [T \ I \ V_I \ V_{NI}]^T,$$

with output

$$z = V = V_I + V_{NI}.$$

Two different sets of time points are used during this analysis. The first and second set of time points come from the partial viral response(PVR) case and Breakthrough case, respectively. This will provide a better illustration of sensitivities given that treatment decays in the Breakthrough case, but doesn't in PVR. The sensitivity rankings are given in Figure 3.3 and Figure 3.4 for over 2000 and 400 parameter sets, respectively. Error bars that are 2 standard deviations from the mean are included.

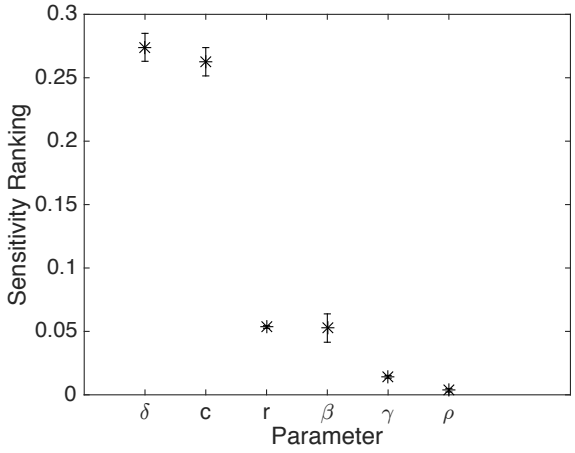

Figure 3.3 Sensitivity rankings using PVR time points.

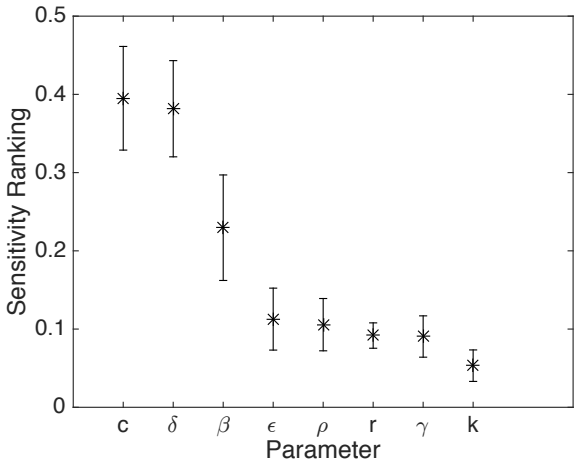

Figure 3.4 Sensitivity rankings using Breakthrough time points.

The sensitive parameters for the PVR and Breakthrough time points are $Q_{PVR} = \{\delta, c, \beta, r, \gamma\}$ and $Q_{Brk} = \{\delta, c, \beta, r, \rho, \gamma, \epsilon\}$, respectively. These parameters are considered in the identifiability analysis. Note that γ is always considered in the identifiability analysis due to there not being a value from literature to fix it to for this model. It is used to determine if it affects the identifiability of other parameters.

3.3.3 Identifiability Analysis Results

Algorithm (1.5) as described in Chapter 1 is used for the local identifiability analysis here. The analysis is applied to all of the parameter sets of sensitive parameters, Q_{PVR} and Q_{Brk}, obtained in the previous section. It is observed from Figure 3.5 that the parameters in $Q_{PVR} = \{\delta, c, \beta, \gamma\}$ are identifiable at least 50% of the time. In Figure 3.6, it is shown that the parameters in $Q_{Brk} = \{\delta, c, \beta, \gamma, \epsilon\}$ are identifiable at least 50% of the time. The parameters contained in Q_{PVR} and Q_{Brk} are estimated.

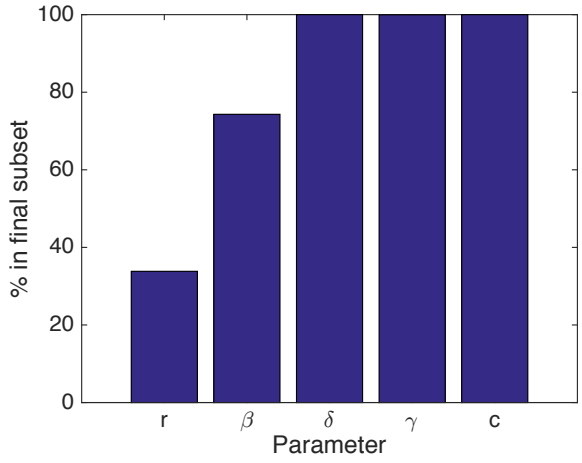

Figure 3.5 Final subset percentages using PVR time points.

3.4 Parameter Estimation

The parameters in Q_{PVR} and Q_{Brk} are estimated using the weighted sum of squares of errors (WSSE) given by

$$J(q) = w_i \sum_{i=1}^{N} [\log(V_d^i) - \log(V(t_i;q))]^2, \tag{3.16}$$

where w_i is the weight associated with the data point V_d^i at time t_i and $V(t_i;q)$ is the model output with parameters q. We used both sampling and gradient based methods to minimize this function implemented in MATLAB. The three different responses during treatment presented in Figure 3.7 are PVR, ETR (end-of-treatment response) and Breakthrough. PVR represents when the patient has an initial positive reaction to the therapy, but then the viral load rebounds during treatment

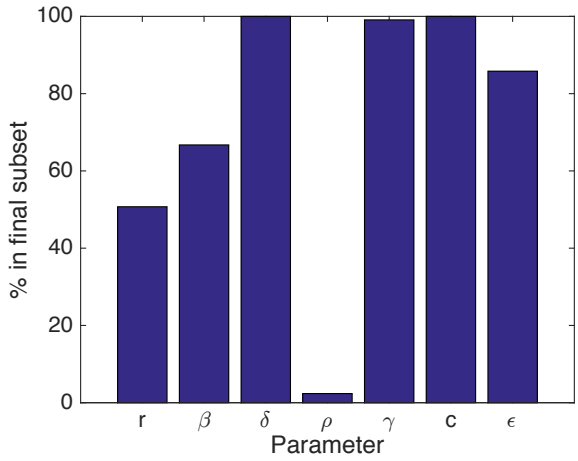

Figure 3.6 Final subset percentages using Breakthrough time points.

and never goes below detection. ETR represents when the viral load drops below detection and does not rebound. Breakthrough represents when the patient's viral load drops below detection, but rebounds. The detection level (red line) represents the LLOQ (15 IU/ml) for this patient data. The Breakthrough patient behavior will not be accurately modeled without robust predictions for what is happening beneath the censoring line. Thus, the EM algorithm described in Chapter 1 is implemented when fitting ETR and Breakthrough viral load profiles. The RBV dosage depends on the patient's body weight and was sometimes modified during treatment due to different symptoms of the patients such as blood thinning. The patients experiencing PVR and Breakthrough had constant RBV dosage for the entire treatment while the patient exhibiting ETR had modified dosage. The RBV efficacy is fixed to $\rho = .1222$ from [5] for the PVR and Breakthrough patient. The efficacies for the ETR patient were modified based on time, t, in days since initial treatment. are presented in Table 3.3.

Parameter	$t \leq 27$	$27 < t \leq 83$	$t > 83$
ρ	.5127	.3185	.219

Table 3.3 Patient ETR's RBV efficacies based on modified dosage.

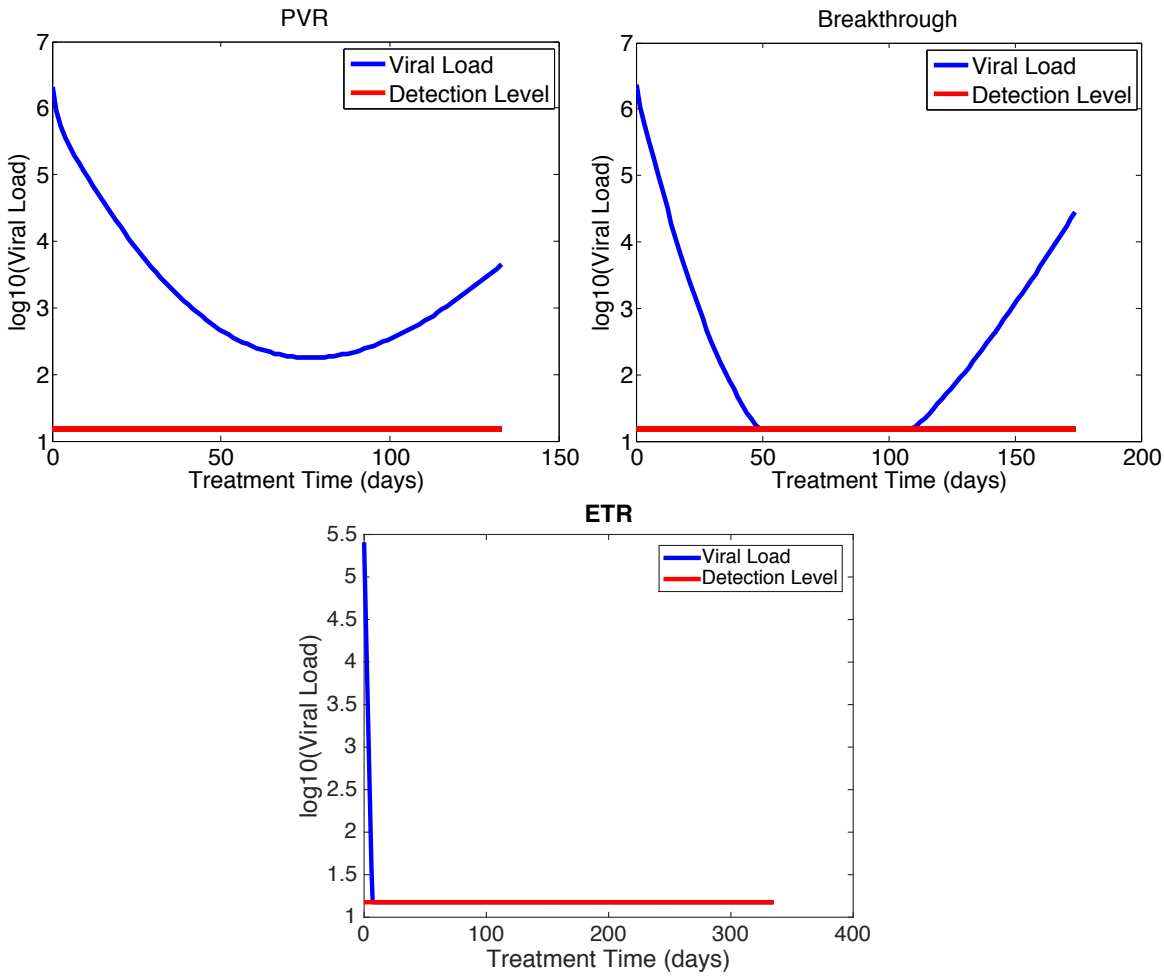

Figure 3.7 Examples of viral load profiles for PVR, ETR, and Breakthrough patients.

Parameter	Value
s	6.17×10^4
r	.00562
$T_{\max}$	1.85×10^7
d	.003
p	25.1
ϵ	.6138

Table 3.4 Fixed parameter values from [95] and [5].

The parameters not in Q_{PVR} or Q_{Brk} are fixed to the values in Table 3.4 from [95] and [5]. As in [95], the infected steady state is used for the initial conditions for (3.5) because the patients considered had chronic infection. The values in Table 3.5 are obtained after estimating the parameters in Q_{PVR} and Q_{Brk}. The estimated parameters are in bold. The PVR parameters have 95% confidence interval half-widths attached. These estimates produce the model fits (graphs on the left) and residuals (graphs on the right) in Figure 3.8. The patient's viral load in the ETR fit goes to zero. The residuals for censored data are set to zero. Moreover, 95% parameter and predictive confidence intervals and prediction intervals for the parameters and predictions are calculated using the asymptotic theory outlined in Section 1.3.4.1, Section 1.3.4.2, and Section 1.3.4.3. The predictive confidence intervals and prediction intervals are shown in Figure 3.9. The prediction intervals are larger than the predictive confidence intervals for the reasons given in Chapter 1.

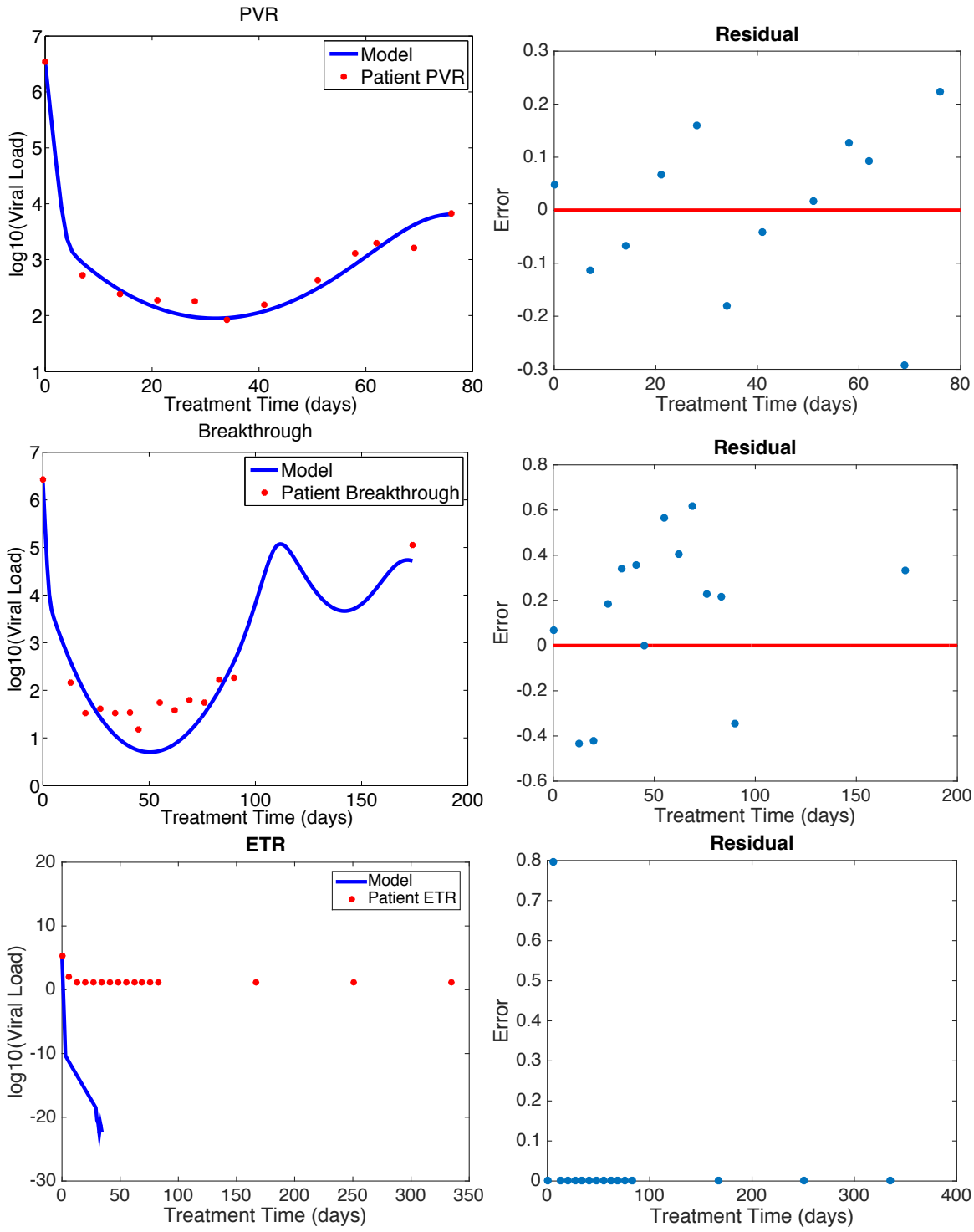

Figure 3.8 Results from parameter estimation for (3.5).

