APPROVAL PAGE FOR GRADUATE THESIS OR PROJECT

GS-13

SUBMITTED IN PARTIAL FULFILLMENT OF REQUIREMENTS FOR DEGREE OF MASTER OF SCIENCE AT CALIFORNIA STATE UNIVERSITY, LOS ANGELES BY

Leslie J. Szijjarto
Candidate

Mathematics
Department

TITLE: **AN AGE-STRUCTURED SIR MODEL FOR CHOLERA EPIDEMICS**

APPROVED:

Dr. Melisa Hendrata Committee Chairperson	Signature
Dr. Randolph Cooper Faculty Member	Signature
Dr. Liming Wang Faculty Member	Signature
Dr. Grant Fraser Department Chairperson	Signature

DATE: **December 5, 2012**

AN AGE-STRUCTURED SIR MODEL FOR CHOLERA EPIDEMICS

A Thesis

Presented to

The Faculty of the Department of Mathematics

California State University, Los Angeles

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

By

Leslie J. Szijjarto

December 2012

ACKNOWLEDGMENTS

This thesis would have been impossible without the support and mentoring of my advisor, Dr. Melisa Hendrata. I appreciate all her time, patience, and understanding to make my M.S. experience productive and stimulating. The joy and enthusiasm she has for mathematics was contagious and motivational for me, and many of my colleagues too. It has been an honor to be a student of hers.

I would like to thank the other thesis committee members: Dr. Randolph Cooper, Dr. Liming Wang, and Dr. Grant Fraser for their time, interest, and helpful comments. The same gratitude goes to Dr. Daphne Liu for inspiring me to take the thesis route.

I would also like to thank the CSULA Math department for the wonderful learning experience. I have been to several great schools, but I have never met such a closely knit group of students, professors, and administrators. It made me feel so at home. I will definitely come back for more learning soon.

Finally, I would like to thank my loving wife, Lisa, for giving me unending patience and support. I dare not even imagine how I would have done this without her. Thank you for believing in me.

ABSTRACT

An Age-Structured SIR Model

For Cholera Epidemics

By

Leslie J. Szijjarto

Mathematical modeling is an important tool in epidemiology. It provides a way to understand the primary forces in disease dynamics, and to conduct theoretical experiments that are not possible in practice. In this paper, we study mathematical models that describe the spreading of epidemic cholera through a human population. We are particularly interested in one which shows the behavior of the disease across different human age groups as well as across time. A major assumption of many mathematical models of epidemics is that the population can be divided into a set of distinct compartments. These compartments are defined with respect to the disease status of individuals in the population. The general SIR model, which is used to simulate behavior of many diseases, consists of three compartments: susceptible (S), infected (I), and recovered (R). Since the general SIR model was first developed in 1927, several revisions have been made by different researchers to tailor it to fit to cholera epidemics. Until recently, most models described behavior with respect to time only. In 2011, Gobbert et al. [9] extended the SIR Cholera model to simulate behavior across both time and age groups. This attempt was based on the fact that in endemic areas, both the risk of contracting the disease, and the response to currently available vaccines are age dependent.

In this thesis we study the SIR-based age-structured epidemic cholera model

as proposed in [9]. The model consists of a coupled system of five differential equations: three partial differential equations, describing the disease dynamics of a human population across both time and age, and two ordinary differential equations, describing cholera bacteria concentrations in the water supply across time. We solve the age-structured model numerically by implementing a first order accurate finite difference method and the Euler method. An agreement with the solution claimed in [9] is obtained. We further suggest and implement the Lax-Wendroff scheme and the Midpoint method to improve the accuracy of the model solution. We discuss our simulation results and suggest a way to improve one of the model parameters. Sensitivity analysis on model parameters are performed to determine their relative importance to disease dynamics, and finally we compare the model to several case studies found in literature.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

CHAPTER 1

Introduction

Cholera is an acute infection of the small intestine that is caused by ingestion of food or water contaminated with the bacterium *Vibrio cholerae*. It is an ancient, water-borne disease that continues to cause epidemics and pandemics despite ongoing efforts to limit its spread. Every year, there are an estimated 3 to 5 million cholera cases, resulting in 100,000 to 120,000 deaths, and the number of cases reported to the WHO (World Health Organization) continues to rise. In 2011 alone, cases were reported in 58 different countries [20], with most cases reported in Asia and Africa.

Epidemic cholera is characteristically explosive in nature. About 80% of symptomatic cholera cases are mild or moderate, but 20% cause severe dehydration, leading to death if untreated. Populations lacking prior immunity can be devastated by the disease in a matter of weeks. Cholera affects all age groups, but children are especially vulnerable, and can die within hours.

The dynamics of cholera epidemics are still not fully understood. A very complex system of interactions occurs between the human host, pathogen, and environment, resulting in some outbreaks being mild and others, explosive in nature. Recently, one of the main mechanisms thought to be responsible for epidemic behavior is the consumption of vibrios freshly shed (from human stool) into public drinking water. Merrell and Butler (2002) [14] reported a study showing that cholera is up

to 700 times more infectious in the first 5 hours following excretion, but then subsequently decays quickly to the normal degree of infectivity. The larger the dose ingested by an individual, the more severe the symptoms, and the larger the amount of vibrios subsequently excreted. So, as long as the main source of the disease is the bacteria naturally occurring in the water source, any outbreak in the population is mild and slow to occur. Once people start routinely coming into contact with water or food that is contaminated with vibrios freshly shed from human feces, the disease is able to spread epidemically.

Mathematical modeling has been used as a tool in epidemiology to understand the spreading mechanism of infectious diseases. It provides a way to gain insight on the primary forces in disease dynamics, and to conduct theoretical experiments that are not possible in practice. In this paper, we study the development of mathematical models that describe the spreading of epidemic cholera through a human population. We are particularly interested in one which shows the behavior of the disease across different human age groups as well as across time.

This thesis is organized as follows. In Chapter 2, we present the general SIR (Susceptible-Infected-Recovered) model for epidemic disease, and describe several developments that lead to the age-structured model. The model consists of a system of partial and ordinary differential equations. Chapter 3 discusses our approach to solve the age-structured model numerically by implementing a first order accurate finite difference method to solve the partial differential equations, and the Euler method to solve the ordinary differential equations. In Chapter 4, we propose several ways to improve on the accuracy of the model, replacing the first order method with a second

order Lax-Wendroff scheme, and the Euler method with the Midpoint method. The comparison of our simulation results for both methods is also presented. In Chapter 5, we perform a sensitivity analysis on model parameters in order to determine their relative importance to disease dynamics. We also suggest a way to improve one of the model parameters. We conclude in the final chapter by comparing model output to several case studies found in literature.

CHAPTER 2

Mathematical Model

Infectious disease modeling has a long history, going back to at least Daniel Bernouli's smallpox model from 1760. The discipline is driven by the desire to understand the dynamics of an outbreak or epidemic in order to plan control strategies.

When dealing with large populations, as in the case of cholera, deterministic (or compartmental) models are commonly used, where individuals in the population are compartmentalized according to disease stage. The most common model is the SIR model which divides the individuals in the population into three compartments: Susceptible (S), Infected (I), and Recovered (R).

In this chapter we discuss the basic disease models, and several developments that have been made to model cholera epidemics. These models provide the building blocks for an age-structured cholera model, which is the main topic of this thesis.

2.1 Basic Models

In this section, we discuss two basic models for general epidemic disease: the SIR and SIS models. In both models, $S(t)$ represents the number of individuals who are susceptible to the disease, but not yet infected at time t, while $I(t)$ denotes the number of individuals who have been infected with the disease at time t and are capable of spreading the disease to those in the susceptible category. Each member of the population typically progresses from the susceptible group into the infected group. In

some diseases, an individual gains an immunity that prevents the person from getting reinfected, while in others, there is a possibility that the recovered individual can get reinfected, and this is the main difference between the SIR and SIS models. In the SIR model, the infected individuals will proceed to the recovered group after they recover, while in the SIS model, the infected individuals will return to be susceptibles after the disease recovery. Figures 2.1 and 2.2 show the flow dynamic of the individuals in different compartments of the two models.

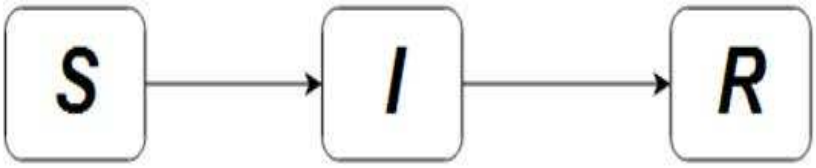

Figure 2.1: SIR Flow Diagram

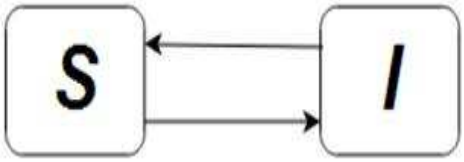

Figure 2.2: SIS Flow Diagram

2.1.1 SIR Model

The SIR model is the most common type used to model the spread of disease in a large population. Developed in 1927 by W. O. Kermack and A. G. McKendrick [11], it was the first to successfully predict the behavior of many recorded epidemics (Brauer & Castillo-Chavez [3]).

In the SIR model, $R(t)$ represents the number of individuals who have recovered from the disease at time t. Those in this category are immune and un-

able to transmit the infection to others. Each population group is differentiable with respect to time, and assuming that the population size is constant, that is, $N = S(t) + I(t) + R(t)$, the rate of change of each population group is given by the following system of ODEs:

$$\frac{dS}{dt} = -\lambda SI \tag{2.1}$$

$$\frac{dI}{dt} = \lambda SI - \gamma I \tag{2.2}$$

$$\frac{dR}{dt} = \gamma I, \tag{2.3}$$

where λ = rate of infection and γ = rate of recovery.

This SIR model assumes that each infected individual has an equal probability of transmitting the disease to other λS individuals, and each infected has equal recovery rate γ. It is also assumed that the rate of infection and recovery is much faster than the time scale of births and deaths, therefore, these factors are ignored in this model. From the system (2.1)-(2.3), one can guess that the number of susceptibles will decrease, while the number of recovered individuals will increase over time. The solution to this SIR model with parameter values $\lambda = 0.00003$, $\gamma = 0.03$, and initial values S(0)=10000, I(0)=1, and R(0)=0 over a period of 24 weeks is shown in Figure 2.3.

2.1.2 SIS Model

In the basic SIS model, individuals recover with no immunity to the disease, that is, they are immediately susceptible once they have recovered. Removing the equation

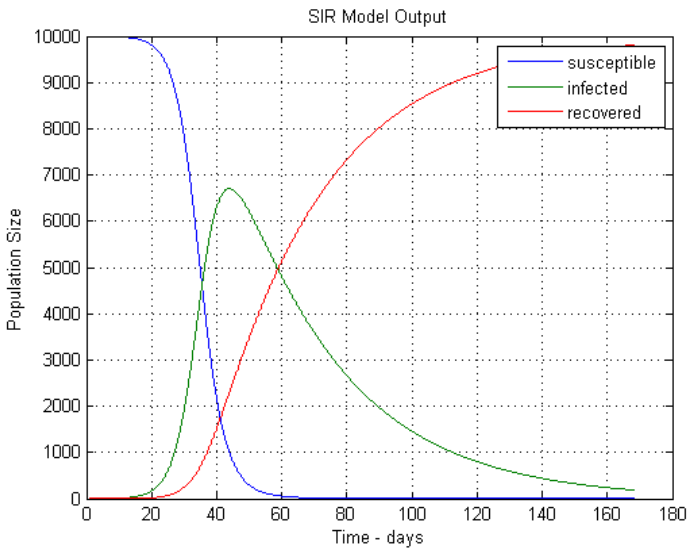

Figure 2.3: SIR Model Output

representing the recovered population from the SIR model and adding those removed from the infected population into the susceptible population gives the following two ordinary differential equations:

$$\frac{dS}{dt} = -\lambda SI + \gamma I \tag{2.4}$$

$$\frac{dI}{dt} = \lambda SI - \gamma I \tag{2.5}$$

2.2 Modeling Disease Transmission

One critical component of all infectious disease models is the mode of transmission, the most common of which are environment-to-human and human-to-human transmissions.

2.2.1 Environment-to-human Transmission

Prior to 2001, cholera based SIR models assumed that cholera bacteria are transmitted only via the interaction between the infected with susceptible individuals. In 2001, Cláudia Codeço [7] proposed the role of indirect transmission via the environmental reservoir. Her model explicitly accounts for the concentration of *V. Cholerae* bacteria in the drinking water supply. It is an extension of the demographic SIS model (i.e. includes birth and natural death rates), and is composed of the following three ODEs:

$$\begin{aligned}
\frac{dS}{dt} &= b(N - S) - \beta\lambda S \\
\frac{dI}{dt} &= \beta\lambda S - \gamma I \\
\frac{dB}{dt} &= B(\psi - \delta) + \xi I,
\end{aligned}$$

where B = concentration of *V.cholerae* in water, b = human birth and death rates, N = total human population, β = ingestion rate, γ = recovery rate, ψ = *V. cholerae* growth rate, δ= *V. cholerae* loss rate, ξ = vibrio shedding rate, which is the contribution from an infected individual to the vibrio reservoir through excretion. The infection rate $\lambda = B/(\kappa + B)$, where κ is the saturation constant.

Codeço's model was the most accurate of its time, but it was not able to accurately explain the behavior of epidemics that were explosive in nature.

2.2.2 Human-to-human Transmission

In 2002, Andrew Camilli et al. [6] discovered that freshly shed *V. cholerae* from human intestines outcompeted other *V. cholerae* by as much as 700-fold for at least

the first 5 hours in the environment. In 2006, David Hartley et al. [10] improved Codeço's model which only accounted for environment-to-human infection (through contaminated food or water) by distinguishing highly infectious (HI) and non-highly infectious (non-HI) stages of the *V. cholerae* pathogen. Hartley's model provides a basis for the human-to-human transmission pathway (i.e., fecal to oral).

Hartley's model is an extension of the demographic SIR model, which includes the natural death rate of individuals in each of the three categories. It distinguishes the infection rate from HI and non-HI vibrio groups, and the dynamics between these two vibrio groups. The model consists of five ODEs given by

$$\frac{dS}{dt} = b(N - S) - (\beta_L \lambda_L + \beta_H \lambda_H)S \tag{2.6}$$

$$\frac{dI}{dt} = (\beta_L \lambda_L + \beta_H \lambda_H)S - (\gamma + b)I \tag{2.7}$$

$$\frac{dR}{dt} = \gamma I - bR \tag{2.8}$$

$$\frac{dB_H}{dt} = \xi I - \chi B_H \tag{2.9}$$

$$\frac{dB_L}{dt} = \chi B_H - \delta_L B_L, \tag{2.10}$$

where the subscripts H and L represent the HI and non-HI vibrios, respectively. The list of parameter descriptions is given in Table 2.1.

Hartley's cholera model was the first that accurately described the observed explosive epidemic patterns of past cholera outbreaks.

2.3 Age-Structured Model

In this section we discuss two improvements to the Hartley model proposed by Gobbert et al.(2010) [9].

The first improvement is done by introducing several new parameters relevant specifically to cholera epidemics. One such parameter is *waning immunity*. This is based on the fact that individuals gain a certain immunity once they recover from cholera, but this immunity diminishes over a certain period of time. Likewise, newborn babies inherit a certain level of immunity at birth, but it also vanishes as they grow older.

The equation (2.6) in the Hartley's model can then be improved as follows

$$\frac{dS}{dt} = \Lambda + \omega R - (\beta_L \lambda_L + \beta_H \lambda_H)S - bS, \tag{2.11}$$

where Λ is the recruitment rate, that is the number of susceptibles entering the population, and ω is the rate of waning immunity of those who have been recovered from the disease and become susceptible again.

Gobbert's model also includes the *oral rehydration therapy* factor which can effectively reduce mortality due to the disease. Furthermore, it distinguishes the recovery rate of treated versus untreated cholera, which makes the model more applicable for epidemic cholera in remote areas in which not all infected individuals can receive medical treatment. With these additional terms, equations (2.7)-(2.8) can be

modified as follows:

$$\frac{dI}{dt} = (\beta_L \lambda_L + \beta_H \lambda_H)S - bI - (1-h)\Delta I - (\gamma_1(1-u) + \gamma_2 u)I \tag{2.12}$$

$$\frac{dR}{dt} = (\gamma_1(1-u) + \gamma_2 u)I - bR - \omega R \tag{2.13}$$

The second improvement is based on the argument that the risk for contracting cholera and/or dying from it depends on the age of the humans, and thus an age-structured model can offer additional insights and the possibility to study the effects of treatment options. We now discuss how to add age structure into the model equations (2.11)-(2.13).

We follow the well-known McKendrick approach to convert an ODE into an age-structured PDE [4]. To illustrate the method, we take the equation (2.1) as an example:

$$\frac{dS}{dt} = -\lambda SI.$$

Let $S(a,t)$ and $I(a,t)$ be the number of susceptibles and infectives of age a at time t. Then at time $h = \Delta t$ later, the individuals who are still susceptibles will have aged by an amount of $\Delta a = \Delta t = h$. That is,

$$S(a+h, t+h) = S(a,t) - \lambda S(a,t)I(a,t)h.$$

We expand $S(a+h, t+h)$ in Taylor series around (a, t) to get

$$\begin{aligned} S(a+h,t+h) =& S(a,t) + h\frac{\partial S(a,t)}{\partial a} + h\frac{\partial S(a,t)}{\partial t} \\ &+ \frac{h^2}{2!}\frac{\partial^2 S(a,t)}{\partial a^2} + \frac{h^2}{2!}\frac{\partial^2 S(a,t)}{\partial t^2} + h^2\frac{\partial^2 S(a,t)}{\partial t \partial a} + O(h^3) \end{aligned}$$

If h is sufficiently small, then h^2 and higher order terms are negligible, giving us

$$\begin{aligned} h\Big(\frac{\partial S}{\partial a} + \frac{\partial S}{\partial t}\Big) &= S(a+h,t+h) - S(a,t) \\ &= -\lambda S(a,t) I(a,t) h \end{aligned}$$

Hence,

$$\frac{\partial S}{\partial a} + \frac{\partial S}{\partial t} = -\lambda S I, \tag{2.14}$$

for $t \geq 0, 0 \leq a \leq A$, where A is the upper bound on the human's age in the population, and initial condition $S(a, 0) = S_0(a)$.

The age-structured PDEs for the equations (2.11)-(2.13) are derived in a similar manner and the system now becomes

$$\frac{\partial S}{\partial t} + \alpha\frac{\partial S}{\partial a} = \Lambda + \omega R - (\beta_L\lambda_L + \beta_H\lambda_H)S - bS \tag{2.15}$$

$$\frac{\partial I}{\partial t} + \alpha\frac{\partial I}{\partial a} = (\beta_L\lambda_L + \beta_H\lambda_H)S - bI - (1-h)\Delta I - (\gamma_1(1-u) + \gamma_2 u)I \tag{2.16}$$

$$\frac{\partial R}{\partial t} + \alpha\frac{\partial R}{\partial a} = (\gamma_1(1-u) + \gamma_2 u)I - bR - \omega R \tag{2.17}$$

$$\frac{dB_H}{dt} = \int_0^A \xi\eta I \, da - \chi B_H \tag{2.18}$$

$$\frac{dB_L}{dt} = \chi B_H - \delta_L B_L, \tag{2.19}$$

where $\lambda_H = \frac{B_H(t)}{\kappa_H(a)+B_H(t)}$, $\lambda_L = \frac{B_L(t)}{\kappa_L(a)+B_L(t)}$, and $\alpha =$ proportionality factor. The model parameters and their units are listed in Table 2.1.

Note that B_H and B_L are independent of human age, and thus the equations that give their rate of change stay as ODEs. Equation (2.18), however, is dependent on $I(a,t)$ and to get dB_H/dt we need to integrate $I(a,t)$ with respect to a to get the number of infected individuals of all ages at a particular time t.

The initial conditions for the model (2.15)-(2.19) are given by

$$S(a,0) = S_0(a),\ I(a,0) = I_0(a),\ R(a,0) = R_0(a)$$

$$B_L(0) = B_{L0},\ B_H(0) = B_{H0},$$

where $S_0(a), I_0(a)$ and $R_0(a)$ are some functions of a, and B_{L0}, B_{H0} are constants.

The boundary conditions are defined based on the following assumptions:

(1) Babies of age one year old or younger have a natural immunity.

(2) People of ages between 15 and 45 have on average three children throughout their lifetime.

Assumption 2 can be modeled by first defining the *fecundity* as a function of age a. The fecundity function describes the potential reproductive capacity of a population. There are many ways to model the pattern of births within a population. A normal distribution curve is one example. Another example, suggested in Gobbert

(2010) is to use a sine function given by:

$$f(a) = \begin{cases} \frac{1}{5}\sin^2\left[\left(\frac{a-15}{30}\right)\pi\right], & \text{if } 15 < a < 45 \text{ years,} \\ 0, & \text{otherwise,} \end{cases}$$

whose graph is given in Figure 2.4.

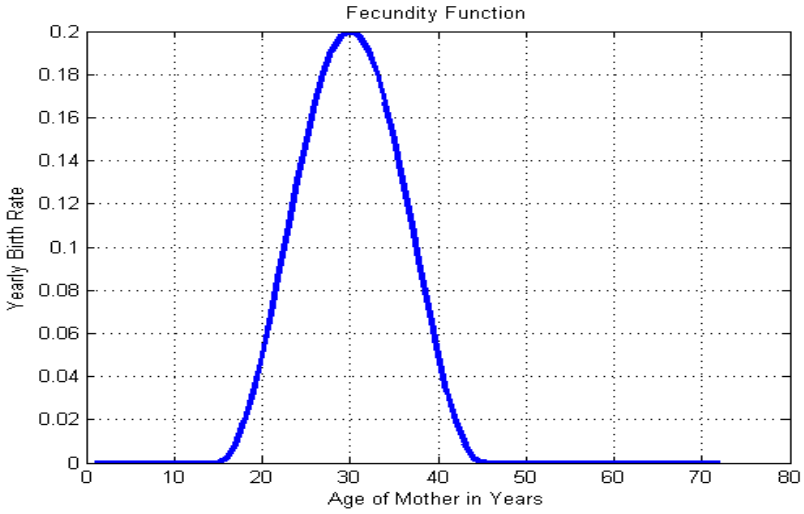

Figure 2.4: Fecundity Function

We then define the boundary condition for $R(0,t)$ by

$$R(0,t) = \int_0^A (S(a,t) + I(a,t) + R(a,t))f(a)da, \tag{2.20}$$

which shows that all newborns are placed into the recovered population, satisfying assumption 1.

Table 2.1: Model Parameters and Their Units

Parameter	**Description**	**Units**
α	proportionality factor (wave speed)	$\frac{week}{days}$
$\Lambda(a,t)$	recruitment rate, number of susceptible humans entering pop of age a at time t	$\frac{humans}{week*days}$
$h(a,t)$	oral rehydration therapy, reduces disease related mortality (90% effective)	none
$u(a,t)$	antibiotic treatment rate for humans	none
$\beta_L(a)$	ingestion rate of non-HI vibrios	$\frac{1}{day}$
$\beta_H(a)$	ingestion rate of HI vibrios	$\frac{1}{day}$
$\kappa_L(a)$	saturation const of non-HI vibrios	$\frac{cells}{ml}$
$\kappa_H(a)$	saturation const of HI vibrios	$\frac{cells}{ml}$
$b(a)$	natural mortality rate of humans	$\frac{1}{day}$
$\omega(a)$	rate of waning immunity of humans	$\frac{1}{day}$
$\Delta(a)$	disease related mortality rate for humans	$\frac{1}{day}$
$f(a)$	maternity rate	$\frac{1}{week}$
γ_1	recovery rate of untreated cholera	$\frac{1}{day}$
γ_2	recovery rate of treated cholera	$\frac{1}{day}$
ξ	shedding rate of vibrios from infected person	$\frac{cells}{ml*day*human}$
η	relative amount of stool per unit time	none
χ	rate of vibrio moving from HI to non-HI state	$\frac{1}{day}$
δ_L	death rate of vibrio in the environment	$\frac{1}{day}$
A	Upper bound on the age of people	weeks

CHAPTER 3

Numerical Solution to the Age-Structured Model

A mathematical model that represents a real life situation is typically complex, often making it very difficult or even impossible to solve analytically. In such case, implementing numerical methods is the approach taken to approximate the solution.

In Section 3.1 we describe the finite difference concept to approximate the derivative of a function, and its implementation to the age-structured model presented in Section 2.3. The parameter values used in the simulation are taken from [9] and are listed in Section 3.2. We solve the age-structured model numerically using a first order scheme and present our computational results in Section 3.3.

3.1 Finite Difference Schemes

In general, a finite difference approximation to the value of some derivative of a scalar function $u(x)$ at a point x_0 in its domain, say $u'(x_0)$, relies on a suitable combination of sampled function values at nearby points, such as $x_0 \pm h, x_0 \pm 2h, \ldots$, for some h. The *step size* h is assumed to be sufficiently small.

We begin with the first order derivative. Recall that the Taylor series expansion for $u(x_0 + h)$ around the point x_0:

$$u(x_0 + h) = u(x_0) + hu'(x_0) + \frac{h^2}{2!}u''(x_0) + \frac{h^3}{3!}u'''(x) + ... \tag{3.1}$$

Solving for u', we get

$$u'(x_0) = \frac{u(x_0 + h) - u(x_0)}{h} + O(h), \tag{3.2}$$

where $O(h)$ refers to the error term that is proportional to h and whose absolute value is bounded by a constant multiple of $|h|$ as $h \to 0$. Equation (3.2) is known as the *forward difference approximation* for $u'(x)$ if $h > 0$ and *backward difference approximation* if $h < 0$. Since the error is proportional to h, this forms a first order approximation to $u'(x_0)$.

To approximate higher order derivatives, we need to evaluate the function at more than two points, which usually are equally spaced for simplicity. We consider the Taylor series expansion for $u(x_0 - h)$ around the point x_0:

$$u(x_0 - h) = u(x_0) - hu'(x_0) + \frac{h^2}{2!}u''(x_0) - \frac{h^3}{3!}u'''(x) + ... \tag{3.3}$$

Adding (3.1) and (3.3) gives

$$u(x_0 + h) + u(x_0 - h) = 2u(x_0) + u''(x_0)h^2 + O(h^4).$$

Solving for $u''(x_0)$ we arrive at the second order *centered finite difference approximation*:

$$u''(x) = \frac{u(x_0 + h) - 2u(x_0) + u(x_0 - h)}{h^2} + O(h^2). \tag{3.4}$$

We will now employ the finite difference formulae to devise a numerical solution scheme for a hyperbolic PDE:

$$\frac{\partial u}{\partial t} + \alpha \frac{\partial u}{\partial a} = f(a, t). \tag{3.5}$$

We assume rectangular mesh (a_i, t_j) with uniform space (age) mesh size $h = \Delta a = a_{i+1} - a_i$ and time step size $k = \Delta t = t_{j+1} - t_j$. We use $u_{i,j} \approx u(a_i, t_j)$ to denote our numerical approximation to the solution $u(a,t)$ at the indicated node. The most elementary numerical solution scheme is obtained by replacing the time derivative by the first order forward difference approximation and the age derivative by the backward difference, giving us the explicit scheme

$$\frac{u_{i,j+1} - u_{i,j}}{k} + \alpha \frac{u_{i,j} - u_{i-1,j}}{h} = f(a_i, t_j). \tag{3.6}$$

which can be solved for $u_{i,j+1}$. The above scheme is often called *the upwind scheme*.

Implementing the upwind scheme to the age-structured model (2.15)-(2.17), we obtain

$$\frac{S_{i,j+1} - S_{i,j}}{k} + \alpha \frac{S_{i,j} - S_{i-1,j}}{h} = [\Lambda + \omega R - (\beta_L \lambda_L + \beta_H \lambda_H) S - bS]_{i,j} \tag{3.7}$$

$$\begin{aligned} \frac{I_{i,j+1} - I_{i,j}}{k} + \alpha \frac{I_{i,j} - I_{i-1,j}}{h} =& [(\beta_L \lambda_L + \beta_H \lambda_H) S \\ & - bI - (1-\theta)\Delta I - (\gamma_1(1-u) + \gamma_2 u) I]_{i,j} \end{aligned} \tag{3.8}$$

$$\frac{R_{i,j+1} - R_{i,j}}{k} + \alpha \frac{R_{i,j} - R_{i-1,j}}{h} = [(\gamma_1(1-u) + \gamma_2 u) I - bR - \omega R]_{i,j}. \tag{3.9}$$

Note that the functions on the right hand side are evaluated at the node (a_i, t_j).

Equation (2.19) involves a first order derivative with respect to a single variable t and its finite difference formula is given by

$$\frac{B_{L_{j+1}} - B_{L_j}}{h} = \left[\chi B_{H_j} - \delta_L B_{L_j}\right] \tag{3.10}$$

Equation (2.18) involves an integral which we also need to compute numeri-

cally. For this, we choose to implement the composite Trapezoidal rule:

$$\int_a^b f(x)dx \approx \frac{h}{2}\sum_{k=0}^{n-1}\left[f(x_{k+1}) + f(x_k)\right], \tag{3.11}$$

where h is the step size chosen and $n = (b-a)/h$.

Applying (3.11) to (2.18) we get

$$\frac{B_{H_{j+1}} - B_{H_j}}{h} = \left[\frac{h}{2}\xi\eta\sum_{i=0}^{n-1}(I_{i+1,j} + I_{i,j})\right] - \chi B_{H_j} \tag{3.12}$$

where $n = A/h$

3.2 Parameter Values

For all simulations, we set the following parameter values, most of which are taken from [9]. The value for parameter b, the mortality rate of human beings, was taken from [19], which is the average for multiple African countries. The parameter χ, which denotes the rate of vibrios moving from HI to non-HI state, was taken from [10] and it is set to be 1/5 per hour or 33.6/week. Table 3.1 lists all parameters and their values. In all of our simulations, we take the time step size $k = \Delta t = 1/50$ week, age mesh size $h = \Delta a = 1$ week, and the initial number of individuals $N = 10^4$, which is distributed uniformly across the age range 0 to 72 years old.

Table 3.1: Model Parameters and Their Values

Parameter	**Quantity**
α	$1\ \frac{week}{days}$
$\Lambda(a,t)$	$0\ \frac{humans}{week*days}$
h(a,t)	0.9
u(a,t)	0.0
$\beta_L(a)$	$1.5/7\frac{1}{day}$
$\beta_H(a)$	$1.5/7\frac{1}{day}$
$\kappa_L(a)$	$10^6\frac{cells}{ml}$
$\kappa_H(a)$	$\kappa_L/700\frac{cells}{ml}$
b(a)	$1/50\frac{1}{year}$
γ_1	$1/5\ \frac{1}{day}$
γ_2	$1/3\frac{1}{day}$
η	0.1
χ	$1/5\frac{1}{hour}$
δ_L	$1/30\ \frac{1}{day}$
$\omega(a)$	$\frac{1}{365}$ (a <=10 yrs), $\frac{1}{2*365}$ ($a > 10$ yrs) $\frac{1}{day}$
$\Delta(a)$	0.032 (a <=10 yrs), 0.007 ($a > 10$ yrs) $\frac{1}{day}$
A	72 years

3.3 Computational Results

We simulate the age-structured model (equations (3.7)-(3.9), (3.10) and (3.12)) in three different scenarios:

- Reference simulation (i.e. no infected population)
- Simulation with high rate of shedding of cholera bacteria
- Simulation with low rate of shedding of cholera bacteria

We run the simulation in two different time periods: 24 weeks and 208 weeks. With the 24-week simulation we can get a more detailed picture of the interplay between variables and parameters in the system during a course of the epidemic, while the 208-week simulation can give us the information on the long term behavior of the system. We also present the plot of the highly infectious (HI) and non-highly infectious (non-HI) vibrios to see their transition. Our numerical scheme is implemented

in the Matlab programming language, whose code is given in Appendix A.

3.3.1 Reference Simulation

For the reference simulation, no infected individuals are introduced into the population. This corresponds to setting $I(a,0) = 0$ at $t = 0$. In this case, we expect almost all individuals in the population to belong to the susceptible category, except the newborn babies who are immune and placed in the recovered population. The number of recovered population is therefore relatively very small compared to the susceptibles. According to the immunity waning function $\omega(a)$, newborns are immune for one year. Hence, we define the initial conditions as follows

$$S(a,0) = \begin{cases} 0, & \text{if } 0 <= a <= 52 \text{ weeks}, \\ \text{d}, & \text{if } a > 52 \text{ weeks}, \end{cases}$$

and

$$R(a,0) = \begin{cases} \text{d}, & \text{if } 0 <= a <= 52 \text{ weeks}, \\ 0, & \text{if } a > 52 \text{ weeks}, \end{cases}$$

where

d = (10000 individuals)/(72 years * 52 weeks/year) = 2.671 individuals/week

The plot of the reference simulation is given in Figure 3.1. Since there are no infected people in this reference simulation, the number of infected people remain zero at all times. Total population increases due to a positive difference between the birth rate and natural death rate (Figure 3.2). The susceptible population decreases slightly due to people who died of natural causes during the simulation, and the recovered population increases slightly due to the addition of newborns. The two

vibrio populations also remain at zero (Figure 3.3).

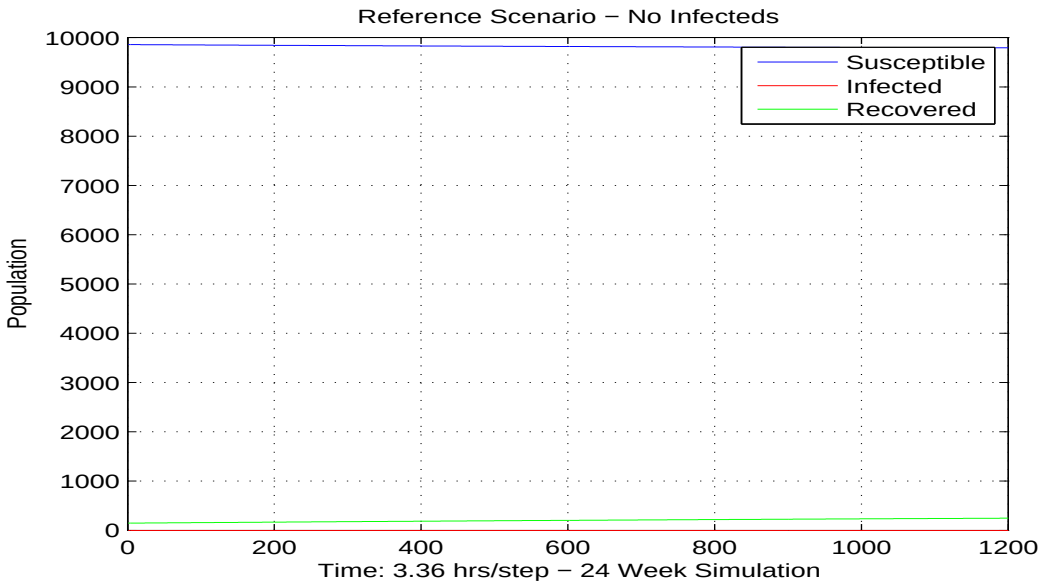

Figure 3.1: Reference Simulation - Human Populations

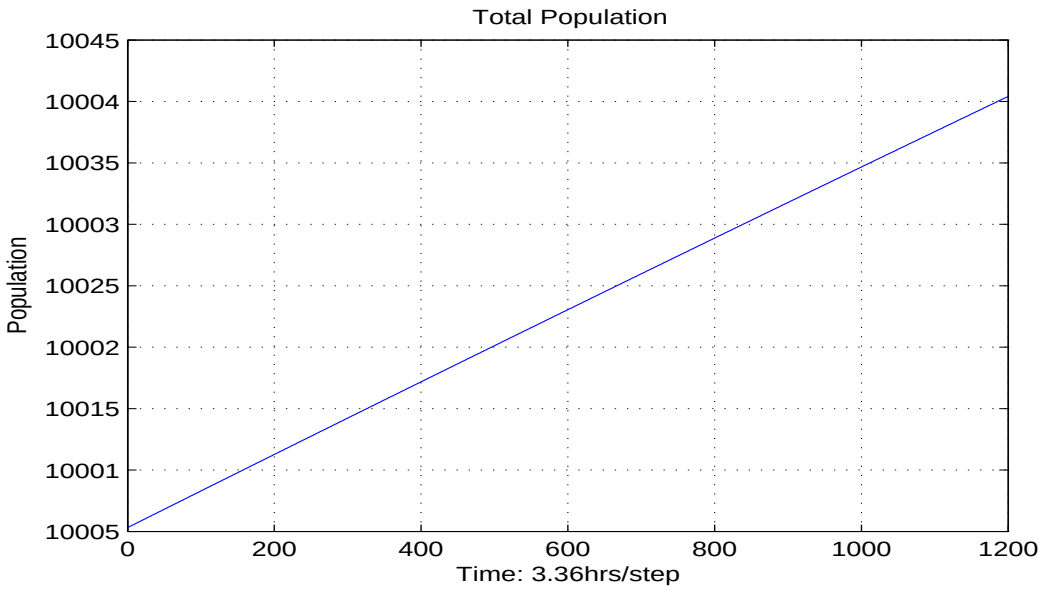

Figure 3.2: Reference Simulation - Total Human Population

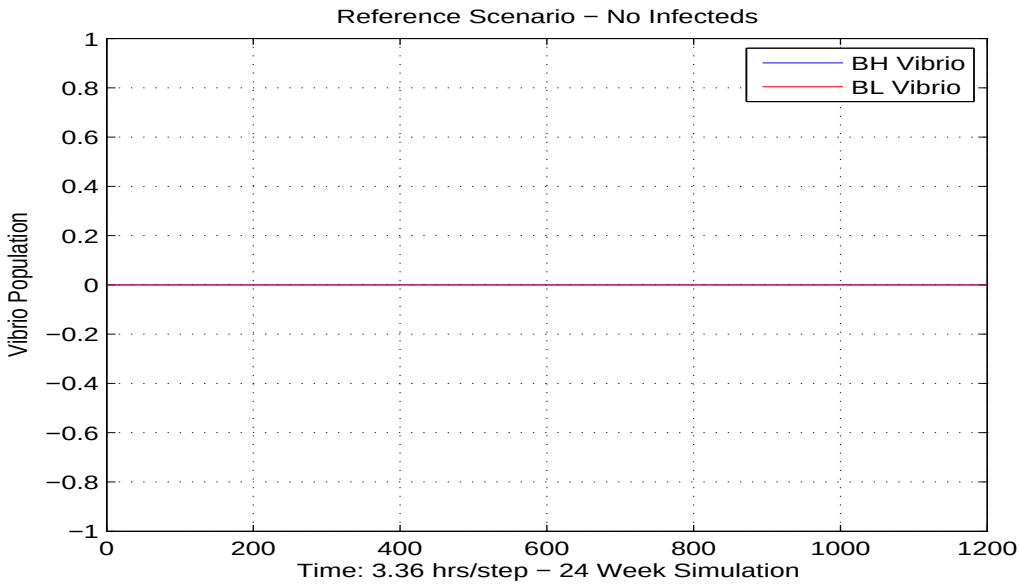

Figure 3.3: Reference Simulation - Vibrio Populations

3.3.2 High Rate Shedding Simulation

For the high rate shedding scenario, we include one extra infected person of age 18 in the initial population. The shedding rate of vibrio from infected individuals is set to $\xi = 10^9$ cells/(ml*day*human). All other parameters have the same values as the reference simulation. In the 24-week simulation, we note the explosive nature of the HI Rate shedding epidemic. The infected population peaks at 5000 people within the first two weeks and then drops down to an endemic state in 4 weeks with approximately 77 infected individuals, 36 susceptible individuals, and the rest of the population in the recovered category. The total population number drops right away and slowly rises back up due to the positive difference between the birth rate and natural death rate. Refer to figures 3.4 through 3.6.

The 208-week high rate shedding simulation mirrored the 24 week simulation. It quickly reached an endemic state of about 77 infected individuals, and 36

susceptible individuals, with the rest of the population in the recovered category. Interestingly, the plots show no major oscillations after the initial peak of the infected population. This will be discussed in Section 5.1. Refer to Figures 3.7 through 3.9.