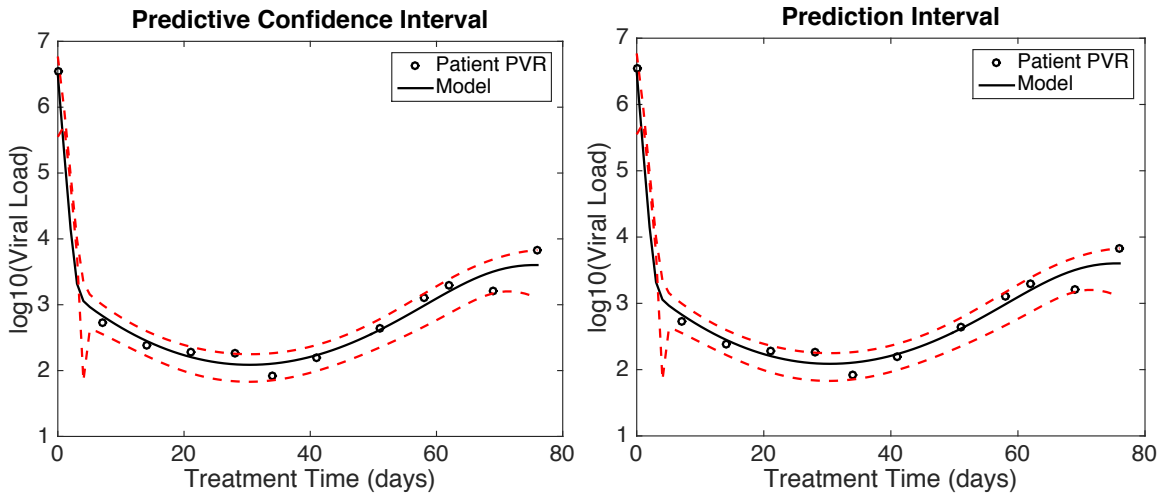

Figure 3.9 Predictive confidence intervals and prediction intervals for (3.5).

Patient	PVR	ETR	Breakthrough
δ	$\mathbf{.1883 \pm .0462}$	**.7211**	**.3293**
c	$\mathbf{2.717 \pm 2.724}$	**11.67**	**2.089**
γ	$\mathbf{.9987 \pm .0015}$	**.9999**	**.6575**
β	$\mathbf{1.875 \times 10^{-5} \pm 1.688 \times 10^{-5}}$	$\mathbf{8.684 \times 10^{-8}}$	$\mathbf{2.259 \times 10^{-6}}$
ϵ	.6138	**.9829**	**.9875**

Table 3.5 Values from parameter estimation for (3.5).

3.4.1 Discussion

The higher values in c and δ in the ETR patient lead us to believe that the immune response along with the drugs have a stronger impact on the mutation and clearance of the virus. It is known that the immune response is strongly correlated with the clearance of the virus. Since the initial conditions of (3.5) are at the infected steady state, introduction of the drugs could be a mechanism to jump start the immune response. We note that even when the virus is not cleared, telaprevir still has a strong impact on viral load decay. This behavior corresponds with how powerful DAA's can be in reducing viral load even when it rebounds. The rebound could be because of mutations which are neglected

in this model as stated earlier. There is a dip at around the 150th day in the Breakthrough response that is unquantifiable due to lack of information regarding the other 3 states or a dynamic immune response. However, this type of dip is observed in [13, 95] where data is available around this time. We conjecture that this dip is due to the immune response being stimulated by the spike in viral load and infection. In Chapter 4, further inspection identifies the mechanism for this dip. The residuals in the PVR fit in Figure 3.8 seem to be i.i.d. because the errors seem to be randomly distributed and are on both sides of the zero axis. This is unlike the Breakthrough fit which have most of the residuals above the zero axis. The predictive confidence intervals and prediction intervals look almost the same, but the variance is very small ($\hat{\sigma}^2 \approx .25$ from (1.32)) and $\upsilon_0 \approx .9$ from (1.33). Therefore, the terms that differ between the two intervals are given by

$$\hat{\sigma}\sqrt{\upsilon_0} \approx .0593,$$
$$\hat{\sigma}\sqrt{1+\upsilon_0} \approx .0862.$$

Hence the intervals look very close to each other, but are not the same. This is also why the intervals are small.

3.5 Conclusion

The missing data between weeks 12-24, 24-36, and 36-48 for the ETR and Breakthrough patients makes parameter estimation challenging. The predictions would also be more robust if information concerning states T, I, and V_{NI} were available. These issues should be considered when making remarks about the estimations and confidence measures. DAAs were introduced in 2011, so there is not as much data available, but in the future, we hope for a larger quantity of data to make more precise estimations.

This chapter describes a model for patients with HCV that are treated with IFN, RBV, and telaprevir combination therapy. The development of this model was motivated by the desire for a model that can be validated and calibrated using sensitivity and identifiability techniques while simultaneously incorporating the new DAA, telaprevir. The model can be used to accurately describe patients exhibiting PVR, ETR, and Breakthough.

CHAPTER

4

IMMUNE RESPONSE AND CONTROL

4.1 Introduction

Powerful drugs and optimized treatment schedules are two of the most important components to overcoming an infection from a virus. In Chapter 3 it is observed that the addition of DAA's have a substantial impact on viral load decline. However, due to drug resistance, mutations in the virus and other complications, viral clearance is not a guarantee [60]. Various treatment strategies have been implemented to increase the opportunity of viral clearance. These strategies include 12, 24, 36, and 48 week variations with different combinations of drugs [43, 60, 78]. This chapter will investigate patient-specific drug treatment plans using similar strategies utilized in HIV called structured treatment interruption(STI). STI alters the patient on and off of treatment [1, 7]. STI has also been implemented in patients that were infected simultaneously with HIV and acute HCV [111]. While giving the patient a break from drug therapy, viral load will usually rebound resulting in more infection and stimulating the adaptive immune response. The goal of STI is to utilize the adaptive immune response in conjunction with the drugs to mount a stronger defense against the virus. The model developed in the previous chapter needs to be adjusted to observe an adaptive immune response dynamic. The next section will provide background on the immune response and its functions in the liver. Section 3 will introduce a revised version of (3.5) that incorporates an adaptive

immune response dynamic. In Section 4, subset selection is used to decide which parameters will be estimated. Then, Section 5 will present parameter estimation results for the patients used in Chapter 3. A discussion of the estimation results and a comparison of the models developed in this chapter and Chapter 3 is also performed. Section 6 will give the control formulation and describes how a control is used to give treatment strategy using STI. Finally, concluding remarks are given in Section 7.

4.2 Immune Response

The defense of the body is known as the immune system. It defends the body from viruses, bacteria, and anything that it considers foreign or a danger to our health. The immune response is responsible for clearing the virus on its own in about 15% of patients [108]. The rest of the patients develop chronic infection. There are two types of immune response: the innate response and the adaptive response. There is an initial rapid and early peak of viral replication and then about 4 to 6 weeks later, HCV RNA increases slowly or stabilizes [15]. The innate response is the first to react.

4.2.1 Innate Immune Response

The innate immune response triggers within the first two days of HCV infection. It causes the body to produce IFN-α which is a protein that acts to inhibit the production of virus in the infected cells and release NK(Natural Killer) cells. NK cells release chemicals that promote programmed cell death, produces IFN-α, and also mediates the intrahepatic recruitment of inflammatory cells. The initial production of IFN-α may slow the virus replication, but doesn't block it [15]. That delay, however, aids in inhibiting production of the virus. The infected hepatocytes will produce IFN-β which induces an anti-viral state that extends to non-infected neighboring cells and provides an initial line of defense against infection [84]. This response reacts quickly because it behaves in a non-specific way. The adaptive response differs in that the immune effectors are virus specific.

4.2.2 Adaptive Response

The adaptive response is stimulated by significant increases in infection. It can take over 1-2 months for HCV to be responded to by this response. This could be a significant reason for why the immune system has a hard time clearing the virus. There are two types of adaptive immune response: the humoral response and the cellular response [15].

4.2.2.1 Humoral Response

It usually takes the humoral response 8-20 weeks to respond to the infection which is much later than the cellular response [84]. The humoral response is the body(B-cells) releasing a melange of antiviral antibodies to fight off the HCV infection [16]. If cleared naturally, the infection is cleared within 2-6 months. The humoral response is important in chronic infection because it tends to control viral load by continuing to put pressure on the infected cells. An example being by inducing apoptosis on infected cells. During the severe phase of HCV infected patients, defects in the humoral response are seen because limited quantities of antibodies are detected [15, 84]. Since the humoral immune response is so late, if the virus is going to be removed it will mostly be because of the T-Cell response.

4.2.2.2 T-Cell Response

This cellular response can be detected between 5-9 weeks [84]. There are two main types of T-Cells that are prominently working in this response: $CD4^+$ and cytotoxic lymphocytes(CTLs), specifically $CD8^+$. $CD4^+$ cells send signals to activate the body's immune response after communicating with antigen-presenting cells(infected cells). They are responsible for activating and enhancing B-Cells and CTLs, respectively. HCV-specific CTLs are also stimulated by the antigens in the infected cells and eliminate the problem via rupturing the cell walls of the infected cells. It is conjectured that during the severe phase of HCV-infected patients the CTL cells are defective in the sense that they have impaired production of inteferon-α, low perforin content, defective capacity for expansion, and perform lysis on healthy cells. Possible reasons for the defectiveness could be from the fact that the infection is attacking the liver cells so the CTL cells that are getting infected are being killed off, the liver wall cells aren't allowing the proliferation of the $CD4^+$ cells, and the production of viral proteins having immunomodulatory effects [15, 84]. There are also what's called regulatory T-cells($CD4^+$ $CD25^+$) that control the immune response in the sense that they suppress immunological response against self and foreign antigens. This could be a reason why during chronic infection the immune response is much lower. The reason is that the cells will limit the amount of immune mediated liver damage [16]. Infected cells release alanine aminotransferase(ALT) when they become damaged. From [14], it is concluded that there may not be a correlation between ALT levels and how much viral load the body has. However, the higher the ALT levels, there is a significant increase of $CD4^+$ $CD25^+$ cells [18]. This makes sense since these cells try to control the liver damage the CTL cells are performing. It seems that when the virus is cleared, there is an abundance of CTL cells, but if it is not cleared then it is because there are not enough CTL cells that are circulating and destroying infected cells. The model in the next section attempts to describe the adaptive cellular response.

4.3 Model

There are several mathematical models and studies that describe immune responses to viral dynamics [8, 76, 108]. The data used is for patients experiencing chronic HCV. Therefore, the goal of this research is to use STI to jumpstart the adaptive immune response and observe how it affects the viral load. Recall from the previous section that the CTL response is stimulated by the infected cells. To this end, an approach given in [76] and similar to [108] is used that models the adaptive response based on the amount of infected cells to give the following revised version of (3.5)

$$\begin{aligned}
\dot{T} &= s + rT\left(1-\frac{T+I}{T_{\max}}\right) - dT - \beta V_I T,\\
\dot{I} &= \beta V_I T + rI\left(1-\frac{T+I}{T_{\max}}\right) - \delta I - k_I I E,\\
\dot{V}_I &= (1-\bar{\rho})(1-\bar{\epsilon})(1-\bar{\gamma})pI - cV_I,\\
\dot{V}_{NI} &= \bar{\rho}(1-\bar{\epsilon})(1-\bar{\gamma})pI - cV_{NI},\\
\dot{E} &= k_E I E - d_E E,
\end{aligned} \tag{4.1}$$

where E is the concentration of immune effectors, k_I is the rate in which the immune effectors kill the infected cells, k_E is the rate in which the immune effectors are stimulated by the infected cells and d_E is the rate in which the immune effectors die. The new parameters in the model, $\{k_I, k_E, d_E\}$, are not known from literature for this model. The treatment schedule and all other parameters values will be used from the PVR, ETR and Breakthrough patients in Chapter 3. As before, the untreated steady states are analyzed and existence and uniqueness of solutions is proven.

4.3.1 Existence and Uniqueness

The nonlinear terms $k_I I E$ and $k_E I E$ are the new concerns that arise in (4.1). However, they can be described by (3.8) in the saturated model. Therefore, the same proof as presented in Chapter 3 is used to obtain existence and uniqueness of solutions from a global Lipschitz condition. This allows us to proceed with the steady state stability analysis.

4.3.2 Steady States and Stability

As in Chapter 3, we want to analyze the steady states when there is no treatment. We recall that we find these equilibria by setting the right side of (4.1) equal to zero and solving for the states. It is observed that $\dot{E} = 0$ implies $E = 0$ or $I = \frac{d_E}{k_E}$. If $E = 0$ then the uninfected and infected steady states

are the same as (3.10) and (3.11), respectively. If $I = \frac{d_E}{k_E}$ the infected steady state is given by

$$\begin{aligned} T &= \frac{H + \sqrt{H^2 + 4F}}{2F}, \\ V &= \frac{p d_E}{c k_E}, \\ E &= \frac{k_E}{d_E * k_I}\left(\frac{\beta p d_E}{c k_E} T + \frac{r d_E}{k_E}\left(1 - \frac{T + \frac{d_E}{k_E}}{T_{max}}\right) - \frac{\delta d_E}{k_E}\right), \end{aligned} \tag{4.2}$$

where

$$\begin{aligned} H &= r - \frac{r d_E}{k_E T_{max}} - d - \frac{\beta p d_E}{c k_E}, \\ F &= \frac{r s}{T_{max}}. \end{aligned}$$

The physically relevant uninfected and infected steady state values for each patient response are given in Table 4.1, Table 4.2, and Table 4.3. After analyzing the eigenvalues of the Jacobian matrix for (4.1) given by

$$J_I = \begin{bmatrix} r(1-\frac{T+I}{T_{max}}) - \frac{rT}{T_{max}} - d - \beta V_I & -\frac{rT}{T_{max}} & -\beta T & 0 & 0 \\ \beta V_I - \frac{rI}{T_{max}} & r(1-\frac{T+I}{T_{max}}) - \frac{rI}{T_{max}} - \delta - k_I E & \beta T & 0 & -k_I I \\ 0 & p & -c & 0 & 0 \\ 0 & p & 0 & -c & 0 \\ 0 & k_E E & 0 & 0 & k_E I - d_E \end{bmatrix}, \tag{4.3}$$

the stability of the two steady states is the same as before with the uninfected being unstable and the infected being stable. The new parameters are set to $k_I = 1$, $k_E = 1 \times 10^{-5}$, and $d_E = .2$ for this analysis.

State	Uninfected Steady State	Infected Steady State
T	1.92×10^7	1.7824×10^4
I	0	2×10^4
V_I	0	1.848×10^5
V_{NI}	0	0
E	0	2.905

Table 4.1 Steady state values using PVR parameters.

State	Uninfected Steady State	Infected Steady State
T	1.92×10^7	1.2524×10^7
I	0	2×10^4
V_I	0	4.302×10^4
V_{NI}	0	0
E	0	1.62

Table 4.2 Steady state values using ETR parameters.

State	Uninfected Steady State	Infected Steady State
T	1.92×10^7	1.1420×10^5
I	0	2×10^4
V_I	0	2.4031×10^5
V_{NI}	0	0
E	0	2.776

Table 4.3 Steady state values using Breakthrough parameters.

In Figure 4.1 it is observed that small changes in the immune response, E, can have significant impact on the infected cells and viral load. This is noteworthy for when STI treatments are discussed later.

4.4 Subset Selection

The same algorithms and techniques that were presented in Chapters 1 and 3 are implemented in this section. Indeed, it is of interest to know the additional uncertainty the three new parameters, $\{k_I, k_E, d_E\}$, bring to (3.5). This uncertainty can lead to different subsets of parameters being both sensitive and identifiable in the local analysis.

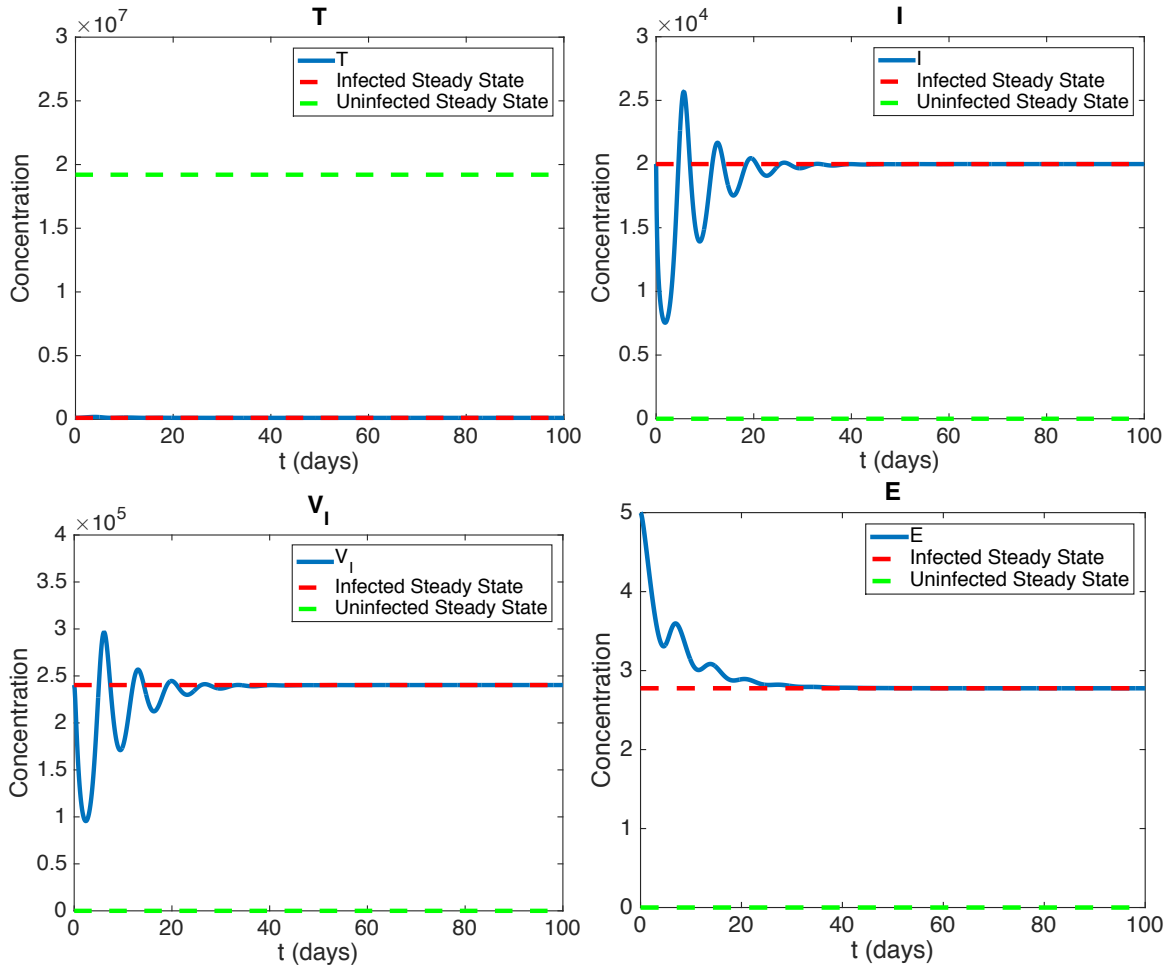

Figure 4.1 Simulation of (4.1) with initial conditions $[T_0, I_0, V_{I0}, E_0] = [1.142 \times 10^5, 2 \times 10^4, 2.403 \times 10^5, 5]$ and using parameter values used in Table 4.3.

4.4.1 Sensitivity Analysis Results

The parameter vector for (4.1) is

$$q_S = (r, \delta, c, \beta, \rho, \epsilon, \gamma, k, k_I, k_E, d_E). \tag{4.4}$$

The sensitivity rankings are given in Figure 4.2 and Figure 4.3 for over 2000 parameter sets. Error bars that are 2 standard deviations from the mean are included. It is hard to discern the cutoff for which parameters are considered sensitive in Figure 4.3 because unlike in Figure 4.2, there isn't an obvious large change in magnitude between the rankings. Thus, an analytical approach similar to the techniques presented in [47] is employed. The method finds the lowest ranked parameter in which a x% perturbation in the parameter poses a significant change in the output. A 25% perturbation is used for this analysis. It is clear that k_I is not impactful because its sensitivity is nearly zero so the method is started with k. It influences the output enough that it is included in Q_{Brk} for the identifiability analysis.

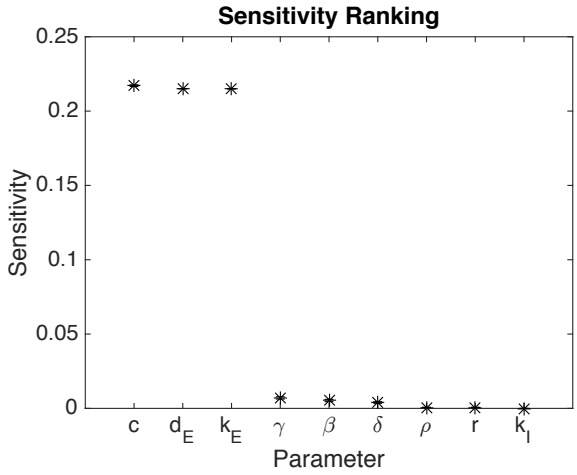

Figure 4.2 Sensitivity rankings using PVR time points.

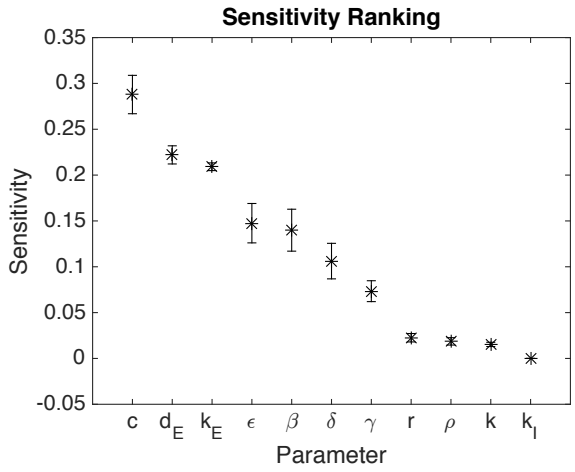

Figure 4.3 Sensitivity rankings using Breakthrough time points.

The set of sensitive parameters that is considered for the identifiability analysis using PVR and Breakthrough time points is given by $Q_{PVR} = \{c, k_E, d_E\}$ and $Q_{Brk} = \{r, \delta, c, \beta, \rho, \epsilon, k, \gamma, k_E, d_E\}$, respectively.

4.4.2 Identifiability Analysis Results

The final subsets to estimate are $Q_{PVR} = \{c, k_E, d_E\}$ and $Q_{Brk} = \{r, \delta, c, \beta, \epsilon, \gamma, k_E, d_E\}$. They are obtained by observing in Figure 4.4 and Figure 4.5 which sensitive parameters are identifiable at least 50% of time.

4.5 Parameter Estimation

The WSSE given in (3.16) is used to determine the parameters in Q_{PVR} and Q_{Brk} using the same schemes as in Chapter 3. From [108], k_I is fixed such that $k_I = 1$ and other parameters not estimated here are set to the values used in Section 3.4. The estimated parameter values are in bold in Table 4.4. The PVR parameters have 95% confidence interval half-widths attached. These estimates produce the model fits (graphs on the left) and residuals (graphs on the right) in Figure 4.6. The patient's viral load in the ETR fit goes to zero. The residuals for censored data are set to zero. Prediction and confidence intervals given in Figure 4.7 are very similar to the model without the immune response given in Figure 3.9.

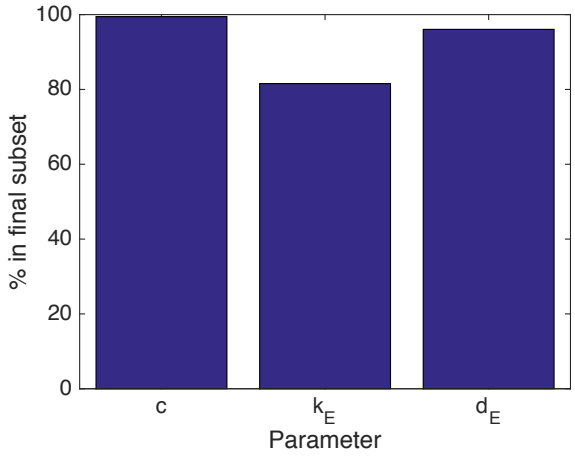

Figure 4.4 Final subset percentages using PVR time points.

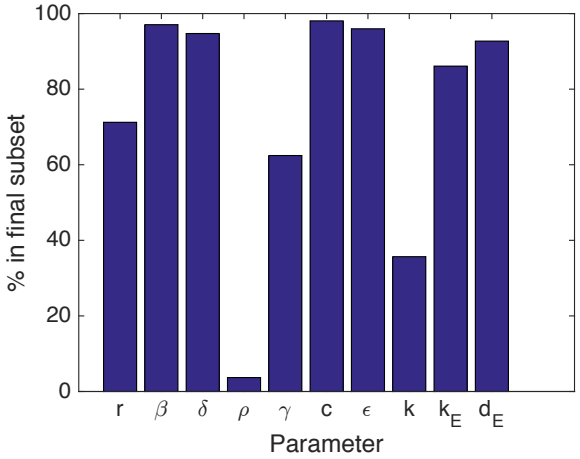

Figure 4.5 Final subset percentages using Breakthrough time points.

Patient	PVR	ETR	Breakthrough
r	5.62×10^{-3}	**.0045**	**.0069**
δ	.1883	**.1729**	**.2229**
c	$\mathbf{1.883 \pm .1848}$	**1.3**	**2.332**
γ	.9987	**.9999**	**.6624**
β	1.875×10^{-5}	$\mathbf{3.945 \times 10^{-9}}$	$\mathbf{1.651 \times 10^{-6}}$
ϵ	.6138	**.4923**	**.9871**
k_E	$\mathbf{4.392 \times 10^{-7} \pm 9.485e \times 10^{-7}}$	$\mathbf{2.645 \times 10^{-6}}$	$\mathbf{1.599 \times 10^{-6}}$
d_E	$\mathbf{.1143 \pm .2291}$	**.0284**	**.2227**

Table 4.4 Values from parameter estimation for (4.1).