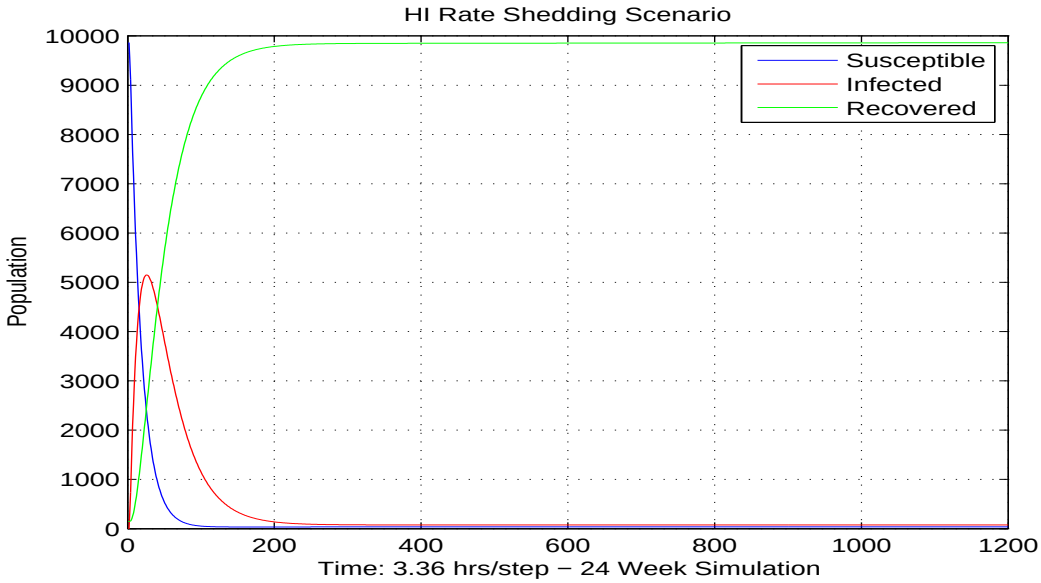

Figure 3.4: HI Rate Shedding Simulation - Human Populations

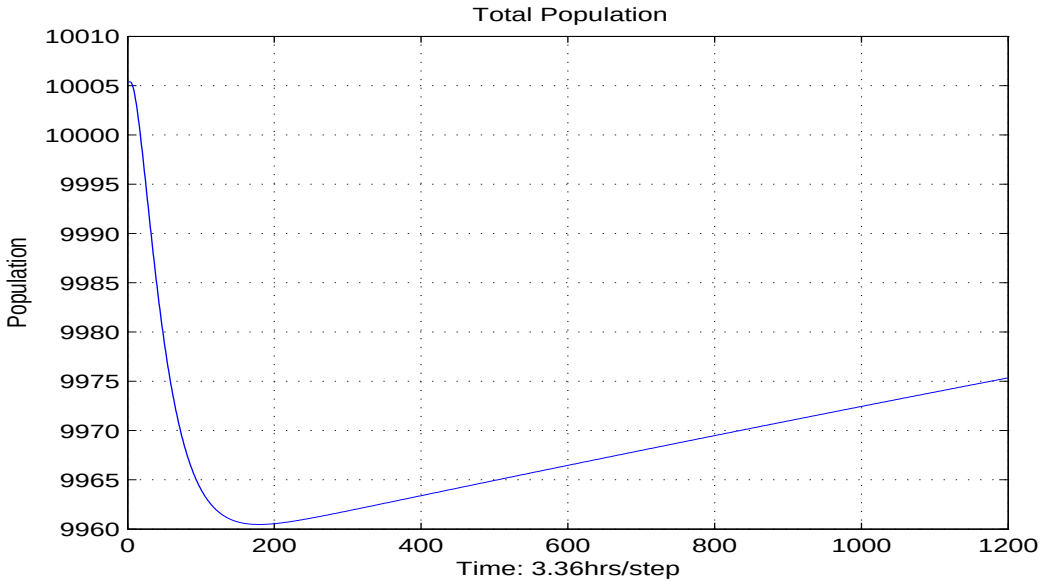

Figure 3.5: HI Rate Shedding Simulation - Total Human Population

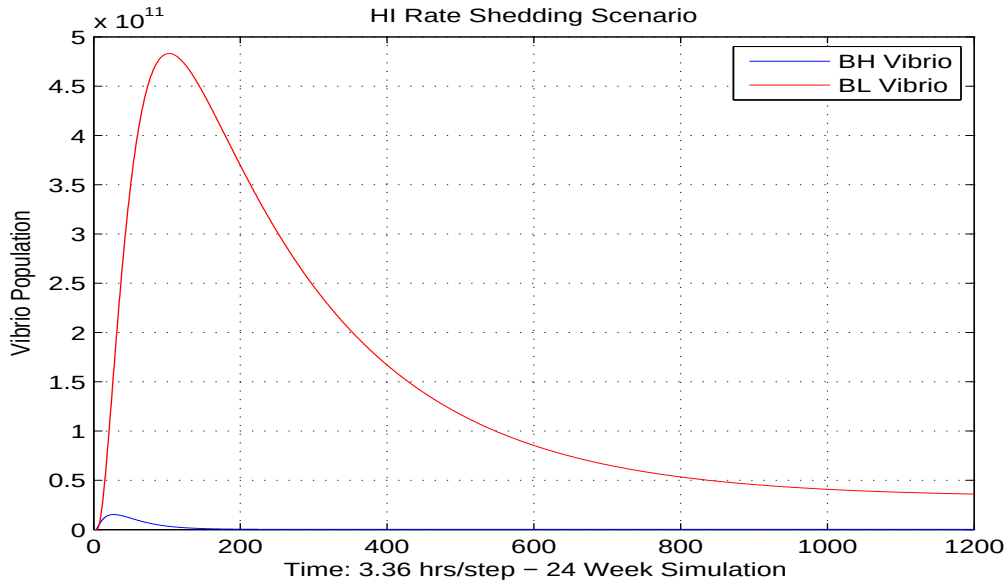

Figure 3.6: HI Rate Shedding Simulation - Vibrio Populations

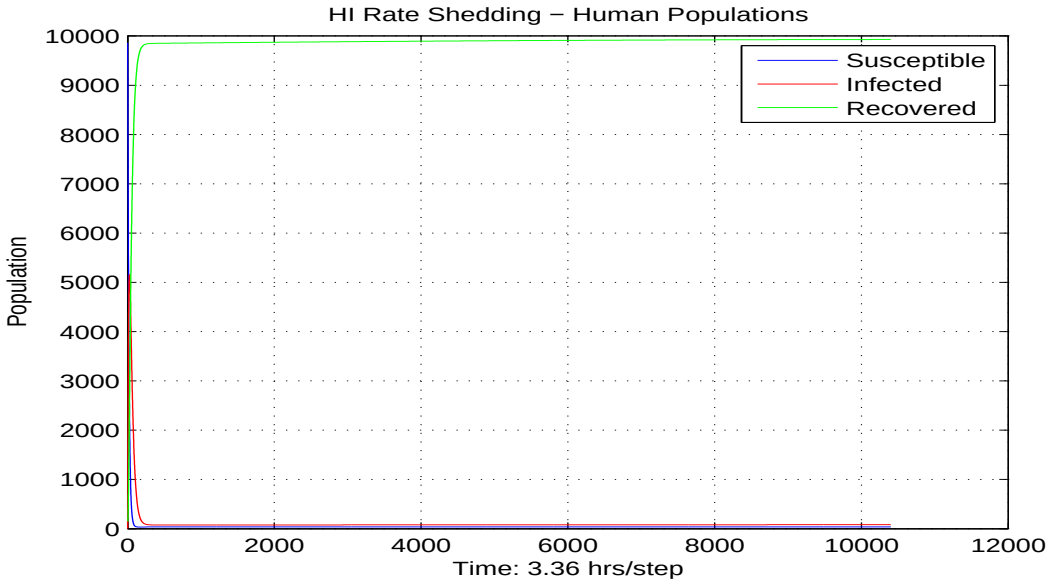

Figure 3.7: HI Rate Shedding Simulation 208 Weeks - Human Populations

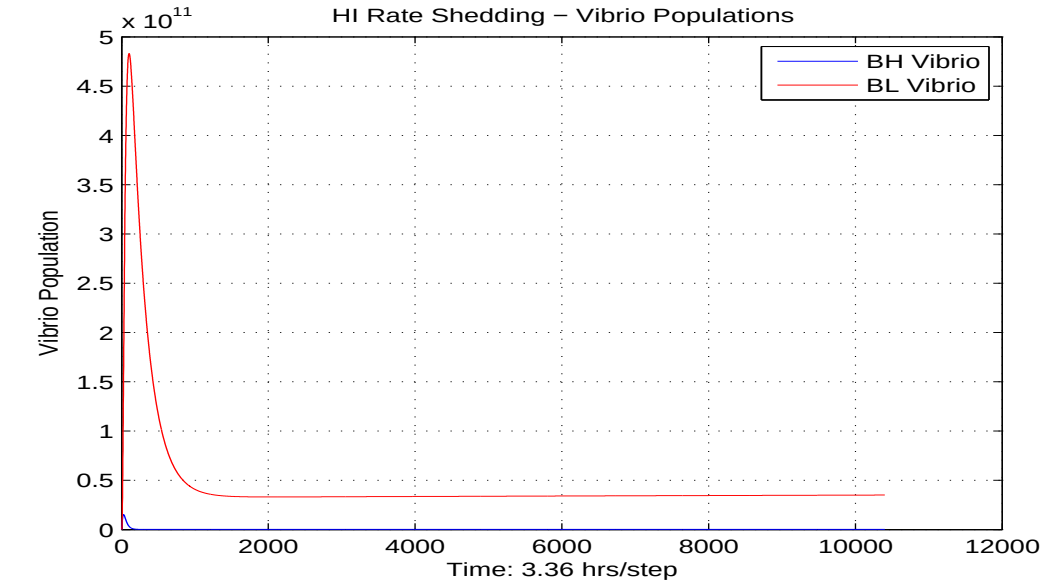

Figure 3.8: HI Rate Shedding Simulation 208 Weeks - Vibrio Populations

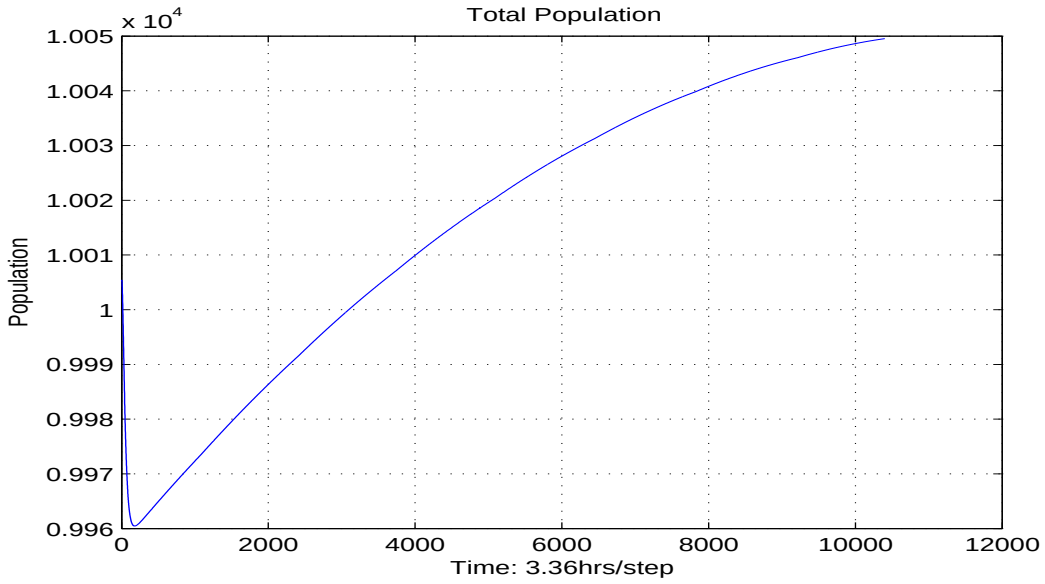

Figure 3.9: HI Rate Shedding Simulation 208 Weeks - Total Human Population

3.3.3 Low High Rate Shedding Simulation

Parameter values for the low rate shedding simulation are identical to those of the high rate shedding simulation, except for the shedding rate, (ξ), which is set to $\xi = 10^2$ cells/(ml*day*human). Compared to the 24-week simulation of HI Rate shedding, the Lo Rate shedding simulation produces a much more mild outbreak with the infected population peaking at approximately 1500 people at week six. The outbreak curve is also spread across 12 weeks, giving humanitarian agencies more time to react. Note that the susceptible population increases slightly at the end of the time frame. This is an unexpected behavior which will be discussed in Section 5.1. Refer to figures 3.10 through 3.12.

The Lo rate shedding simulation for 208 weeks produced a strong epidemic peak of approximately 1500 people, followed by decaying oscillations to an equilibrium (endemic state) of about 6 infected individuals. The endemic state is one in which the disease persists with a constant number of susceptible, infected, and recovered individuals. This agrees with mathematical theory predicting decaying oscillations [2]. Refer to Figures 3.13 through 3.15.

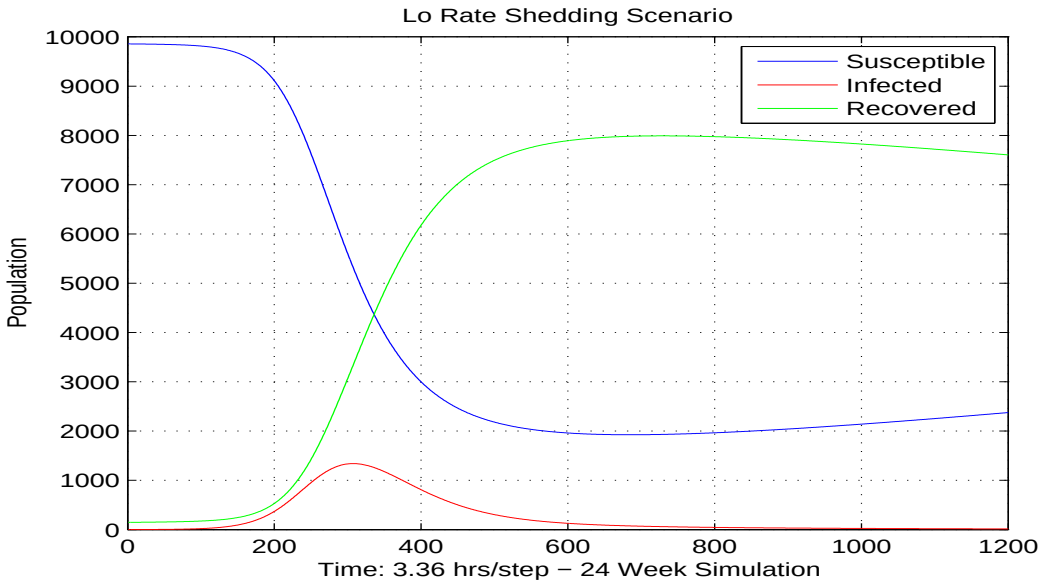

Figure 3.10: Lo Rate Shedding Simulation - Human Populations

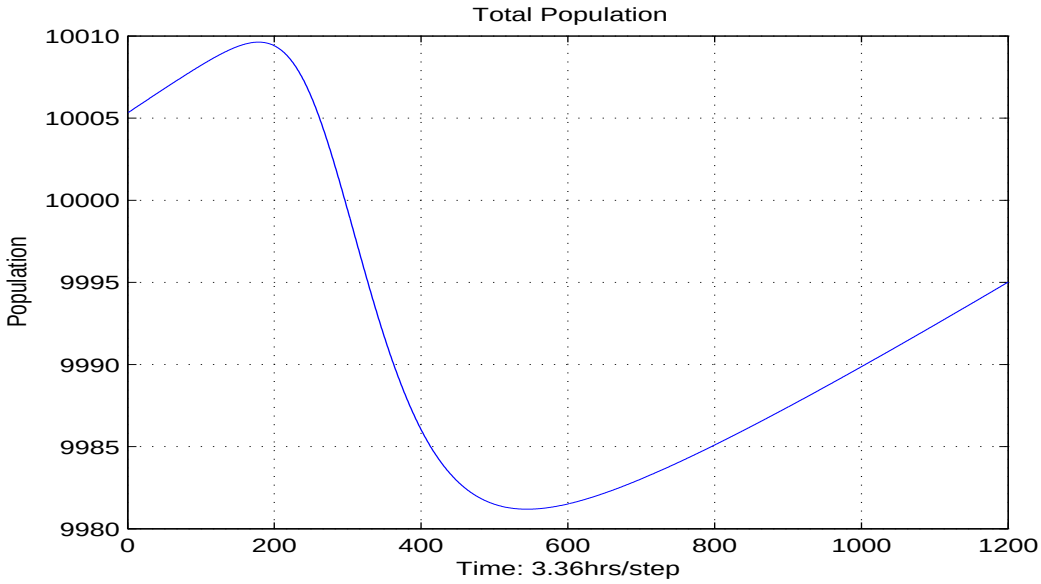

Figure 3.11: Lo Rate Shedding Simulation - Total Human Population

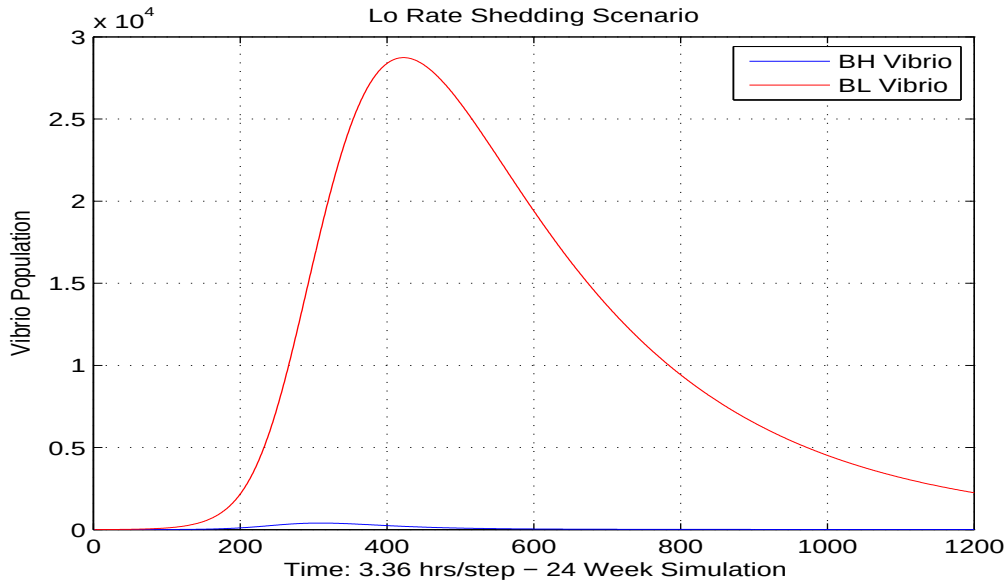

Figure 3.12: Lo Rate Shedding Simulation - Vibrio Populations

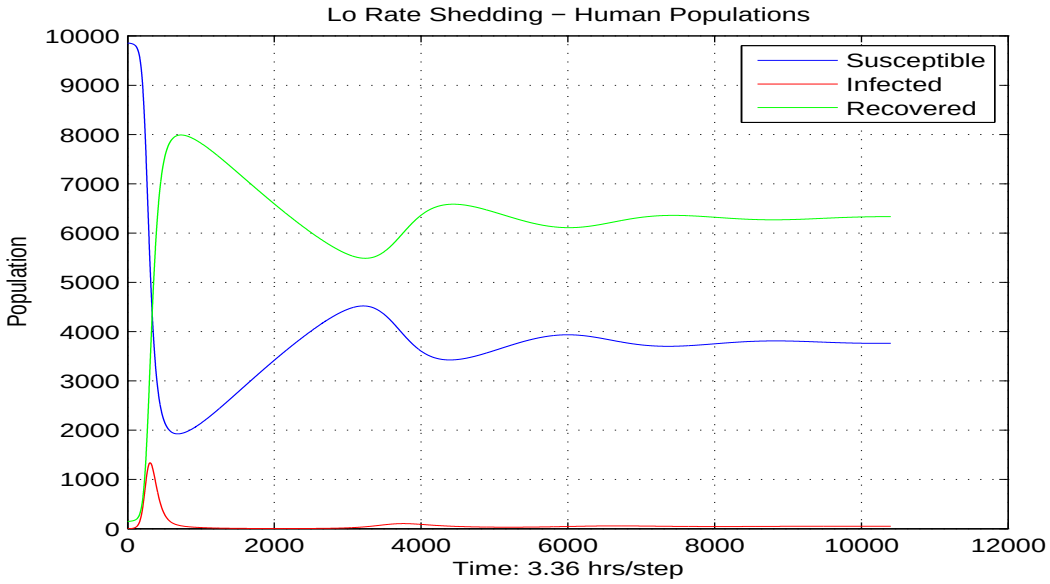

Figure 3.13: Low Rate Shedding Simulation 208 Weeks - Human Populations

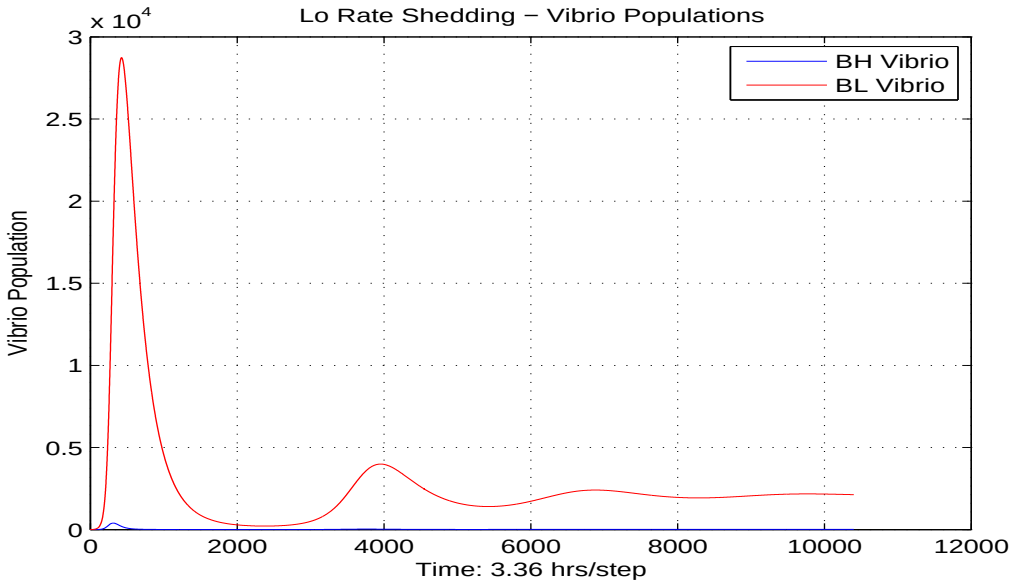

Figure 3.14: Low Rate Shedding Simulation 208 Weeks - Vibrio Populations

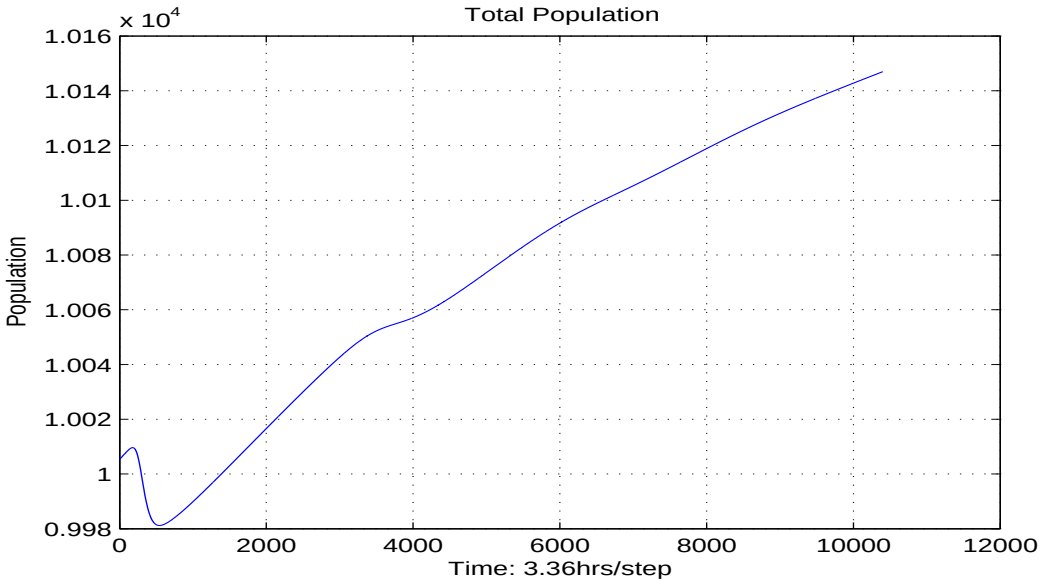

Figure 3.15: Lo Rate Shedding Simulation 208 Weeks - Total Human Population

CHAPTER 4

Second Order Method for the Age-structured Cholera Model

The upwind scheme done in the previous chapter (Equations 3.7 to 3.9) uses forward differencing in time and backward differencing in space, both of which are first order accurate. We improve on the order of accuracy by using the Lax-Wendroff scheme which is second order accurate in both time and space. The scheme is based in the following function:

$$u_{i,j+1} = f(u_{i-1,j}, u_{i,j}, u_{i+1,j}) \tag{4.1}$$

Since the computational molecule is triangular in shape it can handle the entire space(age)-time (a_i, t_j) grid except for the right hand side boundary (the left hand side are the given boundary values). There are many ways to solve this. For example, the second order accurate Upwind Scheme, which can handle the right hand boundary, can be combined with with the Lax-Wendroff scheme. We choose instead to simply extend the simulation out twice as far in the i (age) direction, allowing us to obtain the values of the right hand boundary.