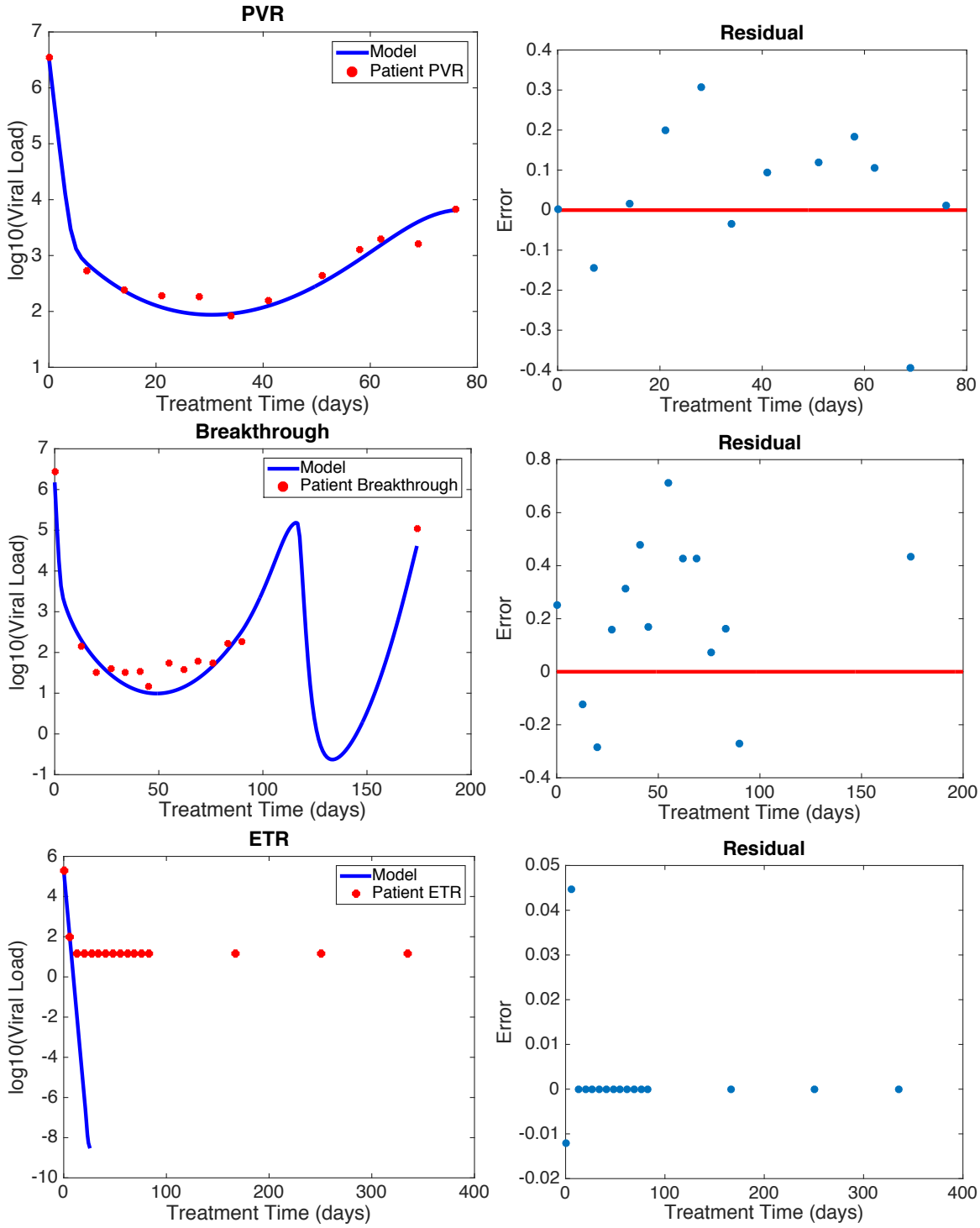

Figure 4.6 Results from parameter estimation for (4.1).

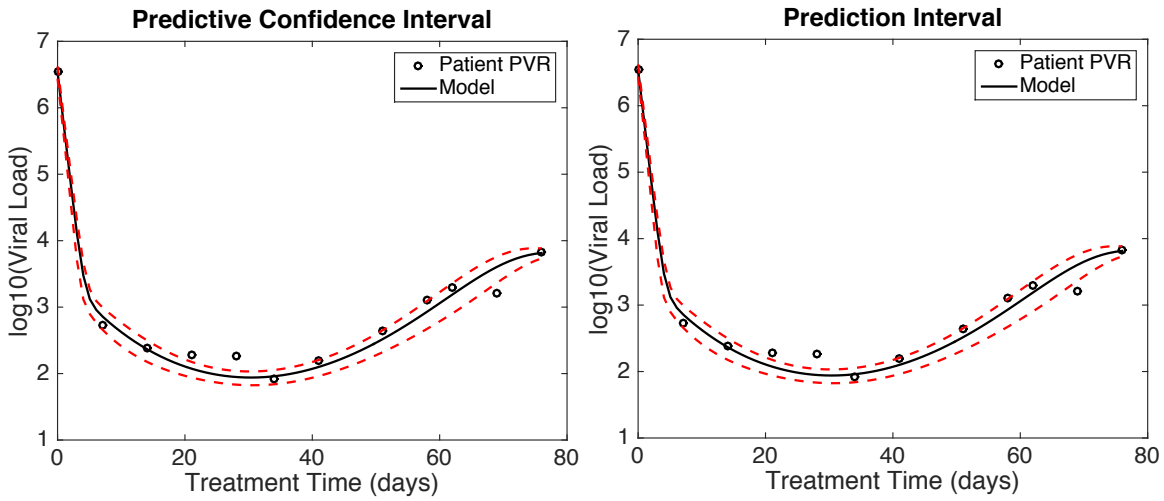

Figure 4.7 Predictive confidence intervals and prediction intervals for (4.1).

4.5.1 Discussion

The addition of the adaptive immune response gives lower values for δ and c than in Chapter 3. This makes sense because more information about the clearance of infected cells by the immune response can be quantified in the term $k_I EI$. Also, note that the dip after the telaprevir treatment ends in the Breakthrough fit is much deeper in Figure 4.6 than in Figure 3.8. Recall that a conjecture was posed that the dip was due to the immune response being ignited by the sudden increase in viral load which spikes the infected cell concentration. Figure 4.8 confirms that when the infected cells rapidly increase, so does the adaptive immune response. In turn, the quantity of infected hepatocytes sharply decreases due to the CTLs eliminating them. None of the residuals in Figure 4.6 seem to be i.i.d. because they don't seem to be randomly distributed. In both the PVR and Breakthrough fits, most of the residuals are above the zero axis. However, the fits in Figure 4.6 are very similar to the fits acquired from the estimations without the adaptive immune response dynamic. Therefore, a comparison between the models presented in this and the previous chapters is considered.

4.5.2 Akaike Information Criteria

To decide if the model in (3.5) or (4.1) describes a behavior of the system better, the Akaike Information Criteria(AIC) is used as in [47]. Developed in 1979, it measures the amount of information lost when a model is used to describe the dynamics of a system. If two or more models are used to describe the same behavior, the model with the lowest AIC value is favored. The AIC score is

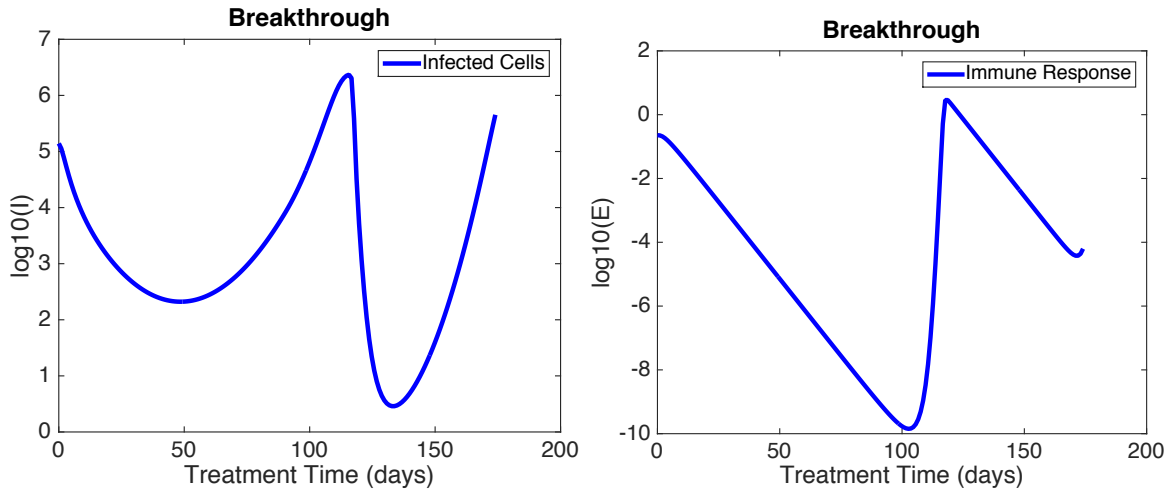

Figure 4.8 Immune response for Breakthrough patient.

computed using the equations

$$AIC = k \cdot \ln(\frac{J}{k}) + 2p,$$
$$J = \frac{1}{max_j(y_j)} \sum_{i=1}^{n} [y_j - y(t_i, q)]^2,$$

where k is the number of data points, p is the number of model parameters, y_j is the data point at time t_j and y is the model output. Table 4.5 shows that (3.5) is a better model for PVR and Breakthrough, but not for SVR. The small differences in the AIC means (4.1) has a better goodness of fit, but is penalized by the term $2p$ for having three more parameters. While (3.5) has a better AIC, it cannot account for the dip in the Breakthrough fit.

Model	PVR AIC	ETR AIC	Breakthrough AIC
No Immune Response Model	-39.37	-54.32	-30.48
Immune Response Model	-33.37	-139.7	-28.61

Table 4.5 The AIC scores for (3.5) and (4.1) for each patient behavior.

4.6 Control

This section strives to devise new drug treatment plans using a control based strategy. It is observed in Figure 4.8 that when treatment ends there will be a spike in the immune response due to the sudden increase in infected cells. This jolt in immune response subsequently drove the viral load to very low levels before it rebounded. The goal is to acquire a control for (4.1) that provides a treatment schedule that will take advantage of the immune response while simultaneously considering viral load and drug intake.

4.6.1 Control Formulation

The drugs that we will be considering have two different dosing regimens. IFN is injected weekly whereas ribavirin and telaprevir are consumed orally daily. Thus, two different controls are used for them. Alongside the mathematical model in (4.1), the control problem with objective function given by

$$J(u_1(t), u_2(t)) = \int_{t_0}^{t_1} [W_V V(t) + W_\epsilon h_1(t) + W_{\gamma\rho} h_2(t) - W_E E(t)] d\,t, \tag{4.5}$$

is considered where

$$h_1(t) = u_1(t)\epsilon,$$
$$h_2(t) = u_2(t)(\rho + \gamma),$$

and $(u_1(t), u_2(t))$ are time discretized vector controls such that $u_1(t)$ is the control for IFN and $u_2(t)$ for ribavirin and telaprevir. The cost that is produced by drug treatment such as unintended side effects and treatment expenses is obtained in these controls. The weights for the virus, controls and immune response are W_V, W_ϵ, $W_{\gamma\rho}$, and W_E, respectively. The controls are binary treatment functions with $u_i(t) = 1$ if the drug being controlled is being taken on that day and $u_i(t) = 0$ if it is not taken. Due to dosing regimens, this means that while ribavirin and telaprevir can be switched on and off daily, IFN is controlled weekly. The size of the control vector depends on the duration the patient is treated and is given in Table 4.6. The treatment times are not strict to the guidelines as presented in the treatment schedule section due to human error.

Patient	Duration of Treatment
PVR	76 days
Breakthrough	174 days
ETR	335 days

Table 4.6 Treatment times obtained from the data provided.

The control vectors are size $1 \times x$ where x is the duration of treatment for that patient. Let Λ denote the set of all such control vectors. The desire is to find the optimal control vector pair (u_1^*, u_2^*) satisfying

$$\min_{u_1, u_2 \in \Lambda} J(u_1, u_2) = J(u_1^*, u_2^*) \tag{4.6}$$

subject to (4.1) and J is defined by (4.5). In [26], a continuous control function using Pontryagin's Maximum Principle is derived for patients being treated with IFN and RBV. In practice, a continuous control is difficult to implement because treatment is prescribed at discrete intervals. Therefore, a more realistic control is considered using the subperiod method as described in [1] and summarized next.

4.6.2 Subperiod Method

The number of elements in Λ is finite, thus an optimal control vector pair is guaranteed for (4.6). One way to find the optimal vector pair is by direct search and comparison. The description of this idea is given in algorithm (4.1). Recall that the size of each binary control vector will depend on the

Algorithm 4.1

1. Let J_{end} be empty.
2. Choose 2 vector controls $u_c = (u_1^k, u_2^l) \in \Lambda$.
3. Solve (4.1) using u_c.
4. Retrieve cost J_c from (4.5) using u_c and results from step 3.
5. If J_{end} is empty or $J_c < J_{end}$, set $J_{end} = J_c$ and $u_{end} = u_c$.
6. Repeat 2.-5. for any two vectors in Λ and u_{end} will contain a solution to (4.6).

number of days of treatment for the patient. This implies that the smallest number of components for each control vector will be 76. This forces Λ to have at least 2^{76} elements. If all drugs are taken daily, $(2^{76})^2$ different combinations of vectors are considered. This large amount of elements make it computationally intractable to consider daily changes. The issue is partially alleviated due to IFN being taken weekly. However, this still leaves the number of iterations to be greater than 2^{86}.

The subperiod method, similar to the underlying idea for dynamic programming, seeks to reduce the computational burden by finding optimal policies on subintervals of the duration of treatment. That is, instead of considering controls for the full treatment interval, controls are acquired for subperiods such as [0,28), [0,56), [0,76]. Given a full treatment interval of 76 days, assume 28 day subperiods are used with seven day treatment segments. The optimal pair for the interval will be denoted by u^*_{ij} where i is the control and j is the subperiod. Since seven day segments are used, $u^*_{i,j}$ can be shortened so that it has size 1×4 where each component represents seven days. For example, if $u^*_{i,j} = [1,1,0,0]$ then full treatment is used on days [0,14) and off for the rest of the interval. The control vectors will have the form

$$u_1(t) = \begin{bmatrix} u^*_{1,1} & u^*_{1,2} & u^*_{1,3} \end{bmatrix},$$
$$u_2(t) = \begin{bmatrix} u^*_{2,1} & u^*_{2,2} & u^*_{2,3} \end{bmatrix}.$$

The method begins by obtaining an optimal pair $(u^*_{1,1}, u^*_{2,1})$ for the first interval [0,28). An optimal control law can be attained quickly because only $(2^4)^2 = 256$ comparisons need to be analyzed. After completion of this step, $u_i(t)$ is appended with $u^*_{i,j}$ and the interval is extended to [0,56). The second step is to attain $(u^*_{1,2}, u^*_{2,2})$ that solves (4.6) for the new subperiod. $(u^*_{1,1}, u^*_{2,1})$ is still employed for the interval [0,28), so attaining the new control policies will still require $(2^4)^2 = 256$ iterations. The new control law is appended the same as before and the interval is extended. Suppose the control policy extends past the subperiod that is being considered. In this case, the control pair will only be optimized until the end of the subperiod. The method is complete after the full treatment interval has been controlled. Indeed, this method will result in a suboptimal control in general. However, it is shown in [2] that the results are reasonable approximations to the more robust, but impractical continuously controlled therapy.

4.6.3 Numerical Simulations

Simulations are executed using 28 day subperiods with 7 day segments for $u_{1,*}$ and 3 day segments for $u_{2,*}$. This means each bin will have $(2^4)(2^9) = 2^{13} = 8,192$ comparisons. As in dynamic programming, the subperiod method is heavily dependent on the initial conditions of the state system. Recall that

the patients are dealing with chronic infection, so the initial conditions are set to the infected steady state. The weights $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\}$ are also a major factor in the results of the method. The weights are varied to observe outcomes where different parts of (4.1) are emphasized. Since the subperiod method uses bins instead of the entire treatment interval, similar emphasis on components with different weights can result in a better outcome for the patient than the other going forward in time. A discussion is given in the next few sections about the simulated results using the subperiod method for each patient. In the remaining figures, the legend is organized as follows for each patient and state (T, I, V, E):

- All Med Model (blue dots) - The state model fit with treatment being administered as prescribed in Figure 3.2.
- OptModel (black line) - The state model fit with treatment being administered as prescribed by the subperiod method.
- No Med Model (green dots) - The state model fit with no treatment being administered (infected steady state).
- Control On (cyan dots) - This indicates that full treatment is being taken on that day.
- Control Off (magenta dots) - This indicates that no treatment is being taken on that day.

The control for IFN ($u_1(t)$) is described at the bottom of each figure. The control for telaprevir and ribavirin ($u_2(t)$) is described at the top of each figure.

4.6.3.1 PVR Patient

In Figure 4.9 and Figure 4.10 the subperiod method found that the best strategy was for the patient to be continuously treated for the entire duration. The viral load is emphasized in Figure 4.10, but the subperiod method could not find a better treatment schedule to reduce the viral load. If the immune response is heavily weighted as in Figure 4.11 then less treatment is used, but the viral load is elevated. This is not desired because the patient is not getting healthier. Thus, it is advised that this type of patient be fully treated.

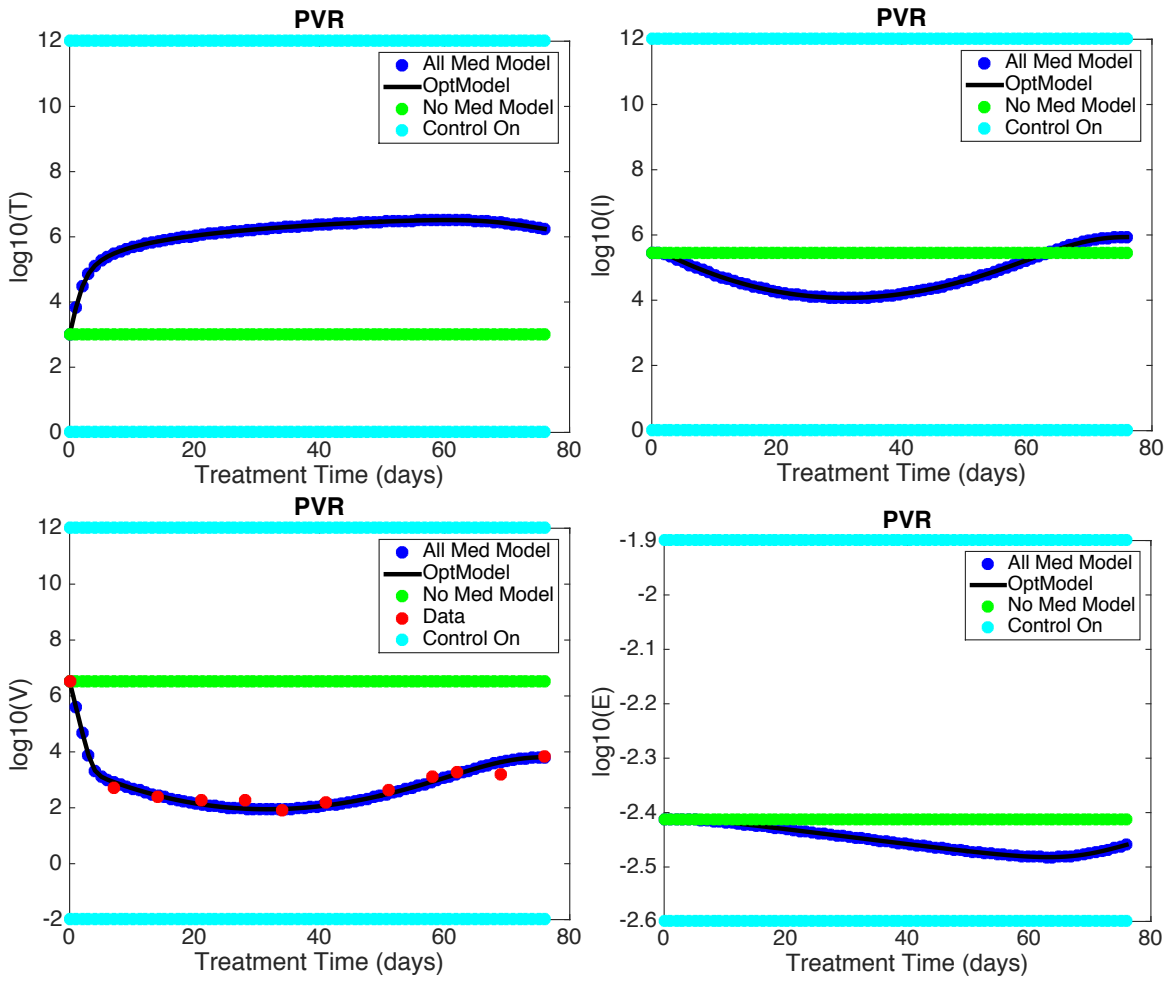

Figure 4.9 The weights for this implementation of the subperiod method on the PVR patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [1, 0, 0, 1]$

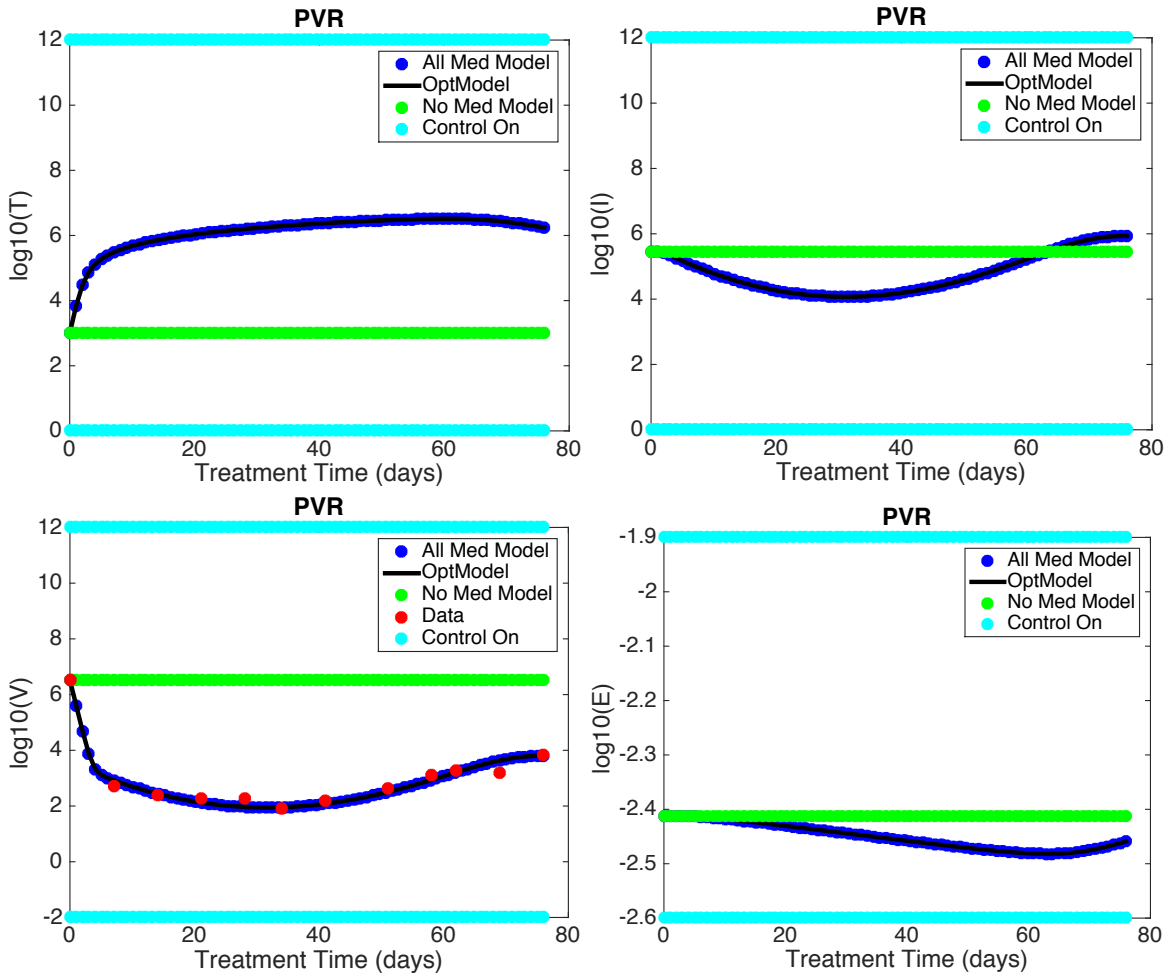

Figure 4.10 The weights for this implementation of the subperiod method on the PVR patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [10, 1, 1, 1]$

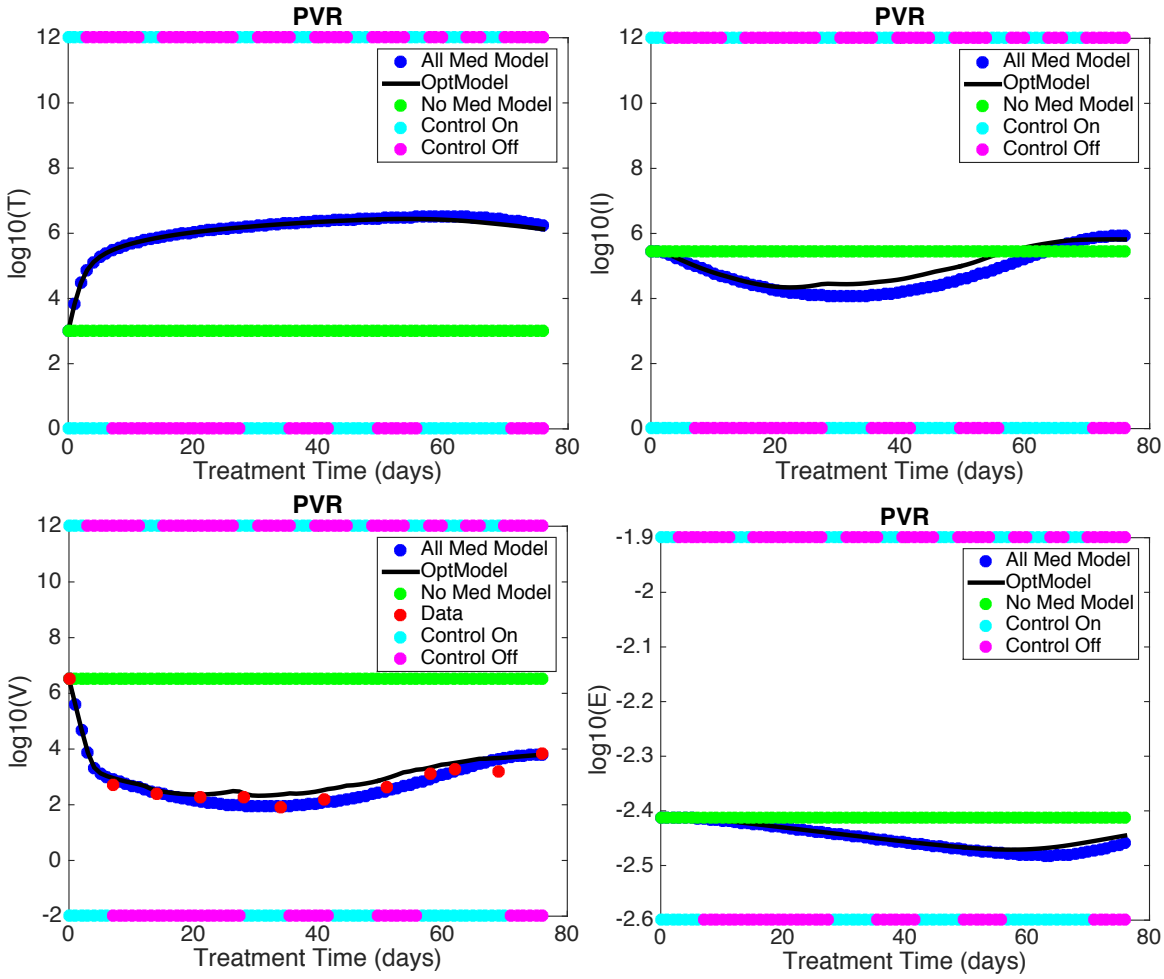

Figure 4.11 The weights for this implementation of the subperiod method on the PVR patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [.001, 1, 1, 100]$

4.6.3.2 Breakthrough Patient

The weights in Figure 4.12 and Figure 4.13 are chosen to put emphasis on minimizing viral load. The optimized viral load result is the same, but less drugs are prescribed in Figure 4.13 than in Figure 4.12 using the subperiod method. This is a consequence of setting $W_\epsilon = W_{\gamma\rho} = 1$ instead of zero. The adaptive immune response is emphasized in Figure 4.14 and has less drug treatment than the previous two figures. The viral load oscillates, simulating the STI effect that is desired. However, the viral load is elevated from the previous figures and will gravitate back towards the infected steady state since it is a stable equilibrium.