In this chapter we derive both the Lax-Wendroff scheme, and a simplified version of it. We then compare the simulation output of the first order model with the ones of the Lax-Wendroff schemes.

4.1 The Lax-Wendroff Scheme

The first order model Equations 2.15 through 2.17) are nonhomogeneous hyperbolic equations of the form:

$$u_t = -\alpha u_x + f \tag{4.2}$$

Recall the Taylor series for $u(x, t+k)$, where $u(x,t)$ is a solution to the non-homogeneous equation 4.2, is given by

$$u(x, t+k) = u(x,t) + k u_t(x,t) + \frac{k^2}{2} u_{tt}(x,t) + O(k^3) \tag{4.3}$$

Differentiating equation 4.2 with respect to t, we get

$$u_{tt} = -\alpha u_{tx} + f_t \tag{4.4}$$

Differentiating equation 4.2 with respect to x, we get

$$u_{tx} = -\alpha u_{xx} + f_x \tag{4.5}$$

Combining equations 4.4 and 4.5 we get

$$u_{tt} = \alpha^2 u_{xx} - \alpha f_x + f_t. \tag{4.6}$$

The Taylor series 4.3 then becomes

$$u(x, t+k) = u(x,t) - \alpha k u_x(x,t) + \frac{\alpha^2 k^2}{2} u_{xx}(x,t) + kf - \frac{\alpha k^2}{2} f_x + \frac{k^2}{2} f_t + O(k^3) \tag{4.7}$$

Replacing the derivatives in x by second-order accurate differences and f_t by

a forward difference, we obtain

$$\begin{aligned} u(x,t+k) =& u(x,t) - \frac{\alpha k}{2h}[u(x+h,t) - u(x-h,t)] \\ &+ \frac{\alpha^2 k^2}{2h^2}[u(x+h,t) - 2u(x,t) + u(x-h,t)] \\ &+ \frac{k}{2}[f(x,t+k) + f(x,t)] - \frac{\alpha k^2}{4h}[f(x+h,t) - f(x-h,t)] \\ &+ O(kh^2) + O(k^3) \end{aligned}$$

This gives the Lax-Wendroff explicit scheme with second order accuracy in x and t:

$$\begin{aligned} u_{i,j+1} = u_{i,j} - \frac{k\alpha}{2h}(u_{i+1,j} - u_{i-1,j}) + \frac{k^2\alpha^2}{2h^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) \\ + \frac{k}{2}(f_{i,j+1} + f_{i,j}) - \frac{\alpha k^2}{4h}(f_{i+1,j} - f_{i-1,j}) \end{aligned} \tag{4.8}$$

or, equivalently,

$$\begin{aligned} \frac{1}{k}(u_{i,j+1} - u_{i,j}) + \frac{\alpha}{2h}(u_{i+1,j} - u_{i-1,j}) - \frac{k\alpha^2}{2h^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) \\ = \frac{1}{2}(f_{i,j+1} + f_{i,j}) - \frac{\alpha k}{4h}(f_{i+1,j} - f_{i-1,j}) \end{aligned} \tag{4.9}$$

Using the Lax-Wendroff Scheme, the corresponding second order finite difference equations for the age-structured model are below.

$$\begin{aligned} \frac{1}{k}(S_{i,j+1} - S_{i,j}) + \frac{\alpha}{2h}(S_{i+1,j} - S_{i-1,j}) - \frac{\alpha^2 k}{2h^2}(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}) = \\ \frac{1}{2}[\Lambda - (\beta_L\lambda_L + \beta_H\lambda_H)S - bS + \omega R]_{i,j+1} \\ + \frac{1}{2}[\Lambda - (\beta_L\lambda_L + \beta_H\lambda_H)S - bS + \omega R]_{i,j} \\ - \frac{k\alpha}{4h}[\Lambda - (\beta_L\lambda_L + \beta_H\lambda_H)S - bS + \omega R]_{i+1,j} \\ + \frac{k\alpha}{4h}[\Lambda - (\beta_L\lambda_L + \beta_H\lambda_H)S - bS + \omega R]_{i-1,j} \end{aligned} \tag{4.10}$$

$$\begin{aligned}\frac{1}{k}(I_{i,j+1}-I_{i,j})+\frac{\alpha}{2h}(I_{i+1,j}-I_{i-1,j})-\frac{\alpha^2k}{2h^2}(I_{i+1,j}-2I_{i,j}+I_{i-1,j})=\\ \frac{1}{2}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i,j+1}\\ +\frac{1}{2}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i,j}\\ -\frac{k\alpha}{4h}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i+1,j}\\ +\frac{k\alpha}{4h}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i-1,j}\end{aligned}\tag{4.11}$$

$$\begin{aligned}\frac{1}{k}(R_{i,j+1}-R_{i,j})+\frac{\alpha}{2h}(R_{i+1,j}-R_{i-1,j})-\frac{\alpha^2k}{2h^2}(R_{i+1,j}-2R_{i,j}+R_{i-1,j})=\\ \frac{1}{2}[\gamma_1(1-u)I+\gamma_2uI-bR-\omega R]_{i,j+1}\\ +\frac{1}{2}[\gamma_1(1-u)I+\gamma_2uI-bR-\omega R]_{i,j}\\ -\frac{k\alpha}{4h}[\gamma_1(1-u)I+\gamma_2uI-bR-\omega R]_{i+1,j}\\ +\frac{k\alpha}{4h}[\gamma_1(1-u)I+\gamma_2uI-bR-\omega R]_{i-1,j}\end{aligned}\tag{4.12}$$

By putting the terms with $(i,j+1)$ on the left hand side, we can rewrite the equations (4.10)-(4.12) as follows

$$\begin{aligned}[\frac{1}{k}+\frac{1}{2}(\beta_L\lambda_L+-\beta_H\lambda_H)+b]S_{i,j+1}-\frac{1}{2}\omega R_{i,j+1}=\\ \frac{1}{k}S_{i,j}+\frac{1}{2}\Lambda-\frac{\alpha}{2h}(S_{i+1,j}-S_{i-1,j})+\frac{\alpha^2k}{2h^2}(S_{i+1,j}-2S_{i,j}+S_{i-1,j})+\frac{1}{2}\Lambda\\ +\frac{1}{2}[\Lambda-(\beta_L\lambda_L+-\beta_H\lambda_H)S-bS+\omega R]_{i,j}\\ -\frac{k\alpha}{4h}[\Lambda-(\beta_L\lambda_L+-\beta_H\lambda_H)S-bS+\omega R]_{i+1,j}\\ +\frac{k\alpha}{4h}[\Lambda-(\beta_L\lambda_L+-\beta_H\lambda_H)S-bS+\omega R]_{i-1,j}\end{aligned}\tag{4.13}$$

$$\begin{aligned}[\frac{1}{k}+\frac{1}{2}(b+(1-h)\Delta+\gamma_1(1-u)I+\gamma_2u]I_{i,j+1}-\frac{1}{2}(\beta_L\lambda_L+\beta_H\lambda_H)S_{i,j+1}=\\ \frac{1}{k}I_{i,j}-\frac{\alpha}{2h}(I_{i+1,j}-I_{i-1,j})+\frac{\alpha^2k}{2h^2}(I_{i+1,j}-2I_{i,j}+I_{i-1,j})\\ +\frac{1}{2}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i,j}\\ -\frac{k\alpha}{4h}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i+1,j}\\ +\frac{k\alpha}{4h}[(\beta_L\lambda_L+\beta_H\lambda_H)S-bI-(1-h)\Delta I-\gamma_1(1-u)I-\gamma_2uI]_{i-1,j}\end{aligned}\tag{4.14}$$

$$
\begin{aligned}
&[\frac{1}{k}+\frac{1}{2}(b+\omega)]R_{i,j+1}-\frac{1}{2}[\gamma_1(1-u)+\gamma_2 u]I_{i,j+1}= \\
&\quad \frac{1}{k}R_{i,j}-\frac{\alpha}{2h}(R_{i+1,j}-R_{i-1,j})+\frac{\alpha^2 k}{2h^2}(R_{i+1,j}-2R_{i,j}+R_{i-1,j}) \\
&+\frac{1}{2}[\gamma_1(1-u)I+\gamma_2 uI-bR-\omega R]_{i,j} \\
&-\frac{k\alpha}{4h}[\gamma_1(1-u)I+\gamma_2 uI-bR-\omega R]_{i+1,j} \\
&+\frac{k\alpha}{4h}[\gamma_1(1-u)I+\gamma_2 uI-bR-\omega R]_{i-1,j}
\end{aligned} \tag{4.15}
$$

This gives a system of equations $A\mathbf{x}=\mathbf{b}$, where

$$
A=\begin{pmatrix} c_1 & 0 & c_2 \\ c_4 & c_3 & 0 \\ 0 & c_6 & c_5 \end{pmatrix}, \qquad x=\begin{pmatrix} S_{i,j+1} \\ I_{i,j+1} \\ R_{i,j+1} \end{pmatrix}, \qquad b=\begin{pmatrix} d_1 \\ d_2 \\ d_3 \end{pmatrix}
$$

and the constants d_1, d_2, d_3 are the right hand side of equations 4.13, 4.14, and 4.15, respectively, and

$$
\begin{aligned}
c_1 &= [\frac{1}{k}+\frac{1}{2}(\beta_L\lambda_L+-\beta_H\lambda_H)+b] \\
c_2 &= -\frac{1}{2}\omega \\
c_3 &= [\frac{1}{k}+\frac{1}{2}(b+(1-h)\Delta+\gamma_1(1-u)I+\gamma_2 u] \\
c_4 &= \frac{1}{2}(\beta_L\lambda_L+\beta_H\lambda_H) \\
c_5 &= [\frac{1}{k}+\frac{1}{2}(b+\omega)] \\
c_6 &= \frac{1}{2}[\gamma_1(1-u)+\gamma_2 u].
\end{aligned}
$$

The system can be solved numerically by implementing the Jacobi or Gauss-Seidel methods [16]. Alternatively, by substitution we can also convert the equations (4.13)-(4.15) into their explicit forms. For a detailed explanation of the substitution process and its associated Matlab code for the Lax-Wendroff scheme, see Appendix C and D.

4.2 Simplified Lax-Wendroff Scheme

The following is a simplified version of the Lax-Wendroff scheme [17] that is easier to solve explicitly. It is second order accurate in x and first order accurate in t.

Equation 4.7 is repeated here for convenience

$$u(x,t+k) = u(x,t) - \alpha k u_x(x,t) + \frac{\alpha^2 k^2}{2} u_{xx}(x,t) + kf - \frac{\alpha k^2}{2} f_x + \frac{k^2}{2} f_t + O(k^3) \quad (4.16)$$

Removing the space and time derivatives of f from the above equation we get a new equation that is second order accurate for x and first order accurate for t:

$$u(x,t+k) = u(x,t) - \alpha k u_x(x,t) + \frac{\alpha^2 k^2}{2} u_{xx}(x,t) + kf + O(k^2) \quad (4.17)$$

Replacing u_x and u_{xx} with central difference equations we get the Simplified Lax-Wendroff explicit scheme.

$$u_{i,j+1} = u_{i,j} - \frac{k\alpha}{2h}(u_{i+1,j} - u_{i-1,j}) + \frac{k^2\alpha^2}{2h^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + kf_{i,j} \quad (4.18)$$

or, equivalently,

$$\frac{1}{k}(u_{i,j+1} - u_{i,j}) + \frac{\alpha}{2h}(u_{i+1,j} - u_{i-1,j}) - \frac{k\alpha^2}{2h^2}(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) = f_{i,j} \quad (4.19)$$

Implementing the simplified Lax-Wendroff scheme, the age-structured model can be solved by solving the following equations:

$$\begin{gathered}\frac{1}{k}(S_{i,j+1} - S_{i,j}) + \frac{\alpha}{2h}(S_{i+1,j} - S_{i-1,j}) - \frac{\alpha^2 k}{2h^2}(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}) = \\ [\Lambda - (\beta_L \lambda_L + \beta_H \lambda_H)S - bS + \omega R]_{i,j}\end{gathered} \quad (4.20)$$

$$\frac{1}{k}(I_{i,j+1} - I_{i,j}) + \frac{\alpha}{2h}(I_{i+1,j} - I_{i-1,j}) - \frac{\alpha^2 k}{2h^2}(I_{i+1,j} - 2I_{i,j} + I_{i-1,j}) = \\ [(\beta_L \lambda_L + \beta_H \lambda_H)S - bI - (1-h)\Delta I - \gamma_1(1-u)I - \gamma_2 uI]_{i,j} \tag{4.21}$$

$$\frac{1}{k}(R_{i,j+1} - R_{i,j}) + \frac{\alpha}{2h}(R_{i+1,j} - R_{i-1,j}) - \frac{\alpha^2 k}{2h^2}(R_{i+1,j} - 2R_{i,j} + R_{i-1,j}) = \\ [(\gamma_1(1-u) + \gamma_2 u)I - bR - \omega R]_{i,j} \tag{4.22}$$

4.3 Improving the Accuracy of the First Order Model ODEs

In the first order model described in Chapter 3, the ODE

$$\frac{dB_H}{dt} = \int_0^A \xi\eta I da - \chi B_H$$

was implemented using the second-order accurate trapezoidal rule, so this same implementation is also used in the second order model. The ODE

$$\frac{dB_L}{dt} = \chi B_H - \delta_L B_L$$

was solved by using the first order Euler Method (3.10). We improve this to second order accuracy by implementing the second order *Midpoint Method* [5]. The method is defined as follows: Given $dy/dx = f(x,y)$, $a \le x \le b$ with initial value $y(a) = y_0$, let

$$c_1 = f(x_i, y_i), \quad c_2 = f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}c_1 h),$$

where $h = (b-a)/N$ is the step size. Then $y_{i+1} = y_i + c_2 h$.

Applying the Midpoint Method to equation (2.19), we get

$$c_1 = \left[\chi B_{H_j} - \delta_L B_{L_j}\right]$$
$$\frac{B_{L_{j+1}} - B_{L_j}}{k} = \chi \frac{B_{H_j} + B_{H_{j+1}}}{2} - \delta_L (B_{L_j} + \frac{1}{2} c_1 k)$$

4.4 Comparison of First Order Model with Lax Wendroff Models

To compare differences in accuracy between the first order model in Chapter 3 and the two Lax-Wendroff models, all three models were run with the same settings, for the high rate shedding scenario (i.e. $\xi = 1000000000$), and a time duration of four years. The models all agree closely with each other. The greatest difference occurs in the infected population at the extremum where the second order infected population peak is 100 people less than the first order peak 4.5. This corresponds to a difference of 2%. The following are graphs of each population category with magnified views in areas of greatest difference.

All three models were also run for the low rate shedding scenario. The differences between the model outputs were even smaller, so these plots were not included in this paper.

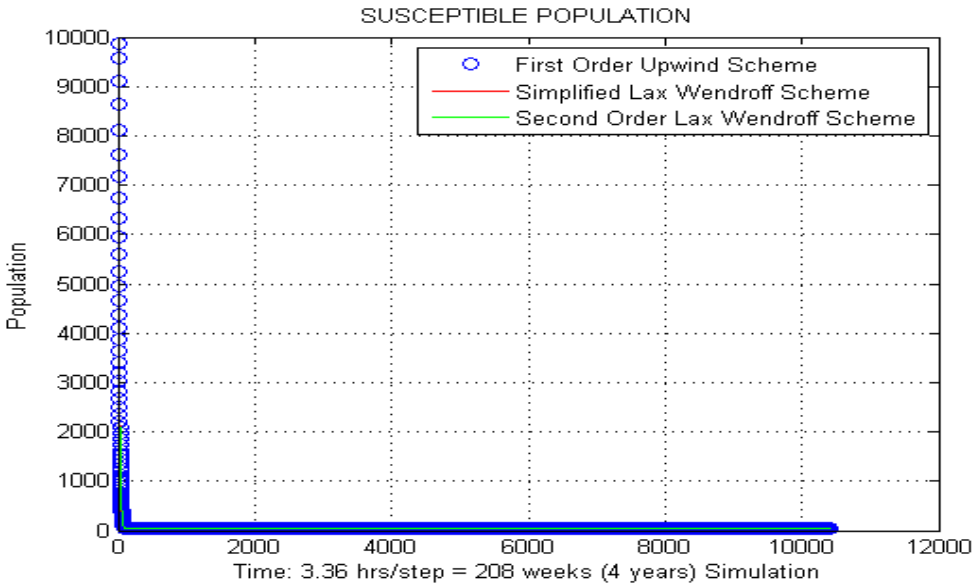

Figure 4.1: HI Rate Shedding - Susceptible Population

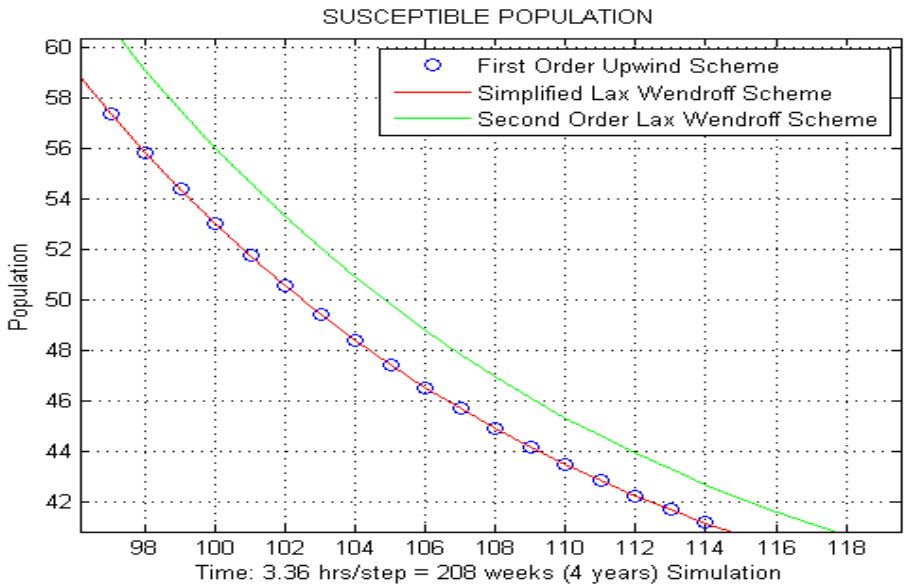

Figure 4.2: HI Rate Shedding - Susceptible Population Zoomed

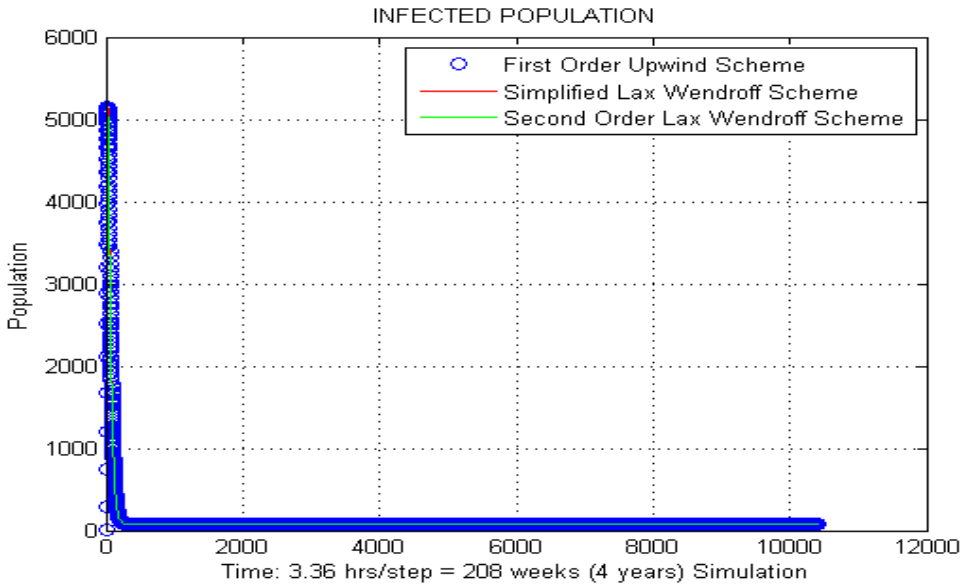

Figure 4.3: HI Rate Shedding - Infected Population

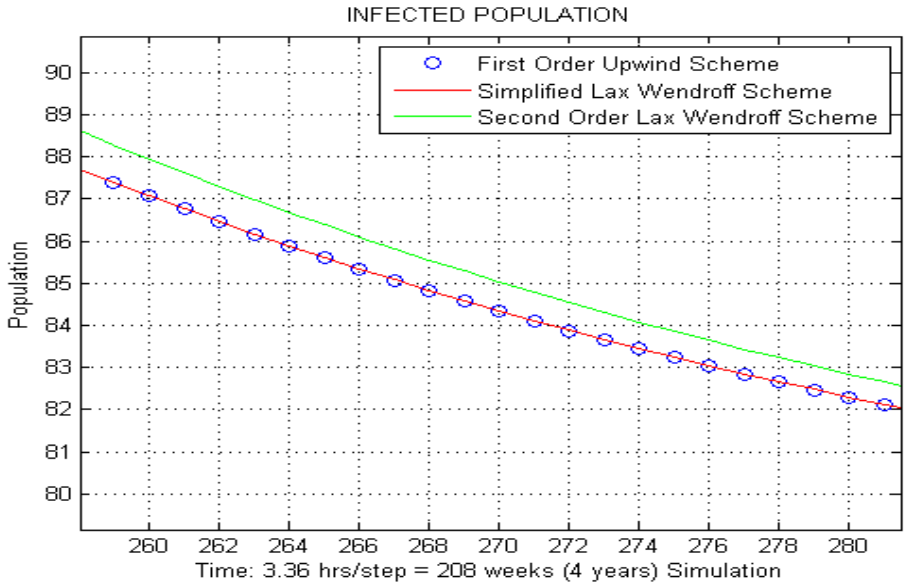

Figure 4.4: HI Rate Shedding - Infected Population Zoomed

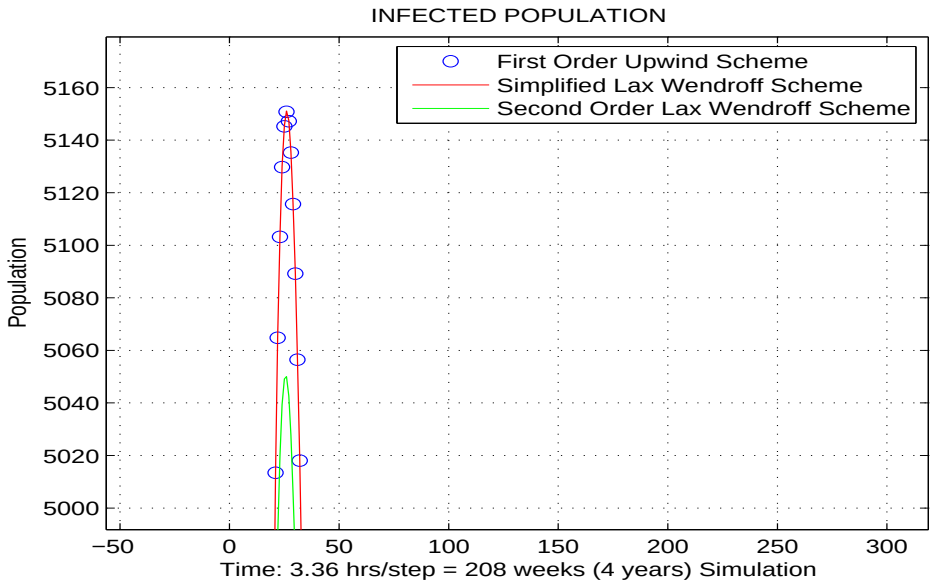

Figure 4.5: HI Rate Shedding - Infected Population Zoomed

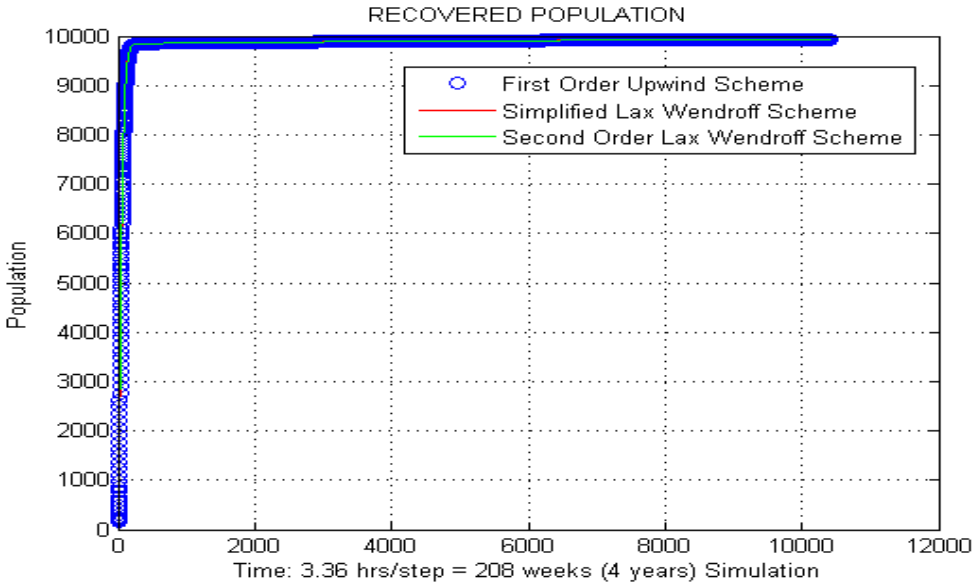

Figure 4.6: HI Rate Shedding - Recovered Population

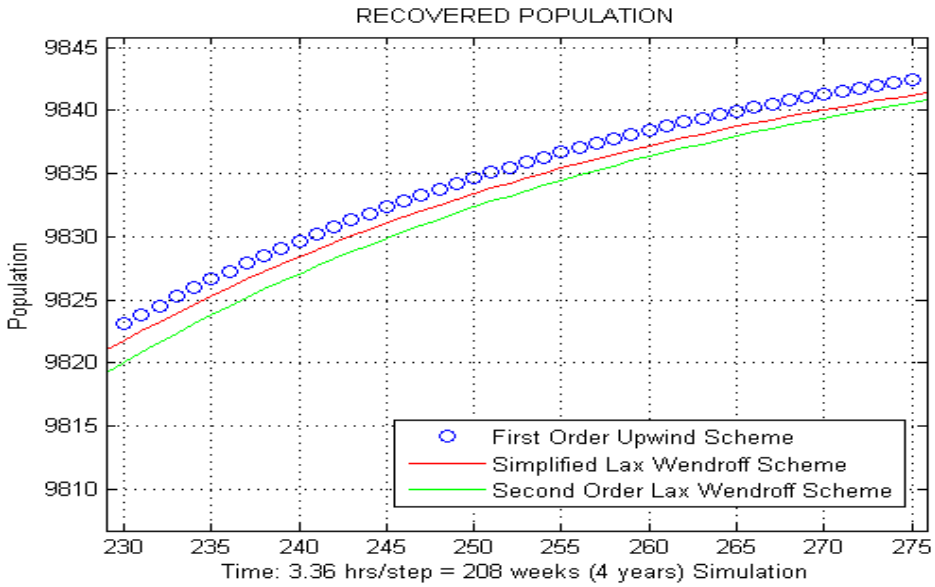

Figure 4.7: HI Rate Shedding - Recovered Population Zoomed

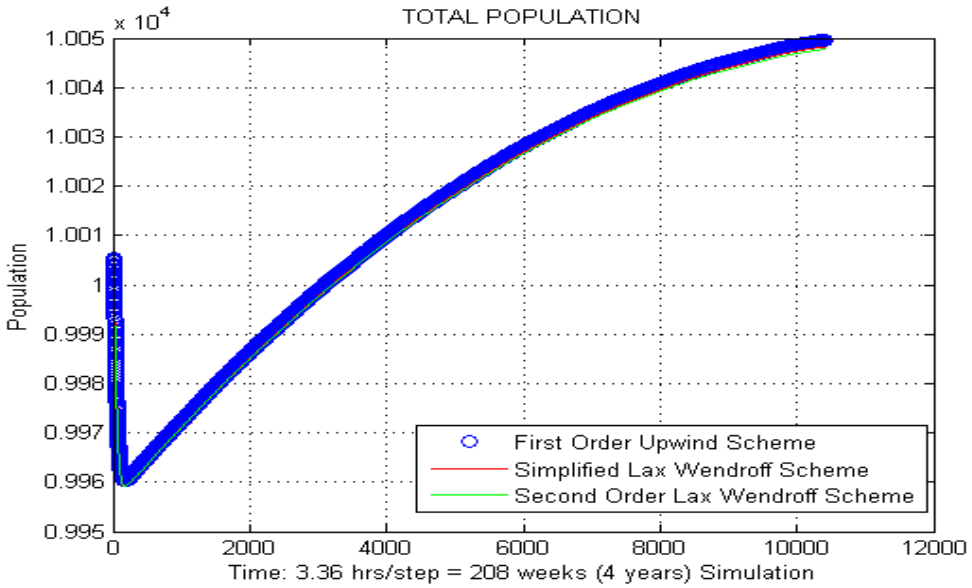

Figure 4.8: HI Rate Shedding - Total Population

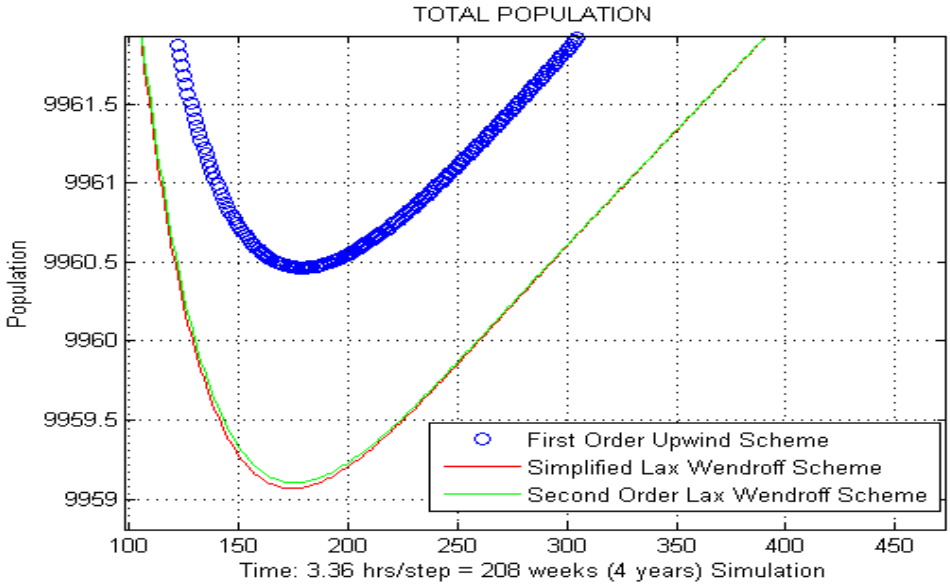

Figure 4.9: HI Rate Shedding - Total Population Zoomed

CHAPTER 5

Analysis on Model Parameters

In this chapter, we analyze several model parameters and their effects on the model output. The first parameter we hone in on is the immunity waning parameter ω. We notice a weakness in the previously defined parameter value of ω which produces unrealistic behavior in simulation curves. We propose a solution which produces a more realistic behavior in both short and long term simulations. In the following section we conduct a sensitivity analysis on various model parameters and their impacts on outbreak severity both in the high rate and low rate shedding scenarios.

5.1 Analysis on Immunity Waning Parameter

From our simulation results presented in Chapter 3, we notice a strange behavior in the Lo Rate shedding simulation as mentioned in [9]. The susceptible population increases near the end of the 24 week time frame. This phenomenon can be seen in both the susceptible population curve with an unexpected increase, and the recovered population curve with an unexpected decrease. See Figure 3.10. The susceptible population is expected to increase slightly due to the infant population constantly losing immunity at one year of age, but the increase should not be as large as the plot shows.

The strange behavior has to do with the way immunity waning parameter ω

is defined as a piecewise step function given by [9]:

$$\omega = \begin{cases} 1/365 \text{ per day} & a \leq 10 \text{ years old} \\ 1/(2 * 365) \text{ per day} & a > 10 \text{ years old.} \end{cases} \tag{5.1}$$

With this definition of ω some people become susceptible immediately after they have recovered from infection. However, studies show that the recovered population is immune for at least a year before becoming susceptible (Ref [13], [22], [8]). Using (5.1), our simulation shows that some people get infected more than once within a short time frame. The most obvious problem shows up in the Hi Rate shedding simulation over a 24 week time period. This scenario produces an infected population curve of Figure 5.1. Here the cumulative infected population totals over 12000 people (Figure 5.2), which is well over the initial population size of 10000. Essentially, over 2000 repeat infections unrealistically occur within the 24 week time frame of the outbreak. This indicates that the use of (5.1) to model the immunity waning is not quite accurate.

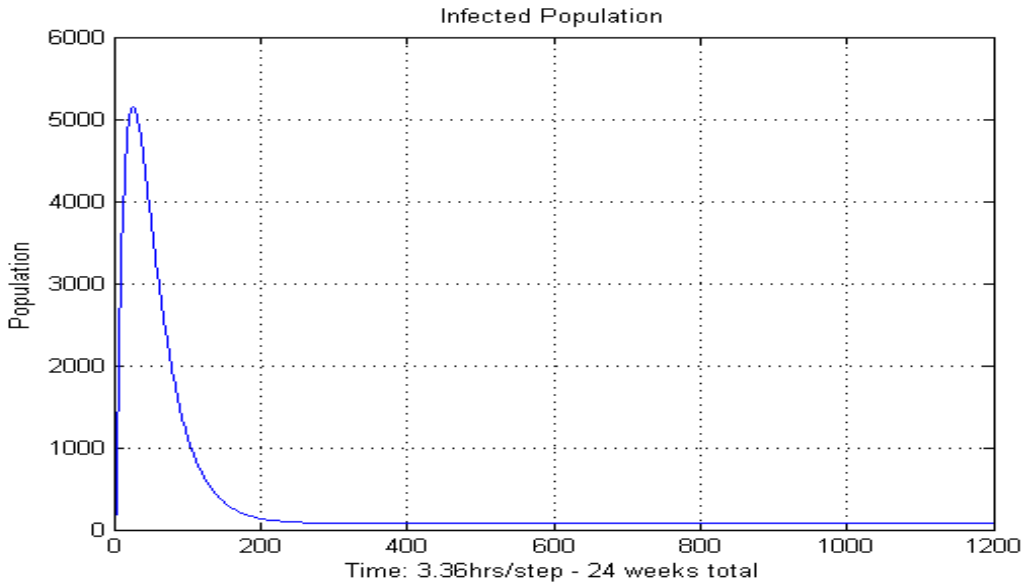

Figure 5.1: Hi Rate Shedding - Infecteds - using definition (5.1)

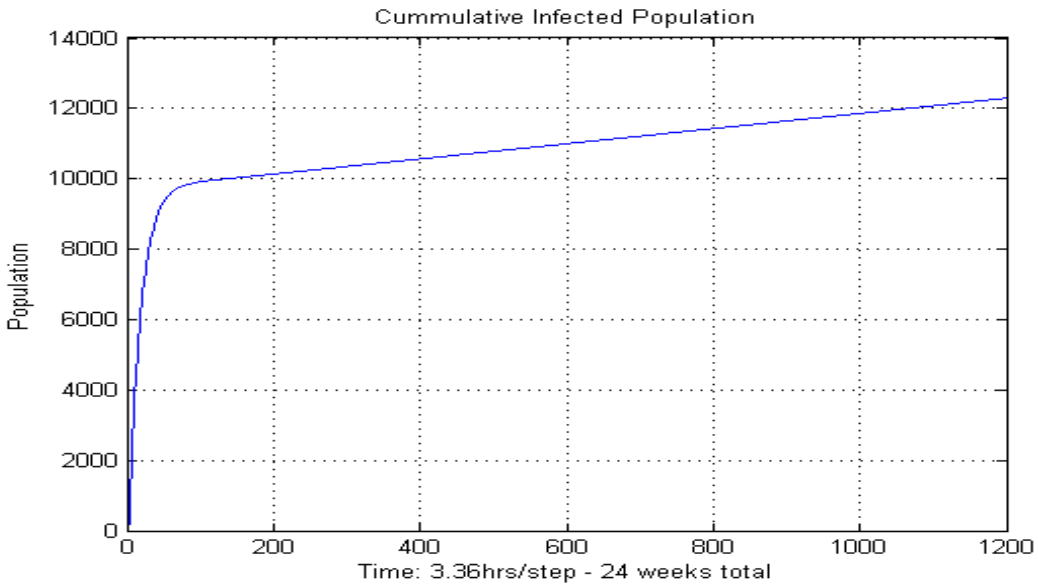

Figure 5.2: Hi Rate Shedding - Cumulative Infecteds - using definition (5.1)

To overcome this unwanted behavior, we modify our algorithm by subtracting the number of infecteds that occurred exactly one immunity period earlier (immunity period of 1 year for children under 10 years old and 2 years for those older than 10)

from the recovered population, and adding that value to the susceptible population. By doing this, our simulation plot of the cumulative infected population does not grow beyond the initial population size, which is more realistic. Refer to Figure 5.3.

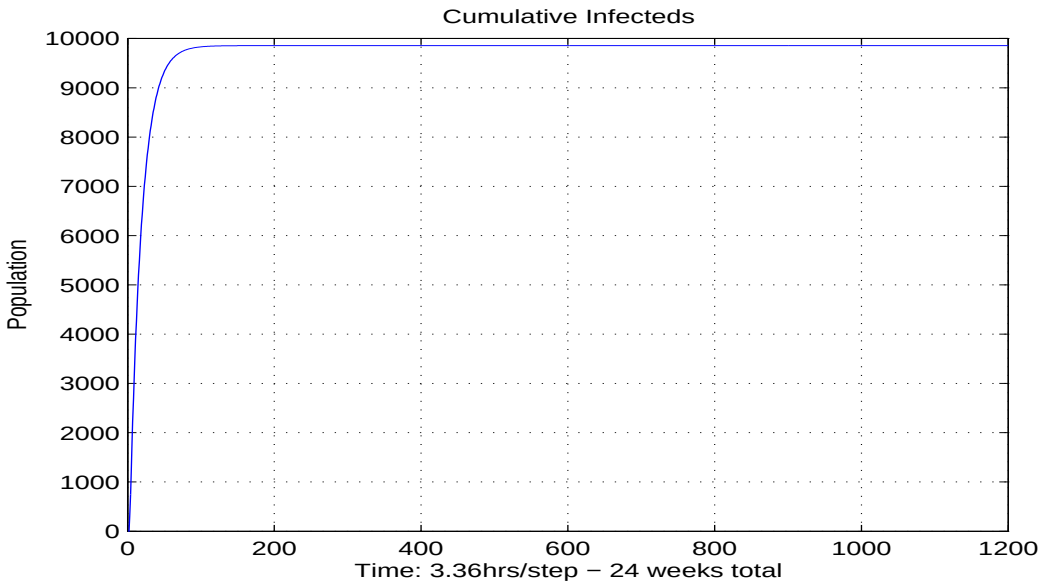

Figure 5.3: HI Rate Shedding - Cumulative Infecteds with our proposed technique

Using the same technique, our Lo Rate shedding simulation also removes the unwanted behavior in the susceptible and recovered population curves. Refer to Figures 5.4 and 5.5 and compare these to Figure 3.10.

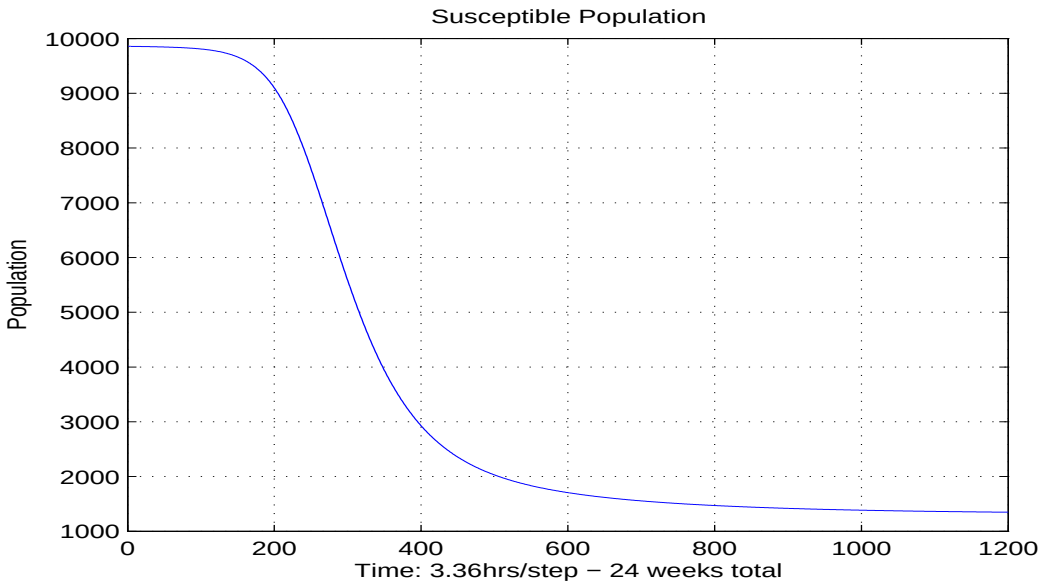

Figure 5.4: Lo Rate Shedding - Susceptible Population with our proposed technique

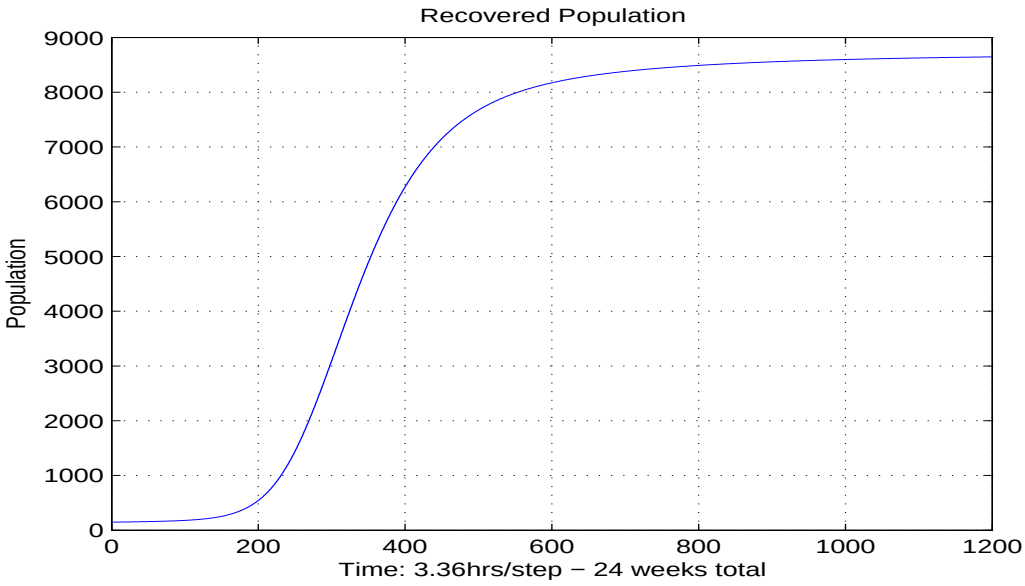

Figure 5.5: Lo Rate Shedding - Recovered Population with our proposed technique

We also ran a long term simulation out to 208 weeks for both HI and Lo rate shedding scenarios implementing this technique. For the Lo Rate shedding scenario, two infection peaks were produced, and the second peak was more pronounced than

in previous simulations using (5.1). Compare Figures 3.13 and 5.6.