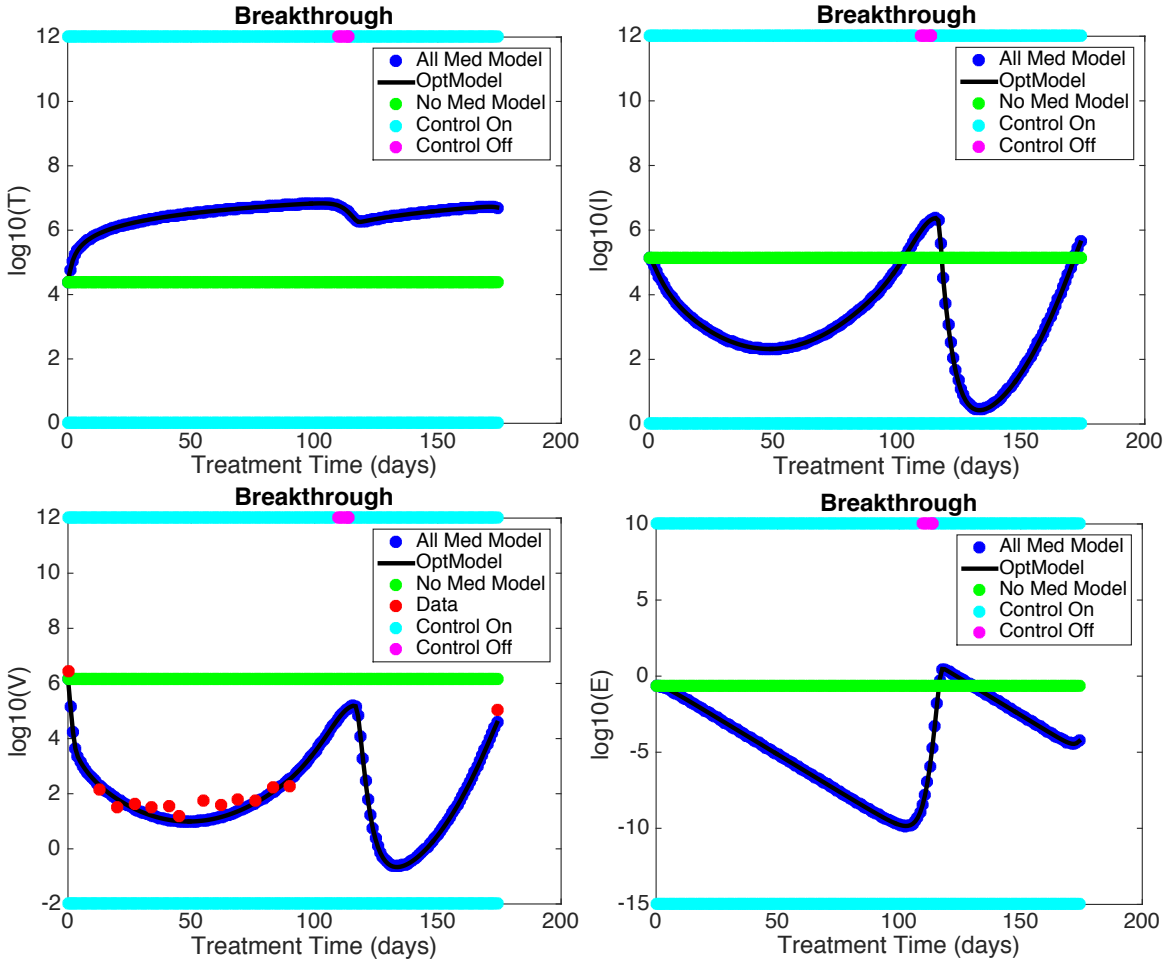

Figure 4.12 The weights for this implementation of the subperiod method on the breakthrough patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [1, 0, 0, 1]$

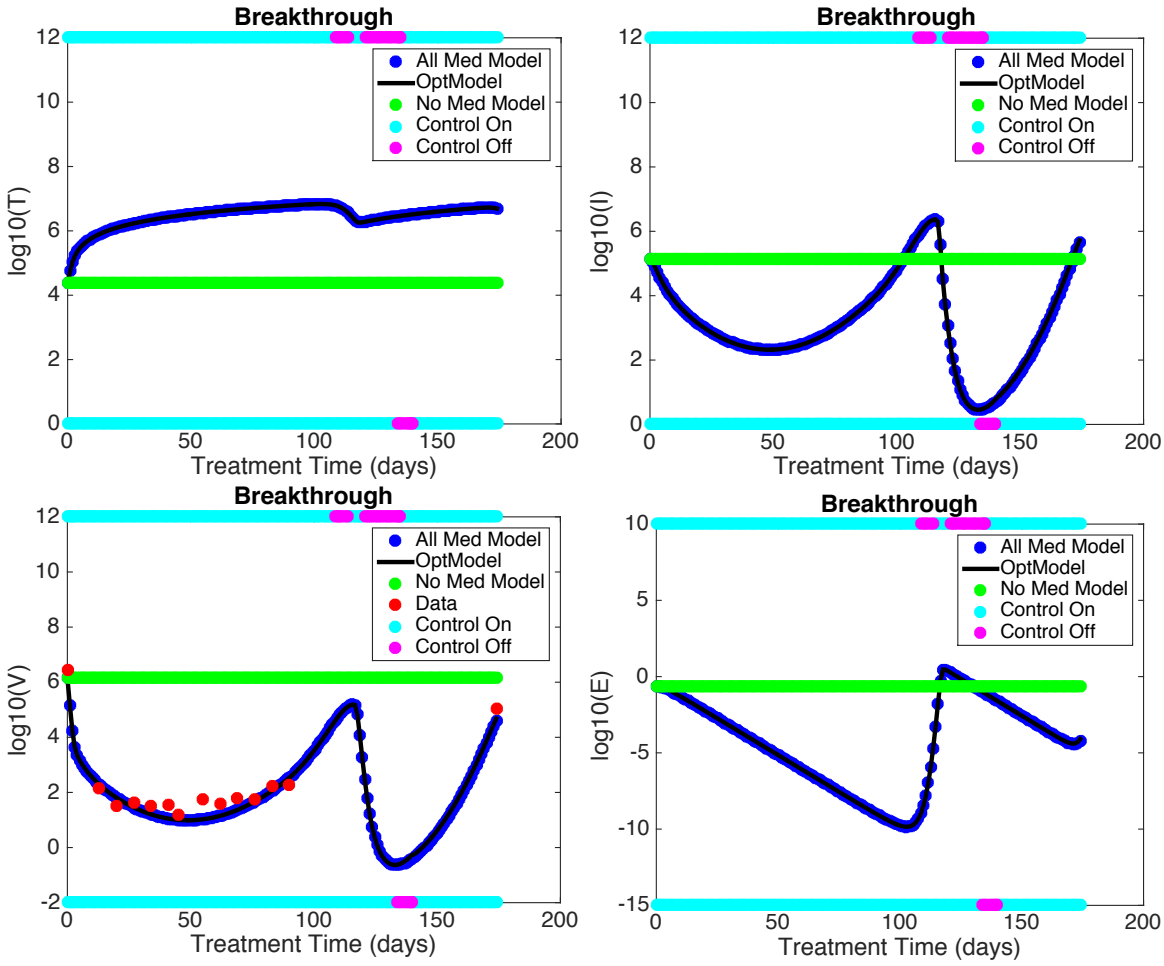

Figure 4.13 The weights for this implementation of the subperiod method on the breakthrough patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [10, 1, 1, 1]$

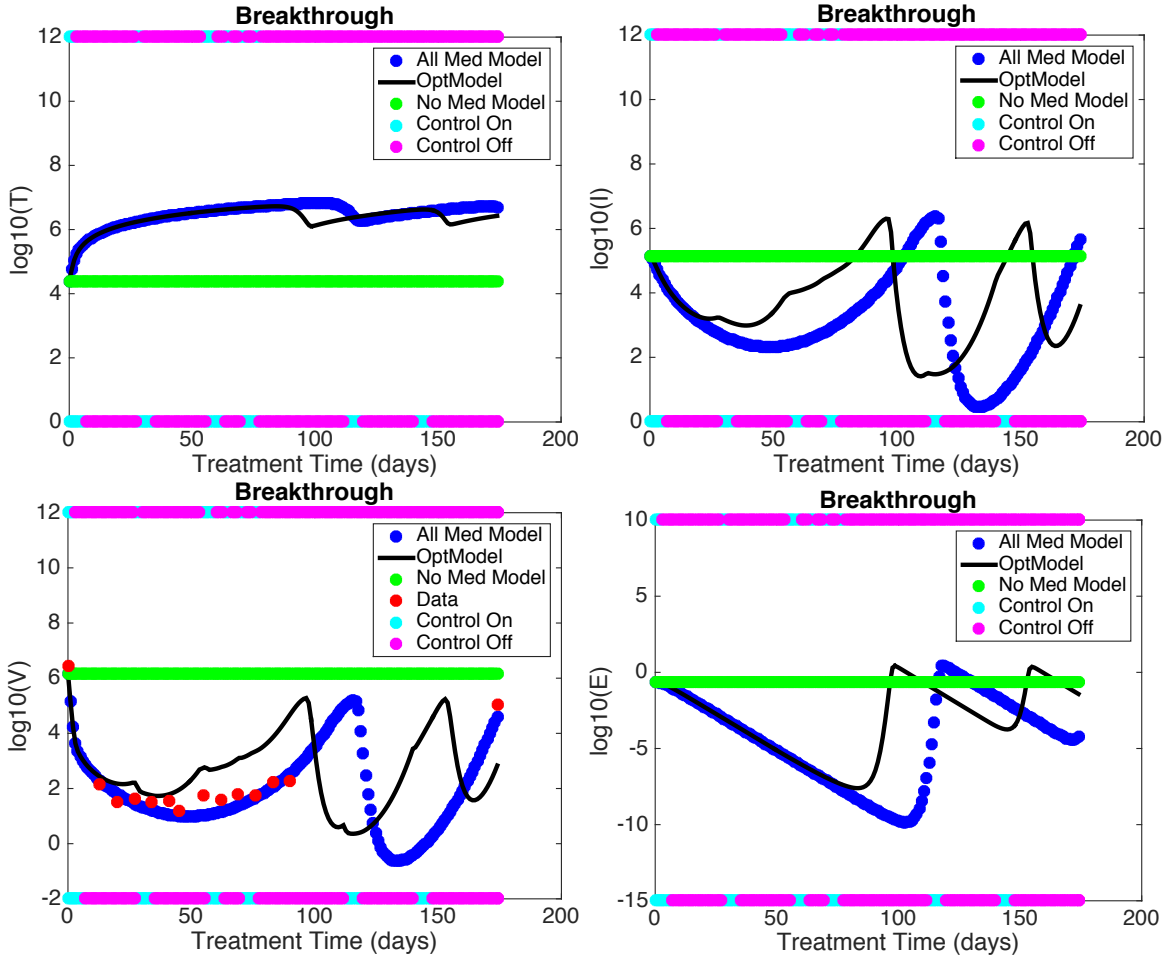

Figure 4.14 The weights for this implementation of the subperiod method on the breakthrough patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [.001, 1, 1, 100]$

4.6.3.3 ETR Patient

Figure 4.15 shows that the ETR patient's viral load is cleared within the first week of treatment. It never shows a spike in the immune response, so it can be concluded that the drugs are primarily responsible for clearance. The combination of drugs are only needed for the first 3 and 7 days, respectively.

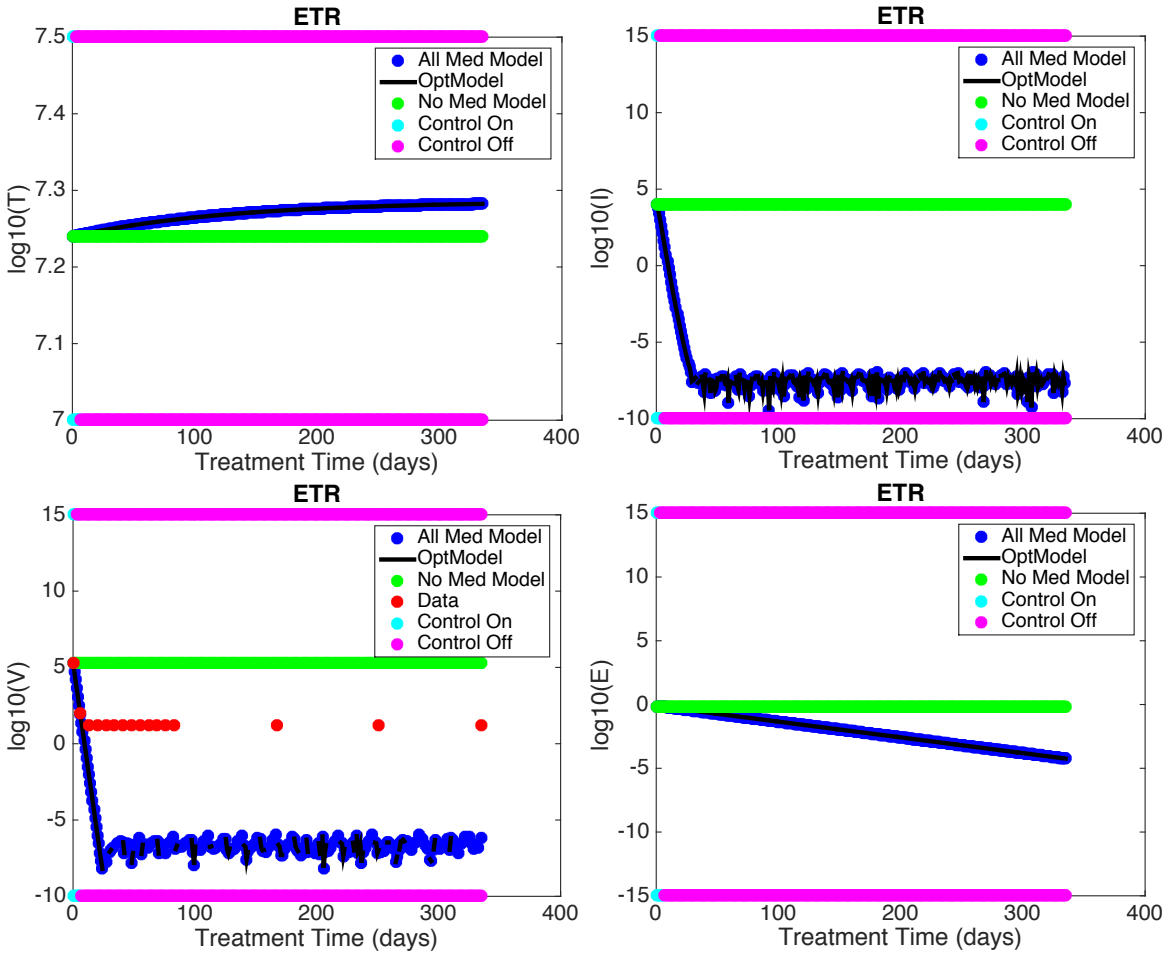

Figure 4.15 The weights for this implementation of the subperiod method on the ETR patient are $\{W_V, W_\epsilon, W_{\gamma\rho}, W_E\} = [1,1,1,1]$

4.7 Conclusion

The revised model presented in this chapter provides a more complete system of differential equations to quantify the dynamics of hepatitis C with the triple drug combination therapy of telaprevir, IFN, and ribavirin. In [75, 95] and the model presented in Chapter 3, the immune response effect is constant and does not account for fluctuations due to treatment. The added dynamic to account for the adaptive immune response allows the researcher to further quantify different aspects of the system. It is clear that the adaptive response is stimulated by spikes in infection and subsequently is a mechanism for driving the infection and viral load down.