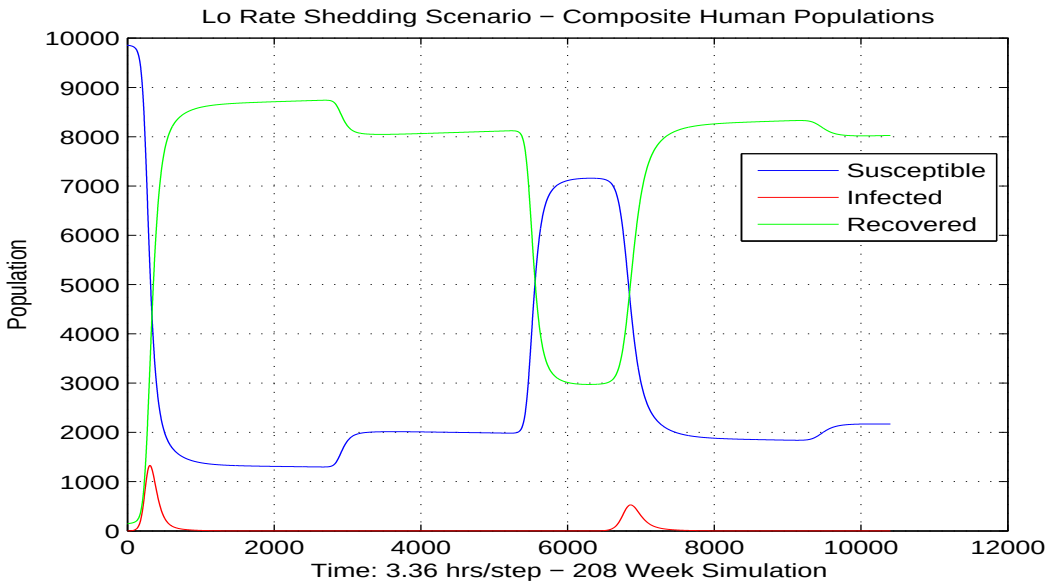

Figure 5.6: Lo Rate Shedding - Human Populations with our proposed technique

The HI Rate scenario produced two series of infection peaks. The first series represents the reinfection of adults within the population. These peaks are large and have a period of two years. The second series represents the reinfection of children within the population. These peaks are small and have a period of one year. The peaks oscillate, decay, and overlap.

This is a marked improvement for the HI rate simulation since the model with ω defined by (5.1) produced only a single peak, and no oscillations. Refer to Figures 3.7 and 5.7. With two series of oscillating and overlapping peaks, the results look noisy and therefore more realistic. If the adult immunity period is not a multiple of the child period, then the results would look even noisier.

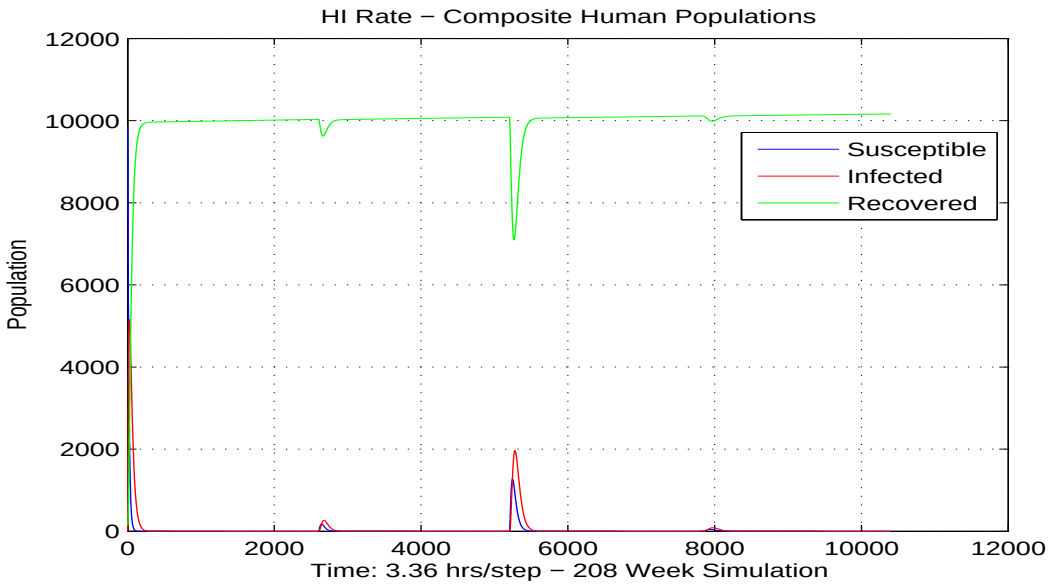

Figure 5.7: HI Rate Shedding - Human Populations with our proposed technique

5.2 Sensitivity Analysis on Various Model Parameters

Cholera-related parameter values can be difficult to determine and can vary considerably between populations [10, 12]. We conduct a sensitivity analysis to determine the impact of changes to key model parameters on various output measures. Three model outputs were chosen as measures of outbreak severity: peak infected population (maximum population infected at any one time), total infected population, and total mortalities.

In our analysis, each key model parameter (Table 3.1) was varied by $\pm$ 5% and $\pm$ 10% while keeping all other parameters at their default values for a 24 week simulation time window. In all, 22 parameters were tested, resulting in a total of 89 simulations. Model output for measures of outbreak severity was compared to the default output from Table 5.1.

Table 5.1: Outbreak severity - default output values for high and low rate shedding

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
HI Rate Shedding	5150.82	12293.11	65.08
Lo Rate Shedding	1340.29	8927.13	45.38

5.2.1 Sensitivity of High Rate Shedding Scenario Parameters

Across the ranges of parameters assessed, the model output was most sensitive to $h(a,t)$, the oral rehydration therapy. Although $h(a,t)$ had little influence on the peak of the infected population, and total infected population, it had a strong influence on total mortalities. Case projections for total mortalities varied from 90% lower to 88% higher than the base case. This indicates that oral hydration therapy is highly effective as an intervention for high rate shedding epidemics.

Model output was moderately sensitive to γ_1, the recovery rate of untreated cholera. γ_1 had a only mild effect on infecteds, but a stronger effect on total mortalities (9% lower to 11% higher).

The model was strongly robust to all other parameters with values changing less than 2% from the nominal value. This tells us that these parameters are less important to estimate accurately.

We list the sensitivity data of the high rate shedding scenario produced by our simulations in section 5.2.3 and present the plots of the most sensitive parameters in Section 5.2.5.

5.2.2 Sensitivity of Low Rate Shedding Scenario Parameters

As in the high rate shedding scenario, across the ranges of parameters assessed, the model output was most sensitive to $h(a,t)$, the oral rehydration therapy. Although

$h(a,t)$ had little influence on the peak of infected population, and total infected population, it had a strong influence on total mortalities. Case projections for total mortalities varied from 90% lower to 89% higher than the base case. This indicates that oral hydration therapy is highly effective as an intervention for low rate shedding epidemics as well.

Model output was moderately sensitive to $\kappa_L(a)$, $\kappa_H(a)$, $\beta_H(a)$, γ_1, η, and χ. All these parameters affect the peak of the infected population up to 16% lower to 19% higher compare to the base case. The parameter γ_1 also has a moderate effect on total mortalities (13% lower to 15% higher).

The model was fairly robust to all other parameters with values changing less than 2% from the nominal value. This indicates that these parameters are less important to estimate accurately.

Although more model parameters are moderately sensitive in the low rate shedding scenario, it should be noted that this scenario is much less critical because of its smaller negative effect on the population. Regardless, more studies need to be done to estimate accurate values for the low rate shedding scenario.

Refer to Section 5.2.4 for sensitivity data of the low rate shedding scenario, and Section 5.2.5 for a collection of plots of the most sensitive parameters.

5.2.3 Sensitivity Data - High Rate Shedding Scenario

In columns two, three, and four of each table, the numbers in parenthesis represent the percentage change from the nominal value listed in Table 5.1.

Table 5.2: Sensitivity of ξ - Rate of Shedding of Cholera Vibrios

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5150.56 (0.00%)	12293.06 (0.00%)	65.08 (0.00%)
95%	5150.69 (0.00%)	12293.09 (0.00%)	65.08 (0.00%)
105%	5150.93 (0.00%)	12293.13 (0.00%)	65.08 (0.00%)
110%	5151.04 (0.00%)	12293.15 (0.00%)	65.08 (0.00%)

Table 5.3: Sensitivity of b - Normal Mortality Rate of Humans

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5150.92 (0.00%)	12294.44 (0.01%)	65.09 (0.02%)
95%	5150.87 (0.00%)	12293.77 (0.01%)	65.08 (0.01%)
105%	5150.77 (0.00%)	12292.45 (0.01%)	65.07 (0.01%)
110%	5150.72 (0.00%)	12291.78 (0.01%)	65.07 (0.02%)

Table 5.4: Sensitivity of $\kappa_L(a)$ Half Saturation Constant of non-HI Vibrios

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5151.06 (0.00%)	12293.16 (0.00%)	65.08 (0.00%)
95%	5150.94 (0.00%)	12293.13 (0.00%)	65.08 (0.00%)
105%	5150.70 (0.00%)	12293.09 (0.00%)	65.08 (0.00%)
110%	5150.59 (0.00%)	12293.07 (0.00%)	65.08 (0.00%)

Table 5.5: Sensitivity of $\kappa_H(a)$ Half Saturation Constant (Hyper-Infective)

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
95%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
105%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
110%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)

Table 5.6: Sensitivity of $\beta_L(a)$ - Ingestion rate of non-HI vibrios at age a

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5049.03 (1.98%)	12288.87 (0.03%)	65.05 (0.04%)
95%	5099.98 (0.99%)	12291.04 (0.02%)	65.06 (0.02%)
105%	5199.49 (0.94%)	12295.08 (0.02%)	65.09 (0.02%)
110%	5246.23 (1.85%)	12296.95 (0.03%)	65.10 (0.04%)

Table 5.7: Sensitivity of $\beta_H(a)$ - Ingestion rate of HI vibrios at age a

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5049.03 (1.98%)	12288.68 (0.04%)	65.05 (0.05%)
95%	5099.73 (0.99%)	12290.95 (0.02%)	65.06 (0.02%)
105%	5199.48 (0.94%)	12295.17 (0.02%)	65.09 (0.02%)
110%	5246.75 (1.86%)	12297.12 (0.03%)	65.11 (0.04%)

Table 5.8: Sensitivity of γ_1 - Recovery Rate of Untreated Cholera

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5347.29 (3.81%)	12281.05 (0.10%)	72.08 (10.75%)
95%	5246.11 (1.85%)	12287.39 (0.05%)	68.40 (5.10%)
105%	5058.62 (1.79%)	12298.30 (0.04%)	62.07 (4.63%)
110%	4971.88 (3.47%)	12303.04 (0.08%)	59.32 (8.85%)

Table 5.9: Sensitivity of $h(a,t)$ - Oral Rehydration Therapy

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5142.31 (0.17%)	12278.55 (0.12%)	122.40 (88.08%)
95%	5146.56 (0.08%)	12285.79 (0.06%)	93.88 (44.26%)
105%	5155.08 (0.08%)	12300.51 (0.06%)	35.98 (44.72%)
110%	5159.35 (0.17%)	12307.99 (0.12%)	6.58 (89.90%)

Table 5.10: Sensitivity of δ_L - Death Rate of Vibrio in the Environment.

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
95%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
105%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)
110%	5150.82 (0.00%)	12293.11 (0.00%)	65.08 (0.00%)

Table 5.11: Sensitivity of η - Relative Amount of Stool Per Unit Time

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	5150.56 (0.00%)	12293.06 (0.00%)	65.08 (0.00%)
95%	5150.69 (0.00%)	12293.09 (0.00%)	65.08 (0.00%)
105%	5150.93 (0.00%)	12293.13 (0.00%)	65.08 (0.00%)
110%	5151.04 (0.00%)	12293.15 (0.00%)	65.08 (0.00%)

Table 5.12: Sensitivity of χ - Rate of Vibrio Moving from HI to non-HI State

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	5150.61 (0.00%)	12293.07 (0.00%)	65.08 (0.00%)
95%	5150.72 (0.00%)	12293.09 (0.00%)	65.08 (0.00%)
105%	5150.91 (0.00%)	12293.13 (0.00%)	65.08 (0.00%)
110%	5150.99 (0.00%)	12293.15 (0.00%)	65.08 (0.00%)

5.2.4 Sensitivity Data - Low Rate Shedding Scenario

In columns two, three, and four of each table, the numbers in parenthesis represent the percentage change from the nominal value listed in Table 5.1.

Table 5.13: Sensitivity of b - Normal Mortality Rate of Humans

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	1340.97 (0.05%)	8930.57 (0.04%)	45.40 (0.04%)
95%	1340.63 (0.03%)	8928.85 (0.02%)	45.39 (0.02%)
105%	1339.95 (0.03%)	8925.41 (0.02%)	45.37 (0.02%)
110%	1339.61 (0.05%)	8923.69 (0.04%)	45.36 (0.04%)

Table 5.14: Sensitivity of $\kappa_L(a)$ Half Saturation Constant of non-HI Vibrios

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	1533.67 (14.43%)	9230.10 (3.39%)	46.96 (3.48%)
95%	1435.26 (7.09%)	9082.30 (1.74%)	46.19 (1.78%)
105%	1248.71 (6.83%)	8764.05 (1.83%)	44.53 (1.87%)
110%	1160.59 (13.41%)	8592.46 (3.75%)	43.64 (3.83%)

Table 5.15: Sensitivity of $\kappa_H(a)$ Half Saturation Constant (Hyper-Infective)

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	1522.86 (13.62%)	9150.76 (2.51%)	46.54 (2.55%)
95%	1429.76 (6.68%)	9041.26 (1.28%)	45.97 (1.30%)
105%	1254.51 (6.40%)	8807.97 (1.33%)	44.76 (1.36%)
110%	1172.39 (12.53%)	8683.35 (2.73%)	44.12 (2.79%)

Table 5.16: Sensitivity of $\beta_L(a)$ - Ingestion rate of non-HI vibrios at age a

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	1328.83 (0.86%)	8838.19 (1.00%)	44.91 (1.03%)
95%	1334.57 (0.43%)	8883.32 (0.49%)	45.15 (0.51%)
105%	1346.02 (0.43%)	8969.67 (0.48%)	45.60 (0.49%)
110%	1351.69 (0.85%)	9010.99 (0.94%)	45.82 (0.97%)

Table 5.17: Sensitivity of $\beta_H(a)$ - Ingestion rate of HI vibrios at age a

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	1124.43 (16.11%)	8623.07 (3.41%)	43.80 (3.47%)
95%	1233.48 (7.97%)	8786.40 (1.58%)	44.65 (1.61%)
105%	1444.69 (7.79%)	9049.73 (1.37%)	46.01 (1.39%)
110%	1546.54 (15.39%)	9157.50 (2.58%)	46.57 (2.62%)

Table 5.18: Sensitivity of γ_1 - Recovery Rate of Untreated Cholera

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	1591.09 (18.71%)	9250.23 (3.62%)	52.23 (15.09%)
95%	1461.17 (9.02%)	9091.94 (1.85%)	48.64 (7.19%)
105%	1227.70 (8.40%)	8755.76 (1.92%)	42.39 (6.58%)
110%	1122.72 (16.23%)	8577.78 (3.91%)	39.65 (12.63%)

Table 5.19: Sensitivity of $h(a,t)$ - Oral Rehydration Therapy

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	1329.70 (0.79%)	8908.83 (0.20%)	85.38 (88.15%)
95%	1334.98 (0.40%)	8917.96 (0.10%)	65.48 (44.29%)
105%	1345.65 (0.40%)	8936.33 (0.10%)	25.08 (44.73%)
110%	1351.02 (0.80%)	8945.56 (0.21%)	4.58 (89.90%)

Table 5.20: Sensitivity of δ_L - Death Rate of Vibrio in the Environment.

Scenario	**Peak Infected Pop**	**Total Infected Pop**	**Total Mortalities**
90% of default	1342.35 (0.15%)	9030.59 (1.16%)	45.92 (1.20%)
95%	1341.31 (0.08%)	8977.40 (0.56%)	45.64 (0.58%)
105%	1339.29 (0.07%)	8879.63 (0.53%)	45.13 (0.55%)
110%	1338.29 (0.15%)	8834.76 (1.03%)	44.89 (1.07%)

Table 5.21: Sensitivity of η - Relative Amount of Stool Per Unit Time

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	1141.47 (14.83%)	8553.11 (4.19%)	43.44 (4.28%)
95%	1244.00 (7.18%)	8755.24 (1.93%)	44.49 (1.97%)
105%	1430.67 (6.74%)	9075.08 (1.66%)	46.15 (1.70%)
110%	1515.54 (13.08%)	9203.75 (3.10%)	46.82 (3.18%)

Table 5.22: Sensitivity of χ - Rate of Vibrio Moving from HI to non-HI State

Scenario	Peak Infected Pop	Total Infected Pop	Total Mortalities
90% of default	1519.04 (13.34%)	9151.55 (2.51%)	46.54 (2.56%)
95%	1428.03 (6.55%)	9041.60 (1.28%)	45.97 (1.30%)
105%	1255.93 (6.29%)	8807.77 (1.34%)	44.76 (1.36%)
110%	1174.95 (12.34%)	8683.10 (2.73%)	44.11 (2.79%)

5.2.5 Selected Plots of Sensitive Parameters

This section contains plots of the effects of the most sensitive parameters on model behavior.

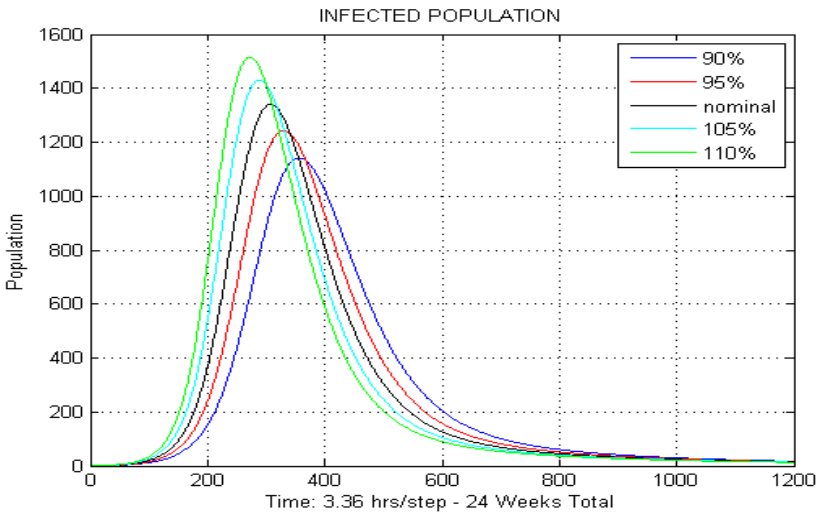

Figure 5.8: Sensitivity of ξ - Rate of Vibrio Shedding - LO rate shedding

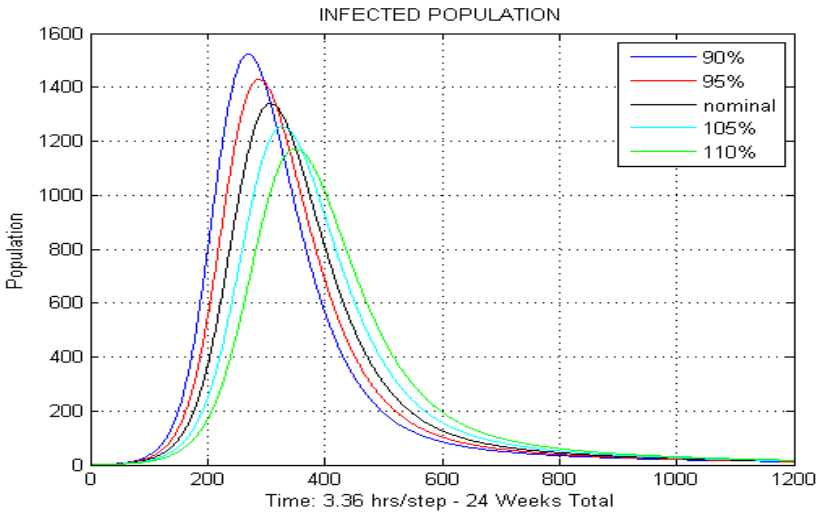

Figure 5.9: Sensitivity of $\kappa_H(a)$ - Half Saturation Const - LO Rate Shedding

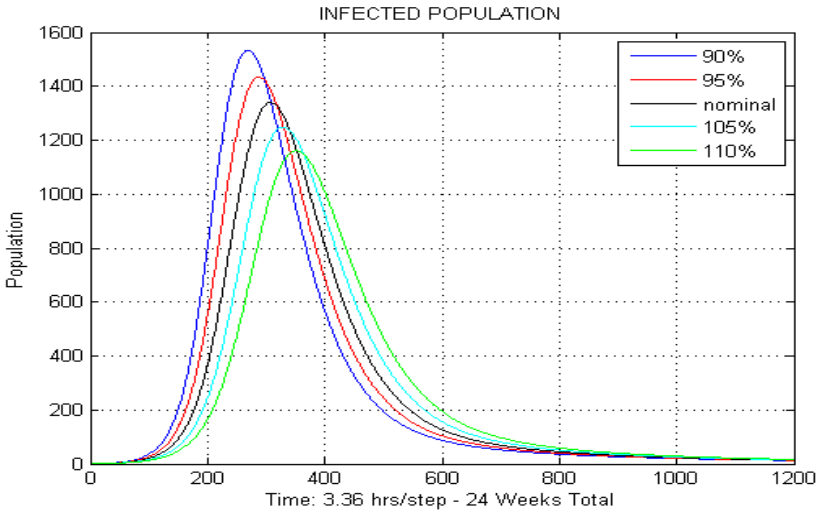

Figure 5.10: Sensitivity of $\kappa_L(a)$ - Half Saturation Const - LO Rate Shedding

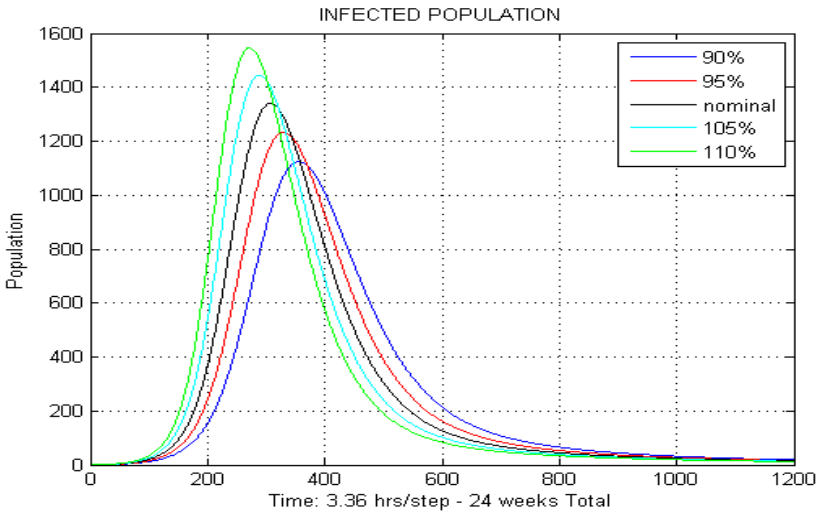

Figure 5.11: Sensitivity of $\beta_H(a)$ - Ingestion rate of HI vibrios - LO Rate Shedding

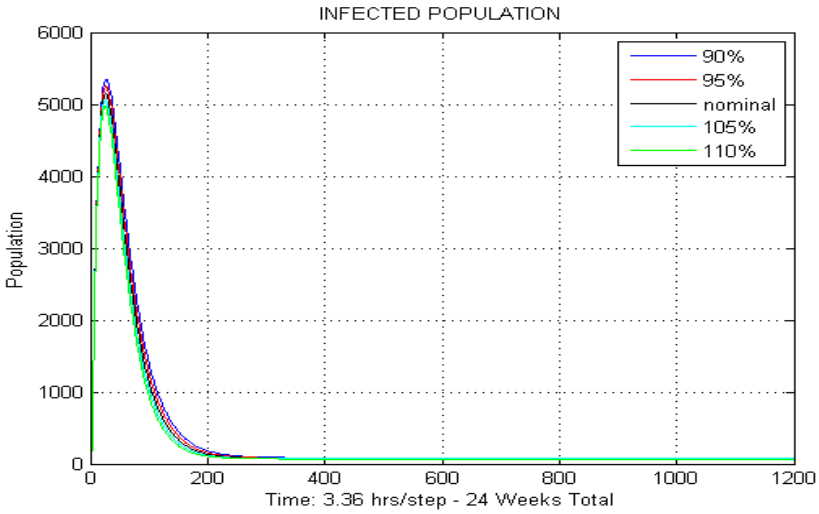

Figure 5.12: Sensitivity of γ_1 - Recovery Rate of Untreated Cholera - HI Rate Shedding

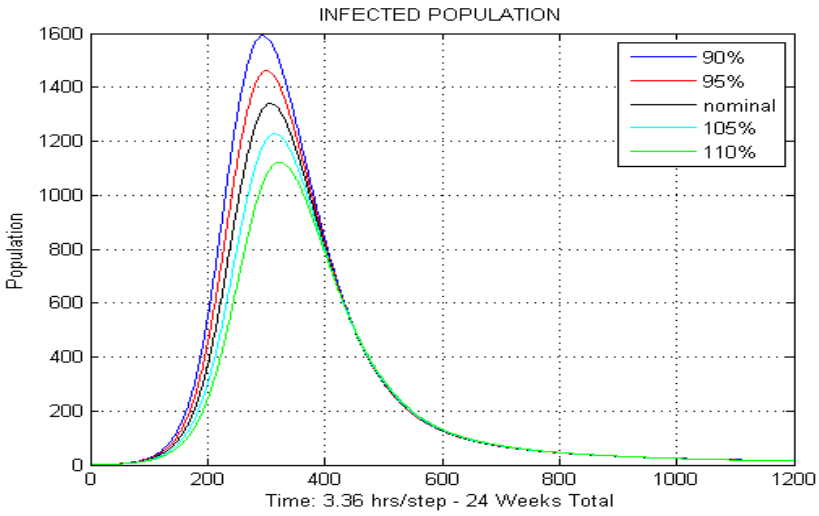

Figure 5.13: Sensitivity of γ_1 - Recovery Rate of Untreated Cholera - LO Rate Shedding

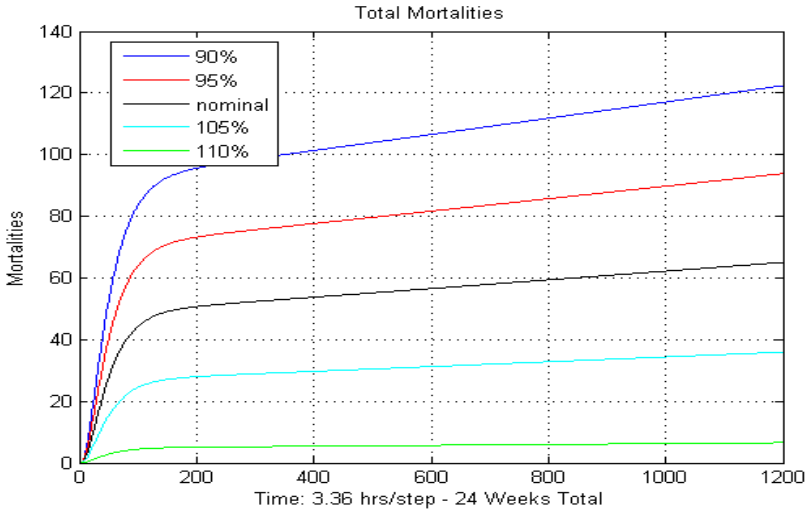

Figure 5.14: Sensitivity of $h(a,t)$ - Oral Rehydration Therapy - HI Rate Shedding

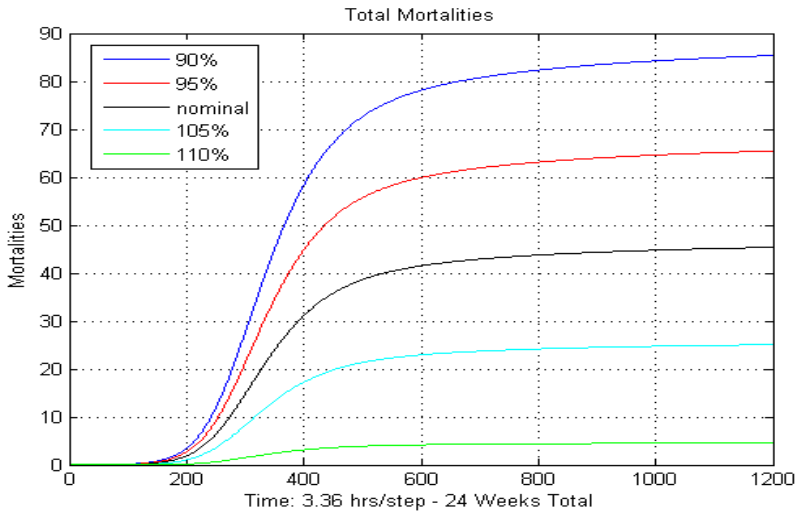

Figure 5.15: Sensitivity of $h(a,t)$ - Oral Rehydration Therapy - LO Rate Shedding

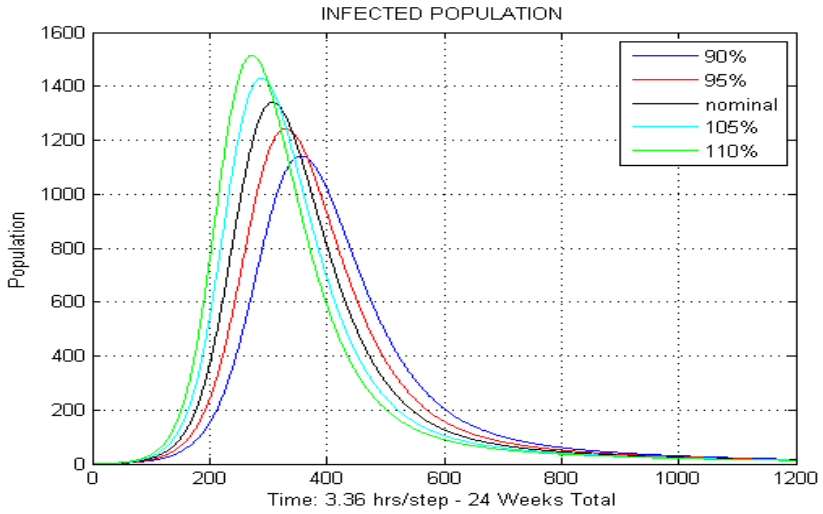

Figure 5.16: Sensitivity of η - Rel Amount of Stool Per Unit Time - LO Shedding

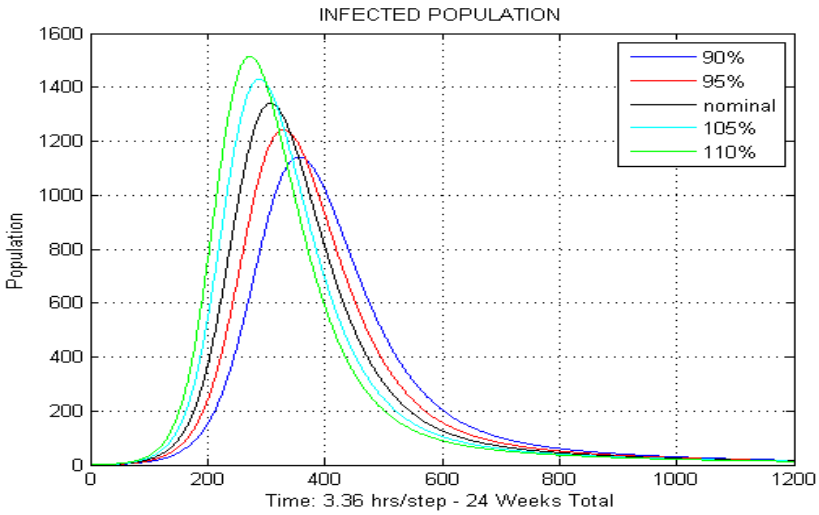

Figure 5.17: Sensitivity of χ - Rate Vibrio Moving from HI to non-HI - LO Shedding

CHAPTER 6

Comparing the Age-structured Model to Case Studies

To see how capable the age-structured model (equations 2.15-2.19) is in predicting future cholera epidemics, we compare the model output to some case studies found in literature. We choose to compare the age-structured model, solved by the first order upwind scheme, as presented in Chapter 3, with two case studies related to a recent cholera outbreak in Zimbabwe, Africa. Due to the limited data, we can only curve fit 4 parameter values, and using these parameter values, the simulation gives results for the total infection cases, and the percentage of the number of disease-related deaths that are relatively accurate. With this age-structured model, we further study the effect of cholera on children versus adults. The World Health Organization (WHO) estimates that over half of all cholera related deaths are children. Using the demographic information in Zimbabwe, our simulation result supports this.

6.1 The Zimbabwe Cholera Case Studies

Zimbabwe is a large landlocked country located in southern Africa. It is divided into eight provinces, and two major cities, spread out over an area of 400,000 square miles. The provinces are subdivided into 59 districts which are then divided further into 1,200 municipalities. The total population of the country is 12.5 million, with 40 percent living in urban centers.

In August 2008, a severe cholera outbreak occurred in Zimbabwe. The disease spread so quickly that it overwhelmed the Zimbabwean health department and the

WHO. Many people, especially those in remote areas, received no treatment, resulting in a high overall mortality rate. By the time the outbreak was over in August 2009, there were a total of 98,585 reported infection cases (and 4,287 resulting deaths), representing approximately 1 percent of the population.

The epidemic spread throughout the country in a nonuniform fashion. In some provinces it experienced explosive growth, while in others it was mild. Outbreaks did not occur simultaneously either, but the composite peak for the country occurred in January, 2009. Figure 6.1 shows a district map of the state of the epidemic nine weeks after the WHO first started recording data. Notice the heterogeneous nature of the spreading epidemic.

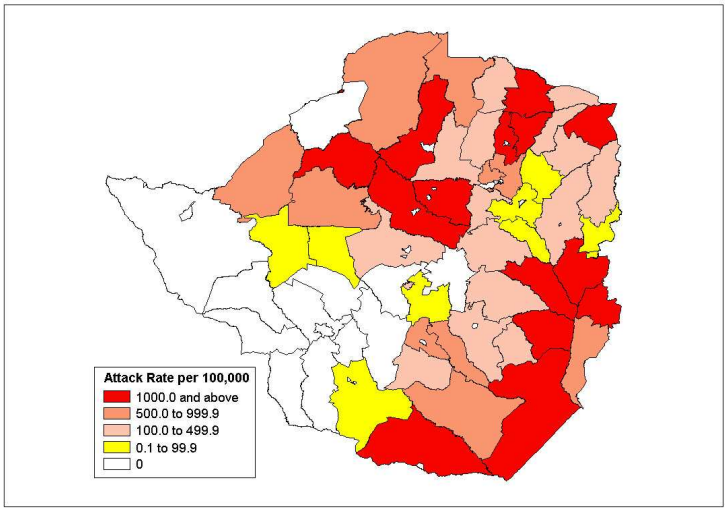

Figure 6.1: World Health Organization Data [21]
Zimbabwe Cholera Epidemic 2008 - Week 9
Cumulative Attack Rates (Cases/100,000)

Epidemic cholera is hard to model. Although much is known about the mechanisms behind its spreading behavior, we still do not fully understand what makes cholera outbreaks happen in some places and not others. Typically, during an epi-

demic, for every person reporting symptoms, up to 250 other people can get the infection but show little or no symptoms [12], and still contribute to the spread of the disease. Assuming that if the entire population of Zimbabwe (12.5 million) did get infected, a population of 98,585 symptomatic infecteds falls within the expected range.

Many factors prevented the epidemic from spreading to the entire population though. Zimbabwe is composed of over 1200 municipalities, each with its own sanitation facilities, so not everyone was equally exposed to contaminated water and food sources. During the epidemic, Zimbabwe health centers, and the WHO were both actively trying to vaccinate the population. Most ($\sim 60\%$) people live in rural areas where the population density can be below the threshold required to produce a large outbreak. Lastly, cholera is endemic to Zimbabwe, so some people had immunity.

There are two major assumptions with the SIR epidemic cholera model we use. The model only describes the spread of the disease through the population capable of showing symptoms of infection. So, for each case study, N represents, not the size of an entire population in an area, but the size of the sub-population at risk to symptomatic infection from Cholera. The model also assumes that everyone in the at-risk population is equally likely to get infected.

In the Zimbabwe cholera case studies, we adjust four model parameters ($N =$ size of population at risk to symptomatic infection, $\xi =$ shedding rate of vibrio, $\beta_L =$ Ingestion rate non-HI Vibrio, and $\beta_H =$ Ingestion rate HI Vibrio) to produce a best fit curve over the infection data. Table 6.1 contains model parameter data specific to the country of Zimbabwe, taken from reference [15]. All other parameters in the

model are the default values taken from Table 3.1.

Table 6.1: Model Parameters and Values

PARAM	DESCRIPTION	VALUE
A	Life expectancy	43.5 years old
β_L	Ingestion rate non-HI Vib - (95% conf interval)	$0.16 - 1.54$
β_H	Ingestion rate HI Vib (95% conf interval)	$0.0011 - 0.0016$

Since the epidemic is not uniform throughout the country, we focus our comparison to two of the most populous provinces: Masvingo and Midlands. From our curve fitting, we show that Midlands had a much higher shedding rate compared to Masvingo. Both cases are considered to be low rate shedding scenarios though.

6.1.1 Masvingo Case Study

Masvingo is a province located in the south-eastern section of Zimbabwe. It is one of the most populous with 1.4 million people. Masvingo contains seven districts: Masvingo city, Gutu, Chivi, Bikita, Chiredzi, Mwenezi, and Zaka. Cholera was first reported between November 13 and 17 in Masvingo city, Chiredzi and Mwenezi. Each district experienced its own distinct infection curve with different characteristics of severity and timing, and all but the Zaka province eventually reported infection cases. By the end of the outbreak a total of 11,644 infections, and 691 deaths were reported.

In this case study, we varied two major parameters (N = size of population at risk to symptomatic infection, ξ = shedding rate of vibrio) and two minor parameters (β_L = Ingestion rate non-HI Vibrios, β_H = Ingestion rate HI Vibrios) to produce a curve to fit the infection data. Table 6.2 shows the parameter values of the best fit model curve, and Figure 6.2 shows the model curve superimposed over the data. All

other parameters are either listed in Table 6.1 or have default values from Table 3.1.

Simulation results show a total of 15,275 total infection cases (Figure 6.3). The model also predicts 835 infection related deaths (Figure 6.4), which is approximately 5.4% of the total infection cases. The actual data shows 11,644 reported infections with 691 reported deaths, which is about 5.9% of the total infection cases.

Table 6.2: Masvingo - Model Parameters and Values

PARAM	DESCRIPTION	VALUE
N	Population at risk	16000 people
ξ	shedding rate of vibrio	2200 $cells/(mL * day * human)$
β_L	Ingestion rate non-HI Vib	0.85
β_H	Ingestion rate HI Vib	0.0014

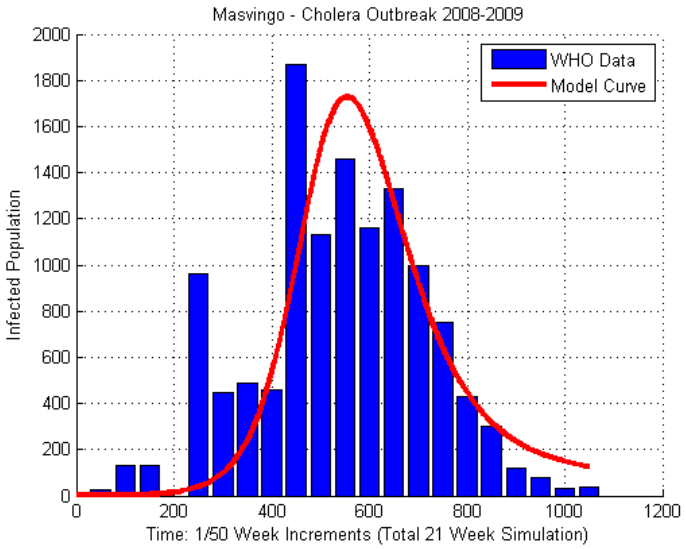

Figure 6.2: Simulation of Masvingo Zimbabwe Cholera Outbreak

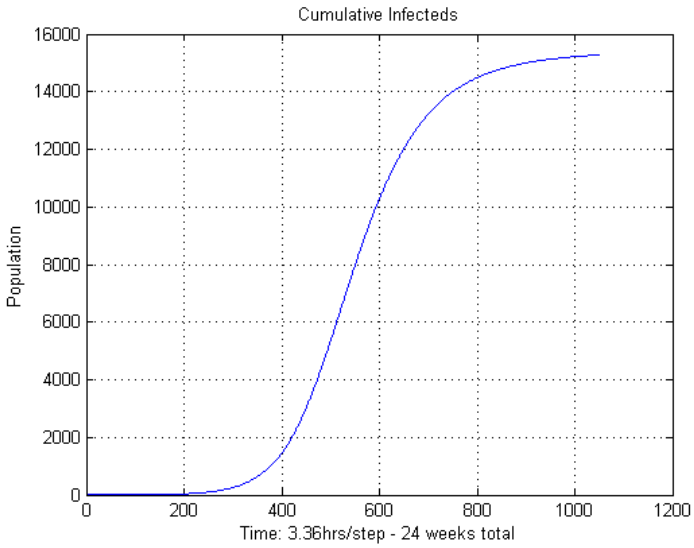

Figure 6.3: Masvingo - Cumultive Infected Population

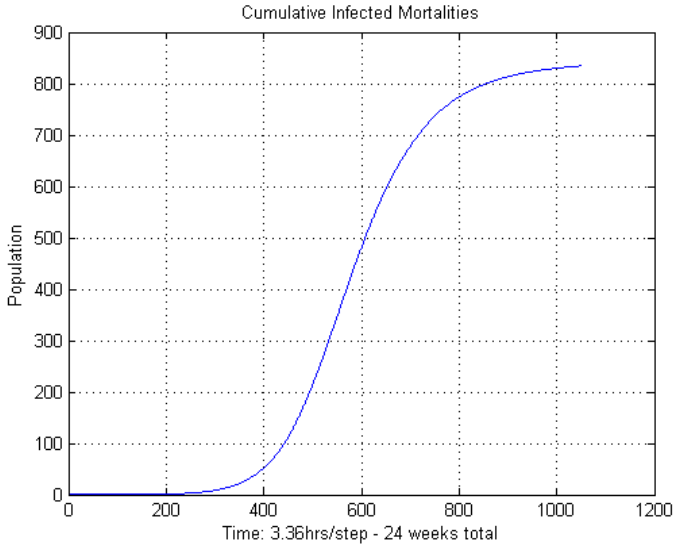

Figure 6.4: Masvingo - Cumulative Infected Mortalities

6.1.2 Midlands Case Study

Midlands is a province located in the midsection of Zimbabwe. Like the Masvingo province, it is also one of the most populous with 1.5 million people. Midlands contains seven districts: Chirumhanzu, Gokwe, Gweru, Kwekwe, Mberengwa, Shurugwi,

and Zvishavane. Cholera was first reported between November 11 and 19. Each district experienced its own distinct infection curve with different characteristics of severity and timing, with Gokwe being hit the hardest. By the end of the outbreak a total of 7,156 infections and 331 deaths were reported.