A realistic suboptimal STI method is employed that searches for the most advantageous quantity of drug holidays to minimize the viral load and systemic costs associated with treatment. The method attempts to take advantage of stimulations of the immune response to minimize drug intake. It has been shown that the infected steady state is stable and uninfected steady state is unstable. Thus, in the long term, since the initial conditions are at the infected steady state, unless all virions and infection is cleared simultaneously, the patient viral load will always be driven back to the infected steady state. It is clear from Figure 4.15 that if the virus is cleared then it is cleared immediately by the drugs driving the viral load and infected cells to zero. Thus, the benefits of the treatment plan using the STI method presented lies in the alleviation of potential negative consequences that come with constant drug treatment. This includes treatment cost that can be on the upside of $\$83K$ [17] and side effects such as nausea/vomiting and life-threatening skin reactions.

CHAPTER

5

CONCLUSION AND FUTURE WORK

This dissertation discusses showcases analytical and numerical tools applied to real world problems. These applications range from the field of high energy physics to the area of biomedical sciences. The algorithms that are utilized are well defined and researched to answer the questions posed in this work.

Deterministic sampling and optimization are applied to the process of klystron designing. Simulation software provided by klystron engineers in conjunction with novel algorithms programmed in MATLAB are used to automatically design previously unattainable klystron builds. A minor drawback to the algorithm presented is that there isn't an analytical system that is used during the optimization process. The simulation software is a black box that is utilized mainly to input sampled parameters and retrieve outputs for the objective function. Therefore, it is not well known if there are better builds that can be attained due to the lack of convergence theory for sampling methods in higher dimensions. This problem is mitigated due to industry concerns being primarily directed at meeting customer demands. Thus, while convergence is not guaranteed, as long as the minimum specifications are achieved then the procedure is a success. In any case, the automatic process is assured to perform better than human trial and error. The plan is to generalize our optimizer scheme so that one may easily implement the concept described in chapter 2 for any klystron or klystron

simulation software. This includes enhancing our program to allow additional input parameters to be optimized, use public domain software for the deterministic sampling methods and incorporate a graphical user interface.

Personalized medicine is a field geared to enhance the patient experience with specialized treatment tailored towards their specific needs. A patient specific mathematical model is developed for describing the dynamics of hepatitis C with triple drug combination therapy including a DAA. The first installment of the model given in (3.5) is adapted from a previously established system that already incorporated IFN and ribavirin to include the DAA, telaprevir. The inverse problem is solved to validate and calibrate the model and it's parameters. Indeed, it is revealed that the system is able to accurately fit data from patients exhibiting PVR, ETR, and Breakthrough. However, there was some phenomenon that could not be quantified due to lack of information as to how the adaptive immune response is affected by drug treatment. Thus, the amended model presented in (4.1) provides more clarity with the addition of an immune response dynamic. As a final exercise, a suboptimal routine is implemented to provide patient specific treatment regimens using STI. The solutions give promise that better treatment schedules can be provided. An intrinsic question that arises is how the observations made in this work can be extended to provide optimal drug treatment schedules in a clinical setting. Indeed, while the research here is patient specific, a population based model would be more appropriate for further study. This will allow for a more probabilistic assessment allowing clinicians to have a better idea which patients are more akin to exhibit a particular response to treatment. In turn, optimal treatment strategies similar to the one presented in this work can be administered to specific populations and will heighten opportunities for cure and reducing systemic costs. A substantial increase in data is needed to provide such an approach.

BIBLIOGRAPHY

[1] B. ADAMS, H. BANKS, H.-D. KWON, AND H. T. TRAN, *Dynamic multidrug therapies for HIV: Optimal and STI control approaches*, Mathematical Biosciences and Engineering, 1 (2004), pp. 223–241.

[2] B. M. ADAMS, *Non-Parametric Parameter Estimation and Clinical Data Fitting With A Model of HIV Infection*, PhD thesis, North Carolina State University, July 2005.

[3] B. S. ADIWIJAYA, E. HERRMANN, B. HARE, T. KIEFFER, C. LIN, A. D. KWONG, V. GARG, J. C. R. RANDLE, C. SARRAZIN, S. ZEUZEM, AND P. R. CARON, *A multi-variant, viral dynamic model of genotype 1 HCV to assess the in vivo evolution of protease-inhibitor resistant variants*, PLOS Computatioal Biology, 6 (2010).

[4] B. S. ADIWIJAYA, T. L. KIEFFER, J. HENSHAW, K. EISENHAUER, H. KIMKO, J. J. ALAM, R. S. KAUFFMAN, AND V. GARG, *A viral dynamic model for treatment regimens with direct-acting antivirals for chronic hepatitis c infection*, PLOS Computatioal Biology, 8 (2012).

[5] J. G. ARTHUR, H. TRAN, AND P. ASTON, *Feasibility of parameter estimation in hepatitis c viral dynamics models*, Journal of Inverse and Ill-Posed Problems, (2016).

[6] J. A.S. GILMOUR, *Klystrons, Traveling Wave Tubes, Magnetrons, Crossed-Field Amplifiers, and Gyrotrons*, Artech House, 2011.

[7] A. R. ATTARIAN, *Patient Specific Subset Selection, Estimation and Validation of an HIV-1 Model with Censored Observations under and Optimal Treatment Schedule*, PhD thesis, North Carolina State University, 2012.

[8] R. AVENDANO, L. ESTEVA, J. FLORES, J. F. ALLEN, G. GOMEZ, AND J. LOPEZ-ESTRADA, *A mathematical model for the dynamics of hepatitis c*, Journal of Theoretical Medicine, 4 (2002), pp. 109–118.

[9] H. BANKS, R. BARALDI, K. CROSS, K. FLORES, C. MCCHESNEY, L. POAG, AND E. THORPE, *Uncertainty quantification in modeling HIV viral mechanics*, Technical Report 16, Center for Research in Scientific Computation, December 2013.

[10] H. BANKS, K. BEKELE-MAXWELL, L. BOCIU, M. NOORMAN, AND K. TILLMAN, *The complex-step method for sensitivity analysis of non-smooth problems arising in biology*, Technical Report 11, Center for Research in Scientific Computation, October 2015.

[11] H. BANKS, A. CINTRON-ARIAS, AND F. KAPPEL, *Parameter selection methods in inverse problem formulation*, Technical Report 03, Center for Research in Scientific Computation, November 2010.

[12] H. BANKS, S. L. ERNSTBERGER, AND S. L. GROVE, *Standard errors and confidence intervals in inverse problems: Sensitivity and asscoicated pitfalls*, Technical Report 10, Center for Research in Scientific Computation, March 2006.

[13] H. BANKS AND H. TRAN, *Mathematical and Experimental Modeling of Physical and Biological Processes*, Chapman and Hall/CRC, January 2009.

[14] R. BARALDI, K. CROSS, C. MCCHESNEY, L. POAG, E. THORPE, K. FLORES, AND H. BANKS, *Mathematical modeling of HCV viral kinetics*, tech. rep., Center for Research in Scientific Computation, Raleigh, NC, July 2013.

[15] A. BERTOLETTI AND C. FERRARI, *Kinetics of the immune response during HBV and HCV infection*, Hepatology, 38 (2003).

[16] A. BERTOLETTI AND A. J. GEHRING, *The immune response during hepatitis b virus infection*, Journal of General Virology, 87 (2006), pp. 1439–1449.

[17] K. BICHOUPAN, V. MARTEL-LAFERRIERE, D. SACHS, M. NG, E. A. SCHONFELD, A. PAPPAS, J. CRISMALE, A. STIVALA, V. KHAITOVA, D. GARDENIER, M. LINDERMAN, P. V. PERUMALSWAMI,

T. D. SCHIANO, J. A. ODIN, L. LIU, A. J. MOSKOWITZ, D. T. DIETERICH, AND A. D. BRANCH, *Costs of telaprevir-based triple therapy for hepatitis c: $189,000 per sustained virological response*, Hepatology, 60 (2014).

[18] F. BOLACCHI, A. SINISTRO, C. CIAPRINI, F. DEMIN, M. CAPOZZI, F. CARDUCCI, C. M. J. DRAPEAU, G. ROCCHI, AND A. BERGAMINI, *Increased hepatitis c virus (HCV)-specific $CD4^+CD25^+$ regulatory T lymphocytes and reduced HCV-specific $CD4^+$ T cell response in HCV-infected patients with normal versus abnormal alanine aminotransferase levels*, Clinical and Experimental Immunology, 144 (2006), pp. 188–196.

[19] M. BOTTON, J. THOMAS M. ANTONSEN, B. LEVUSH, K. T. NGUYEN, AND A. N. VLASOV, *Magy: A time-dependent code for simulation of slow and fast microwave sources*, IEEE Transactions on Plasma Science, 26 (1998).

[20] J. BROK, L. L. GLUUD, AND C. GLUUD, *Ribavirin monotherapy for chronic hepatitis c*, Cochrane Databse of Systematic Reviews, (2009).

[21] R. BRUN, M. KUHNI, H. SIEGRIST, W. GUJER, AND P. REICHERT, *Practical identifiability of asm2d parameters: Systematic selection and tuning of parameter subsets*, Water Research, 36 (2002), pp. 4113–4127.

[22] J. BUKH, R. MILLER, AND R. PURCEL, *Genetic heterogeneity of hepatitis c virus: quasispecies and genotypes*, Seminars of Liver Disease, 15 (1995), pp. 41–63.

[23] R. L. BURDEN AND J. D. FAIRES, *Numerical Analysis*, Bob Pirtle, eighth ed., 2005.

[24] M. BURTH, G. VERGHESE, AND M. VELEZ-REYES, *Subset selection for improved parameter estimation in on-line identification of a synchronous generator*, IEEE Transactions on Power Systems, 14 (1999), pp. 218–225.

[25] G. R. CARMICHAEL, A. SANDU, AND F. A. POTRA, *Sensitivity analysis for atmospheric chemistry models via automatic differentiation*, Atmospheric Environment, 31 (1997), pp. 475–489.

[26] S. P. CHAKRABARTY AND H. R. JOSHI, *Optimally controlled treatment stratedy using interferon and ribavirin for hepatitis c*, Journal of Biological Systems, 17 (2009), pp. 97–110.

[27] A. CHATTERJEE, J. GUEDJ, AND A. S. PERELSON, *Mathematical modelling of HCV infection: what can it teach us in the era of direct-acting antiviral agents*, Antiviral Therapy, (2012), pp. 1171–1182.

[28] S. L. CHEN AND T. R. MORGAN, *The natural history of hepatitis c virus (HCV) infection*, International Journal of Medical Sciences, 3 (2006), pp. 47–52.

[29] I. A. CHERNYAVSKIY, A. N. VLASOV, J. THOMAS M. ANTONSEN, S. J. COOKE, B. LEVUSH, AND K. T. NGUYEN, *Simulation of klystrons with slow and reflected electrons using large-signal code tesla*, IEEE Transactions on Electron Devices, 54 (2007).

[30] M. CHODOROW, E. L. GINZTON, I. NEILSEN, AND S. SONKIN, *Design and performance of a high-power pulsed klystron*, Proceedings of the IRE, 41 (1953), pp. 1584–1602.

[31] T. CHOI AND C. KELLEY, *Superlinear convergence and implicit filtering*, SIAM Journal of Optimization, 10 (2000), pp. 1149–1162.

[32] P. G. CIARLET, *Linear and Nonlinear Functional Analysis*, Society for Industrial and Applied Mathematics, 2013.

[33] A. CINTRON-ARIAS, H. BANKS, A. CAPALDI, AND A. L. LLOYD, *A sensitivity matrix based methodology for inverse problem formulation*, Technical Report 09, Center for Research in Scientific Computation, April 2009.

[34] G. CLERMONT AND S. ZENKER, *The inverse problem in mathematical biology*, Mathematical Biosciences, 260 (2014), pp. 11–15.

[35] P. COLOMBATTO, L. CIVITANO, F. OLIVERI, B. COCO, P. CICCOROSSI, D. FLICHMAN, M. CAMPA, F. BONINO, AND M. BRUNETTO, *Sustained response to interferon-ribavirin combination ther-*

apy predicted by a model of hepatitis c virus dynamics using both HCV rna and alanine aminothransferase, Antiviral Therapy, 8 (2003), pp. 519–530.

[36] H. DAHARI, R. M. RIBEIRO, C. M. RICE, AND A. S. PERELSON, *Mathematical modeling of subgenomic hepatitis c virus replication in huh-7 cells*, Journal of Virology, 81 (2007), pp. 750–760.

[37] J. A. DAVID, *Optimal Control, Estimation, and Shape Design: Analysis and Applications*, PhD thesis, North Carolina State University, Raleigh, North Carolina, 2007.

[38] K. DEEP, K. P. SINGH, M. KANSAL, AND C. MOHAN, *A real coded genetic algorithm for solving integer and mixed integer optimization problems*, Applied Mathematics and Computation, 212 (2009), pp. 505–518.

[39] A. DEMPSTER AND N. L. D. RUBIN, *Maximum likelihood from incomplete data via the em algorithm*, Journal of Royal Statisical Society, 39 (1977), pp. 1–38.

[40] T. DENCE, *Cubics, chaos and newton's method*, The Mathematical Gazette, 81 (1997), pp. 403–408.

[41] M. DEUTSCH AND S. HADZIYANNIS, *Old and emerging therapies in chronic hepatitis c: an update*, Journal of Viral Hepatitis, 15 (2008), pp. 2–11.

[42] N. M. DIXIT, J. E. LAYDEN-ALMER, T. J. LAYDEN, AND A. S. PERELSON, *Modelling how ribavirin improves interferon response rates in hepatitis c virus infection*, Nature, 432 (2004).

[43] F. ERMIS AND E. S. TASCI, *New treatment strategies for hepatitis c infection*, World Journal of Hepatology, 7 (2015), pp. 2100–2109.

[44] F. FABRIZI, V. DIXIT, P. MESSA, AND P. MARTIN, *Interferon monotherapy of chronic hepatitis c in dialysis patients: meta-analysis of clinical trials*, Journal of Viral Hepatitis, (2008), pp. 79–88.

[45] FOOD AND D. ADMINISTRATION, *Guidance for Industry Chronic Hepatitis C Virus Infection: Developing Direct-Acting Antiviral Drugs for Treatment*, Center for Drug Evaluation and Research, 2013.

[46] W. R. FOWKES, G. CARYOTAKIS, E. DOYLE, E. JONGEWAARD, C. PEARSON, R. PHILLIPS, J. SACKETT, E. WRIGHT, H. BOHLEN, G. HUFFMAN, S. LENCI, E. LIEN, E. MCCUNE, AND G. MIRAM, *1.2 MW klystron for asymmetric storage ring B factory*, Proc. PAC, (1995), pp. 1–3.

[47] D. O. FRANK, *Acute Inflammatory Response to Endotoxin Challenge: Model Development, Parameter Estimation, and Treatment Control*, phd thesis, North Carolina State University, Raleigh, North Carolina, 2010.

[48] D. E. GOLDBERG, *Genetic Algorithms in Search, Optimization, & Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.

[49] M. GOLDBERG AND H. CHO, *Introduction to Regression Analysis*, WIT Press, December 2003.

[50] W. H. GREENE, *Econometric Analysis*, Prentice Hall, Upper Saddle River, New Jersey 07458, 5th ed., 2003.

[51] A. GRIEWANK, *On automatic differentiation*, Mathematical Programming: Recent Developments and Applications, (1989), pp. 83–108.

[52] J. GUEDJ AND A. S. PERELSON, *Telaprevir-based therapy increases with drug effectiveness: Implications for treatment duration*, Hepatology, 53 (2011).

[53] S. J. HADZIYANNIS, H. S. JR., T. R. MORGAN, V. BALAN, M. DIAGO, P. MARCELLIN, G. RAMADORI, H. B. JR., D. BERNSTEIN, M. RIZZETTO, S. ZEUZEM, P. J. POCKROS, A. LIN, AND A. M. ACKRILL, *Peginterferon-α2a and ribavirin combination therapy in chronic hepatitis c: a randomized study of treatment duration and ribavirin dose*, Ann. Intern. Med., 140 (2004), pp. 346–355.

[54] D. R. HAMILTON, J. K. KNIPP, AND J. B. H. KUPER, *Klystrons and Microwave Triodes*, Dover Publications, INC., 180 Varick Street New York, N.Y. 10014, 1966.

[55] A. HILL, S. KHOO, J. FORTUNAK, B. SIMMONS, AND N. FORD, *Minimum costs for producing hepatitis c direct-acting antivirals for use in large-scale treatment access programs in developing countries*, Clinical Infectious Diseases Advance Access, (2014).

[56] T. K. ISHII, ed., *Handbook of Microwave Technology*, vol. 2, Academic Press, Inc., 525 B Street, Suite 1900, San Diego, California 92101-4495, 1995.

[57] J. J. E. DENNIS AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, 1983.

[58] A. B. JAZWINSKI AND A. MUIR, *Direct-acting antiviral medications for chronic hepatitis c virus infection*, Gastroenterology & Hepatology, 7 (2011).

[59] A. JENSEN, M. FAZIO, J. NEILSON, AND G. SCHEITRUM, *Developing sheet beam klystron simulation capability in AJDISK*, IEEE Transactions on Electron Devices, 61 (2014), pp. 1666–1671.

[60] R. KE, C. LOVERDO, H. QI, R. SUN, AND J. O. LLOYD-SMITH, *Rational design and adaptive management of combination therapies for hepatitis c virus infection*, PLOS Computational Biology, 11 (2015).

[61] C. KELLEY, *Iterative Methods for Optimization*, Society for Industrial and Applied Mathematics, Raleigh, North Carolina, 1999.

[62] A. I. KIM AND S. SAAB, *Treatment of hepatitis c*, The American Journal of Medicine, (2005), pp. 808–815.

[63] J. J. KISER AND C. FLEXNER, *Direct-acting antiviral agents for hepatitis c virus infection*, Annual Review of Pharmacology and Toxicology, 53 (2013), pp. 427–449.

[64] J. C. LAGARIAS, J. A. REEDS, M. H. WRIGHT, AND P. E. WRIGHT, *Convergence properties of the nelder–mead simplex method in low dimensions*, SIAM Journal of Optimization, 9 (1998), pp. 112–147.

[65] K. LEVENBERG, *A method for the solution of certain nonlinear problems in least squares*, Quarterly Journal of Applied Mathmatics, 2 (1944), pp. 164–168.

[66] B. M. LEWIS, *Optimal Control and Shape Design: Theory and Applications*, Ph.d. Dissertation, North Carolina State University, 2003.

[67] C. J. LINGWOOD, G. BURT, K. J. GUNN, R. G. CARTER, R. MARCHESIN, AND E. JENSEN, *Automatic optimization of a klystron interaction structure*, IEEE Transactions on Electron Devices, 60 (2013), pp. 2671–2676.

[68] T.-C. LIOU, T.-T. CHANG, K.-C. YOUNG, X.-Z. LIN, AND C.-Y. LIN, *Detection of HCV rna in saliva, urine, seminal fluid, and ascites*, Journal of Medical Virology, 37.3 (1992), pp. 197–202.

[69] D. W. MARQUARDT, *An algorithm for least squares estimation of nonlinear parameters*, SIAM Journal of Applied Mathematics, 11 (1963), pp. 431–441.

[70] J. R. R. A. MARTINS, I. M. KROO, AND J. J. ALONSO, *An automated method for sensitivity analysis using complex variables*, American Institure of Aeronautics and Astronautics, (2000), pp. 1–12.

[71] H. MIAO, X. XIA, A. S. PERELSON, AND H. WU, *On identifiability of nonlinear ode models and applications in viral dynamics*, SIAM Rev Soc Ind Appl Math, 53 (2011), pp. 3–39.

[72] M. MORRIS, *Factorial sampling plans for preliminary computational experiments*, Technometrics, (1991), pp. 161–174.

[73] R. D. NEIDINGER, *Introduction to automatic differentiation and matlab object-oriented programming*, SIAM Review, 52 (2010), pp. 545–563.

[74] J. A. NELDER AND R. MEAD, *A simplex method for function minimization*, The Computer Journal, (1965).

[75] A. NEUMANN, N. LAM, AND H. DAHARI, *Hepatitis c viral dynamics in vivo and the antiviral efficacy of interferon-alpha therapy*, Science, 282 (1998), pp. 103–7.

[76] M. A. NOWAK AND C. R. M. BANGHAM, *Population dynamics of immune response to persistent viruses*, American Association for the Advancement of Science, 272 (1996), pp. 74–79.

[77] R. K. PARKER, R. H. ABRAMS, B. G. DANLY, AND B. LEVUSH, *Vacuum electronics*, IEEE Transactions on Microwave Theory and Techniques, 50 (2002), pp. 835–845.

[78] J.-M. PAWLOTSKY, *New hepatitis c therapies: The toolbox, strategies, and challenges*, Reviews in Basic and Clinical Gastroenterology and Hepatology, 146 (2014), pp. 1176–1192.

[79] T. POYNARD, P. MARCELLIN, S. S. LEE, C. NIEDERAU, G. S. MINUK, G. IDEO, V. BANE, J. HEATHCOTE, S. ZEUZEM, C. TREPO, AND J. ALBRECHT, *Randomised trial of interferon alpha2b plus ribavirin for 48 weeks or for 24 weeks versus interferon alpha2b plus placebo for 48 weeks for treatment of chronic infection with hepatitis c virus*, Lancet, 352 (1998), pp. 1426–1432.

[80] W. H. PRESS, B. P. FLANNERY, S. A. TEUKOLSKY, AND W. T. VETTERLING, *Numerical recipes in C: the art of scientific computing*, Cambridge University Press, New York, NY, USA, 1988.

[81] T. QUAISER AND M. MONNIGMANN, *Systematic identifiability testing for unambiguous mechanistic modeling - application to jak-stat, map kinase, and nf-kb signaling pathway models*, BMC Systems Biology, 3 (2009).

[82] R. QUILES-PEREZ, P. M. DE RUEDA, A. M.-L. MALDONADO, A. MARTIN-ALVAREZ, J. QUER, AND J. SALMERON, *Effects of ribavirin monotherapy on the viral population in patients with chronic hepatitis c genotype 1: Direct sequencing and pyrosequencing of the hcv regions*, Journal of Medical Virology, 86 (2014), pp. 1886–1897.

[83] K. R. REDDY, S. ZEUZEM, F. ZOULIM, O. WEILAND, A. HORBAN, C. STANCIU, F. G. VILLAMIL, P. ANDREONE, J. GEORGE, E. DAMMERS, M. FU, D. KURLAND, O. LENZ, S. OUWERKERK-MAHADEVAN, T. VERBINNEN, J. SCOTT, AND W. JESSNER, *Simeprevir versus telaprevir with peginterferon and ribavirin in previous null or partial responders with chronic hepatitis c virus genotype 1 infection (attain): a randomised, double-blind, non-inferiority phase 3 trial*, The Lancet Infectious Diseases, (2015), pp. 27–35.

[84] B. REHERMANN, *Hepatitis c virus versus innate and adaptive immune responses: a tale of coevolution and coexistence*, The Jounal of Clinical Investigation, 119 (2009), pp. 1745–1754.

[85] L. RONG, H. DAHARI, R. M. RIBEIRO, AND A. S. PERELSON, *Rapid emergence of protease inhibitor resistance in hepatitis c virus*, Science Translational Medicine, (2010).

[86] L. RONG, R. M. RIBEIRO, AND A. S. PERELSON, *Modeling quasispecies and drug resistance in hepatitis c patients treated with a protease inhibitor*, Bulletin of Mathematical Biology, 74 (2012).

[87] T. SATOGATA AND K. BROWN, eds., *Code Tesla For Modeling and Design of High-Power High-Efficiency Klystrons*, 2011.

[88] G. A. F. SEBER AND C. J. WILD, *Nonlinear Regression*, vol. 585 of Wiley Series in Probability and Statistics, Wiley, Hoboken, NJ, 2003.

[89] C. SHEPARD, L. FINELLI, AND M. ALTER, *Global epidemiology of hepatitis c virus infection*, Lancet Infection Disease, 5 (2005), pp. 558–567.

[90] S. SHERLOCK AND J. DOOLEY, *Diseases of the Liver and biliary System*, Blackwell Science, Oxford, UK, 10 ed., 1998.

[91] M. L. SHIFFMAN, C. M. HOFMANN, R. K. STERLING, V. A. LUKETIC, M. J. CONTOS, AND A. J. SANYAL, *A randomized, controlled trial to determine whether continued ribavirin monotherapy in*

hepatitis c virus-infected patients who responded to interferon-ribavirin combination therapy will enhance sustained virologic response, The Journal of Infectious Diseases, 184 (2001), pp. 405–9.

[92] Y.-M. SHIN, J.-X. WANG, L. R. BARNETT, AND N. C. LUHMANN, *Particle-in-cell simulation analysis of a multicavity w-band sheet beam klystron*, IEEE Transactions on Electron Devices, 58 (2011), pp. 251–258.

[93] M. SKOLNIK, *Role of radar in microwaves*, IEEE Transactions on Microwave Theory and Techniques, 50 (2002), pp. 625–632.

[94] R. C. SMITH, *Uncertainty Quantification: Theory, Implementation, and Applications*, SIAM, 2014.

[95] E. SNOECK, P. CHANU, M. LAVIELLE, P. JACQMIN, E. JONSSON, K. JORGA, T. GOGGIN, AND J. GRIPPO, *A comprehensive hepatitis c viral kinetic model explaining cure*, Clinical Pharmacology & Therapeutics, 87 (2010), pp. 706–713.

[96] D. C. SORENSEN, *Newton's method with a model trust region modification*, SIAM Journal on Numerical Analysis, 19 (1982), pp. 419–426.

[97] D. B. STRADER, T. WRIGHT, D. L. THOMAS, AND L. B. SEEFF, *Diagnosis, management, and treatment of hepatitis c*, Hepatology, 39 (2004), pp. 1147–1171.

[98] K. TAKAYAMA, N. FURUSYO, E. OGAWA, M. SHIMIZU, S. HIRAMINE, F. MITSUMOTO, K. URA, K. TOYODA, M. MURATA, AND J. HAYASHI, *A case of successful treatment with telaprevir-based triple therapy for hepatitis c infection after treatment failure with vaniprevir-based triple therapy*, Journal of Infection and Chemotherapy, (2014), pp. 517–581.

[99] H. TRAN, G. LANKFORD, M. E. READ, R. L. IVES, K. REPPERT, K. CLINE, AND J. GUZMAN, *Optimization of klystron designs using deterministic sampling methods*, IEEE Transactions on Electron Devices, 62 (2015), pp. 1032–1036.

[100] W. F. TRENCH, *Advanced Calculus*, Harper & Row Publishers, 1978.

[101] R. H. VARIAN AND S. F. VARIAN, *A high frequency oscillator and amplifier*, Journal of Applied Physics, 10 (1939).

[102] F. M. VERONESE AND A. MERO, *The impact of pegylation on biological therapies*, BioDrugs, 22 (2008), pp. 315–329.

[103] U. VISHKIN, *Deterministic sampling-a new technique for fast pattern matching*, SIAM Journal on Computing, 20 (1991), pp. 22–40.

[104] A. N. VLASOV, J. THOMAS M. ANTONSEN, D. P. CHERNIN, B. LEVUSH, AND E. L. WRIGHT, *Simulation of microwave devices with external cavities using magy*, IEEE Transactions on Plasma Science, 30 (2002).

[105] J. WELCH, *The maximum efficiency of a conventional klystron output cavity*, SLAC-PUB-3976, (1986).

[106] M. T. WENTWORTH, R. C. SMITH, AND H. BANKS, *Parameter selection and verification techniques based on global sensitivity analysis illustrated for an hiv model*, SIAM/ASA Journal on Uncertainty Quantification, 4 (2016), pp. 266–297.

[107] K. J. WILBY, N. PARTOVI, J.-A. E. FORD, E. D. GREANYA, AND E. M. YOSHIDA, *Review of boceprevir and telaprevir for the treatment of chronic hepatitis c*, Canadian Journal of Gastroenterology, 26 (2012), pp. 205–210.

[108] D. WODARZ, *Hepatitis c virus dynamics and pathology: the role of ctl and antibody responses*, Journal of General Virology, 84 (2003), pp. 1743–1750.

[109] D. M. YOU AND P. J. POCKROS, *Simeprevir for the treatment of chronic hepatitis c*, Expert Opinion on Pharmacotherapy, 14 (2013), pp. 2581–2589.

[110] S. ZENKER, J. RUBIN, AND G. CLERMONT, *From inverse problems in mathematical physiology to quantitative differential diagnoses*, PLOS Computatioal Biology, 3 (2007), pp. 2072–2086.

[111] J. S. ZUR WIESCH, D. PIEPER, I. STAHMER, T. EIERMANN, P. BUGGISCH, A. LOHSE, J. HAUBER, AND J. VAN LUNZEN, *Sustained virological response after early antiviral treatment of acute hepatitis c virus and HIV coinfection*, Clinical Infectious Diseases, 49 (2009), pp. 466–472.