In this case study, we varied two major parameters (N = size of population at risk to symptomatic infection, ξ = shedding rate of vibrio) and two minor parameters (β_L = Ingestion rate non-HI Vibrios, β_H = Ingestion rate HI Vibrios) to produce a best fit curve over the infection data. Table 6.3 shows the values of the best fit model curve, and Figure 6.5 shows that curve superimposed over the data. All other parameters are either listed in Table 6.1 or have default values from Table 3.1. Note that the shedding rate ξ of vibrio in Midlands is much higher compared to the one in Masvingo.

Simulation results show a total of 10,031 total infection cases (Figure 6.6) and 548 infection related deaths (Figure 6.7), which is 5.4% of total infection cases. In comparison, the actual data shows that there were 7,156 reported infections with 331 reported deaths, which is approximately 4.6% of total infection cases.

Table 6.3: Midlands - Model Parameters and Values

PARAM	DESCRIPTION	VALUE
N	Population at risk	10500 people
ξ	shedding rate of vibrio	18000 $cells/(ml * day * human)$
β_L	Ingestion rate non-HI Vib	0.43
β_H	Ingestion rate HI Vib	0.0013

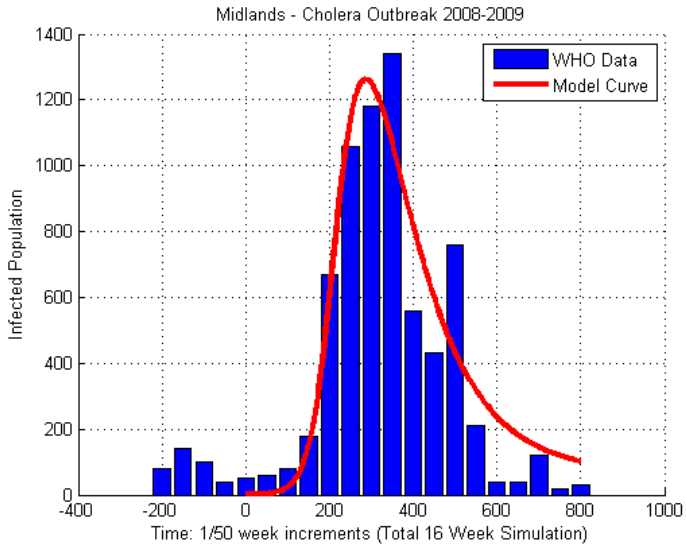

Figure 6.5: Simulation of Midlands Zimbabwe Cholera Outbreak

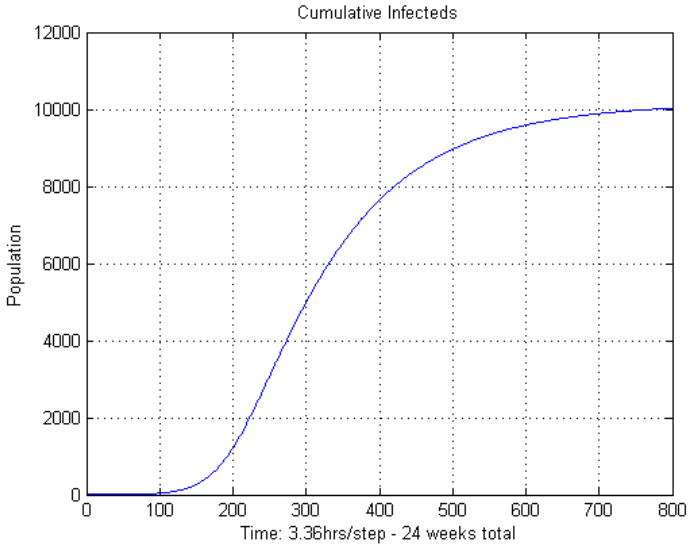

Figure 6.6: Midlands - Cumultive Infected Population

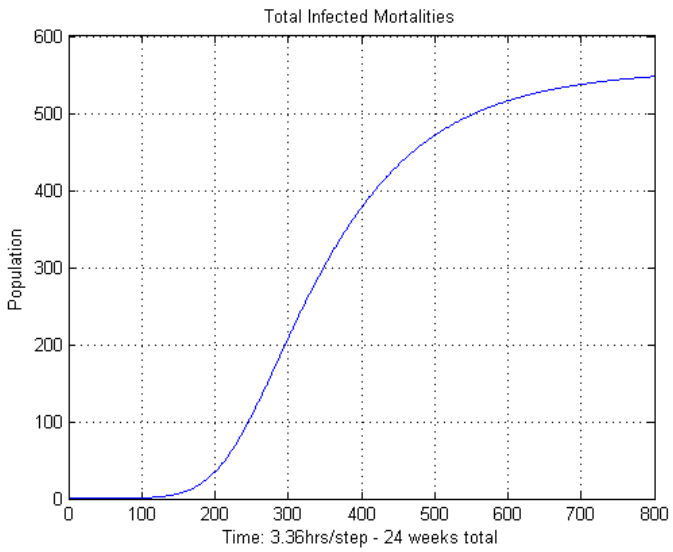

Figure 6.7: Midlands - Cumulative Infected Mortalities

6.2 The Effects of Cholera on Children Versus Adults

Cholera attacks children more severely than it does adults [1]. Children typically have poorer hygiene and are therefore at higher risk to contracting the disease. Once infected, children also suffer more severe symptoms and higher death rates. Most adults in cholera-endemic areas have some antibodies, which helps to protect them from developing the disease, while children in these areas are commonly malnourished so their immune systems are weak. Children are also small, and their bodies contain less fluid, which means they are more likely to die from the disease because they become dehydrated faster than adults. The World Health Organization estimates that over half of all cholera related deaths are children.

Our first order model was modified to allow us to study the effects of cholera on children and adults. A high rate of shedding scenario was tested on a population of 10,000 people for a 24 week period. The population was distributed according to

the following 2005 demographics in Zimbabwe: 40.1% aged 0 to 14, 56.1% aged 15 to 64, and 3.8% aged over 65 [18]. This translates to a population of 2578 children aged 1 though 10, and 5990 people aged 11 years or older (considered adults in this case). Infected elderly people are also more severely affected than adults because they have weaker immune systems, however, they are classified as adults in our model and will not be analyzed separately. Refer to Appendix F for the Matlab program for this model.

Simulation results show that although approximately the same percentage of children and adults are infected, approximately 330 children die, while only 250 adults die due to infection. This means about 14% of infected children die, versus 3% of adults. Model output can be viewed in Figures 6.8 through 6.10.

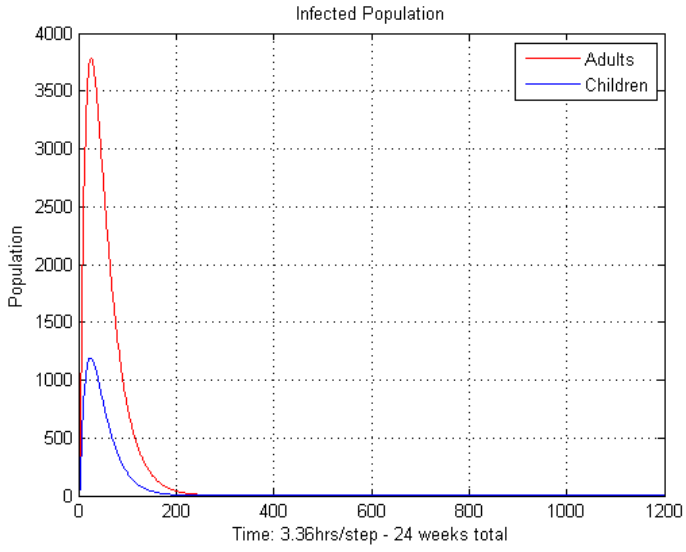

Figure 6.8: Infected Population - Children Versus Adults

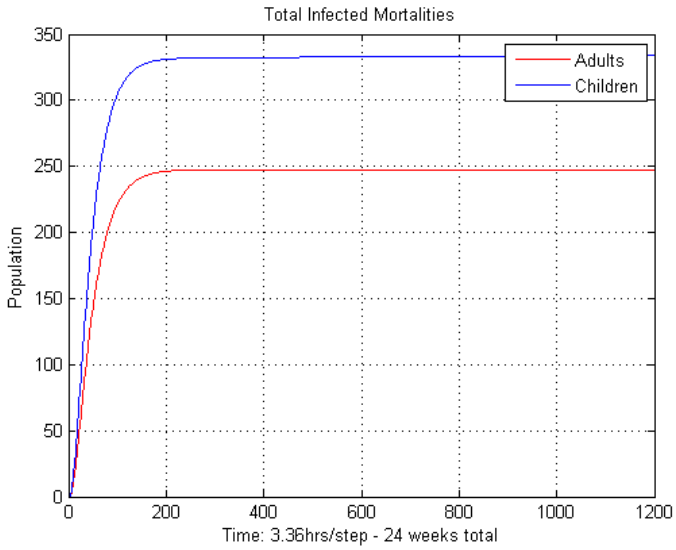

Figure 6.9: Total Infected Mortalities - Children Versus Adults

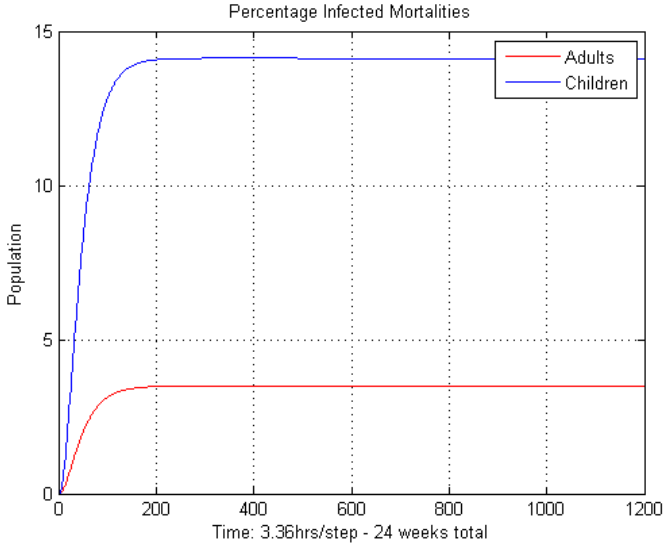

Figure 6.10: Percentage Infected Mortalities - Children Versus Adults

REFERENCES

[1] M. Ali, A.L. Lopez, *The global burden of cholera*, Bulletin of the World Health Organization 2012;90:209-218A. doi: 10.2471/BLT.11.093427, http://www.who.int/bulletin/volumes/90/3/11-093427/en/

[2] . Brauer, *Mathematical Epidemiology*, Lecture Notes in Mathematics, Springer, vol **1945**, (2008)

[3] F. Brauer, *Mathematical Models in Population Biology and Epidemiology*, NY: Springer, first edition (2001).

[4] F. Britton, *Essential Mathematical Biology*, NY: Springer,(2003).

[5] R. Burden, J. Faires *Numerical Analysis*, Brooks Cole,eigth edition (2005).

[6] A. Camilli 2002. *Host-induced epidemic spread of the cholera bacterium.* Nature (2002) 417:642645.

[7] C.T. Codeço, *Endemic and epidemic dynamics of cholera: the role of the aquatic reservoir*, BMC Infect Dis. 1 (1) (2001).

[8] R.I. Glass, *Endemic Cholera in Rural Bangladesh, 1966 to 1980*, American Journal Epidemiology (1982) 116 (6): 959-970.

[9] M.K. Gobbert, A. Alexanderian, K.R. Fister, *An Age-Structured Model for the Spread of Epidemic Cholera: Analysis and Simulation*, Technical Report, Department of Mathematics and Statistics, University of Maryland, Baltimore County, (2011).

[10] D.M. Hartley, J.G. Morris, and D.L. Smith, *Hyperinfectivity: A critical element in the ability of v. cholerae to cause epidemics?*, PLoS Med., 3 (2006).

[11] W.O. Kermack, A.G. McKendrick, *A Contribution to the Mathematical Theory of Epidemics*, Proc. Roy. Soc. Lond. A 115, 700-721, (1927).

[12] A. King, E. Ionides, M. Pascual, M. Bouma, *Inapparent Infections and Cholera Dynamics*, Nature 454, 877-880 (14 August 2008). http://www.nature.com/nature/journal/v454/n7206/full/nature07084.html
also: A New View of Why Cholera Won't Go Away, http://news.sciencemag.org/sciencenow/2008/08/13-02.html

[13] M.M. Levine, *Duration of Infection-Derived Immunity to Cholera*, Journal of Infectious Diseases (1981) 143(6): 818–820

[14] D. S. Merrell, and S. M. Butler, *Host Induced Epidemic Spread of the Cholera Bacterium*, Nature 417 (2002), 642–645

[15] Z. Mukandavirea, *Estimating the reproductive numbers for the 2008 2009 cholera outbreaks in Zimbabwe.*, http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3102413/
The mathematical model and supporting information can be found at http://www.pnas.org/content/suppl/2011/04/21/
/1019712108.DCSupplemental/pnas.201019712SI.pdf

[16] G. D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, Third Editon, Oxford University Press, Applied Mathematics and Computing Series, 1985.

[17] J. Strikwerda, *Finite difference schemes and partial differential equations*, 2nd editon, SIAM, 2004.

[18] *United Nations Estimate of Zimbabwe Population*, CIA Factbook 2007, CIA Publications, Washington D.C.

[19] *Mortality Rates of Countries of the World*, CIA Factbook 2007, CIA Publications, Washington D.C., Sourced through: http://www.indexmundi.com

[20] *The World Health Organization - Cholera Fact sheet.*, http://www.who.int/mediacentre/factsheets/fs107/en/

[21] Ministry of Health and Child Welfare Rapid Response Disease Notification System, World Health Organization, *Cholera in Zimbabwe: Epidemiological Bulletin Number 12, Week 9 (22-28 February 2009)*, *http* : $//www.who.int/hac/crises/zwe/sitreps/$ $/zimbabwe_cholera_bulletin_1 2_2 2_8 feb 2009.pdf$

[22] W.E. Woodward, *Cholera Reinfection in Man*, Journal of Infectious Diseases (1971) Vol 23, No 1

APPENDIX A

Matlab Code for the First Order Accurate Age-Structured Cholera Model

```
function [] = SIR_AgeStructured()
%%  Project - Age Structured Cholera Model
%  Author: John Szijjarto
%  Date:   10/27/11
%  Title: Implementation of Gobbert's Age-Structured Model
%
%  delta-t = 1 week
%%

clear all;  % Close/Delete all figures
close all;  % Free system memory
clc;        % Clear command window

% Simulation Control Variables
reference_sim = false;      % i f true, run ref sim (ie. no infected pop)

xi            = 1000000000;  % Hi Rate of shedding of cholera vibrios from
%xi            = 100;         % Lo Rate of shedding of cholera vibrios from
                             % infected human of age a.

k = 1/50;                    % detla_t = 1/50 of a week
h = 1;                       % delta_a = 1 week
alpha = 1;                   % Proportionality factor (wave speed)
b = (1/50)*(1/52);           % normal mortality rate in deaths per week


A        = 72;           % Upper bound on human age
BL       = 0;            % non-HI Vibrio Population
BH       = 0;            % HI Vibrio Population
kappa_L  = 1000000;      % cells/ml
kappa_H  = kappa_L/700;  % cells/ml
beta_L   = 1.5/7*7;      % per week
beta_H   = 1.5/7*7;      % per week
lambda   = 0*7;    % Human recruitment rate (non-newborns entering pop)
gamma_1  = 1/5*7;  % recovery rate of untreated cholera
gamma_2  = 1/3*7;  % recovery rate of treated cholera
u        = 0*7;    % antiboitic treatmnt rate for humans of age a at time t
H2O      = 0.9;    % hydration therapy related mortality. (h(a,t) in model)
                   % This is a percentage vale.
delta_L  = 1/30*7; % Death rate of vibrio in the environment.

eta      = 0.1;    % Relative amount of stool per unit time - no units

chi      = 1/5*24*7; % Rate of vibrio moving from HI to non-HI state.

% Declare the S, I, R Arrays and zero out all values.
%
m = 3744;         % cols -> total age = 72 * 52 = 3744 weeks
n = 24/k;         % rows -> total simulation time = 24 weeks
S = zeros(m,n);   % Suseptable population
I = zeros(m,n);   % Infected population
R = zeros(m,n);   % Recovered population


%% Boundary Conditions
%       Note: Boundary conditions fo r R array are created
%             within the main simulation loop.
%
```

```
for t = 0:(n-1)
    % Susceptable and Infected Population
    S(0+1,t+1) = 0.0;
    I(0+1,t+1) = 0.0;
end


%% Initial Conditions
%
% Initial Conditions fo r R and S
one_year_old = 52;   % Age in weeks
for a=(m-1):-1:0
    % Susceptable and Recovered Population
    if (a<=one_year_old) % if age < 1 yr old then immune (ie. in R group)
        R(a+1,0+1)  = 2.67;
        S(a+1,0+1) = 0;
    else
        R(a+1,0+1)  = 0;
        S(a+1,0+1) = 2.67; % 2.67 *52wks *72yrs = 10000 people
    end
end

% Initial Conditions for I
%
eighteen_years_old = 936;   % Age in weeks
nineteen_years_old = 988;   % Age in weeks
if (reference_sim == false) % reference sim contains no infected people
    % Include one 18 year old infected human
    for a=(m-1):-1:0
        if (a>=eighteen_years_old) && (a<=nineteen_years_old)
            I(a+1,0+1) = 1/52;
        else
            I(a+1,0+1) = 0;
        end
    end
end

total_BH = zeros(n,1);
total_BL = zeros(n,1);


%% Run the Simulation
%
% Generate All Other Interior Grid Points
for t = 0:(n-1)

    % Suseptable Population - Generate one row
     for a=1:(m-1)
        S(a+1,t+1+1) = ...
          (1-k*alpha/h)* S(a+1,t+1) ...
          + k*alpha/h* S(a-1+1,t+1) ...
          + k*lambda ...                              % recruitment
          - k*beta_L*BL/(kappa_L+BL)*S(a+1,t+1)... % BL infected humans
          - k*beta_H*BH/(kappa_H+BH)*S(a+1,t+1)... % BH infected humans
          - k*b * S(a+1,t+1) ...                      % natural mortalities
          + k*omega(a+1)*7 * R(a+1,t+1);           % pop losing immunity
     end

    % Infected Population - Generate one row
     for a=1:(m-1)
        I(a+1,t+1+1) = ...
          (1-k*alpha/h)* I(a+1,t+1) ...
          + k*alpha/h* I(a-1+1,t+1) ...
          + k*beta_L*BL/(kappa_L+BL)*S(a+1,t+1)... % BL infected humans
          + k*beta_H*BH/(kappa_H+BH)*S(a+1,t+1)... % BH infected humans
          - k*b * I(a+1,t+1) ...                      % natural mortalities
          - k*(1-H2O)*delta(a)*7*I(a+1,t+1) ...    % infected mortalities
          - k*gamma_1*(1-u)*I(a+1,t+1) ... % pop recovering w/o antibiotics
```

```
127             - k*gamma_2*u*I(a+1,t+1);        % pop recovering w antibiotics
128       end
129
130       % Include new born babies into the recovered population this week
131       fecundity = 0;
132       for age=779:1:2339
133           fecundity = fecundity...
134               + (S(age+1,t+1) + I(age+1,t+1) + R(age+1,t+1))...
135               * (1/5) * (sin((age-780)/1560*3.14159))^2;
136       end
137       R(0+1,t+1)= fecundity/52; %Divided by 52 weeks/year
138
139
140       % Recovered Population - Generate one row
141       for a=1:(m-1)
142          R(a+1,t+1+1) = ...
143            (1-k*alpha/h)* R(a+1,t+1) ...
144            + k*alpha/h * R(a-1+1,t+1) ...
145            + k*gamma_1*(1-u)*I(a+1,t+1) ... % pop recovering w/o antibiotics
146            + k*gamma_2*u*I(a+1,t+1) ...     % pop recovering with antibiotics
147            - k*b * R(a+1,t+1) ...           % natural mortalities
148            - k*omega(a+1)*7 * R(a+1,t+1);   % recovered - loosing immunity
149       end
150
151       % Calculate current hyperinfective (BH) and non-hyperinfective (BL)
152       % cholera bacteria populations
153       dBH = get_dBH(I,t,BH,xi,chi,eta,A,k);
154       dBL = get_dBL(chi,BH,delta_L,BL,k);
155       BH = BH + dBH;
156       BL = BL + dBL;
157       total_BH(t+1) = BH;
158       total_BL(t+1) = BL;
159
160       fprintf('%f %f\n',BH,BL);
161
162   end
163
164   % Print Population Totals
165   total_sus_population = zeros(n,1); % column vector of length n
166   total_inf_population = zeros(n,1); % column vector of length n
167   total_rec_population = zeros(n,1); % column vector of length n
168   total_population = zeros(n,1);     % column vector of length n
169
170   fprintf('\nSUSEPTABLE  INFECTED RECOVERED  TOTAL-POPULATION\n\n');
171
172   for t = 0:1:(n-1)
173       for a=0:(m-1)
174           total_sus_population(t+1) = total_sus_population(t+1) ...
175                                       + S(a+1,t+1);
176           total_inf_population(t+1) = total_inf_population(t+1) ...
177                                       + I(a+1,t+1);
178           total_rec_population(t+1) = total_rec_population(t+1) ...
179                                       + R(a+1,t+1);
180           total_population(t+1)     = total_sus_population(t+1) ...
181                                       + total_inf_population(t+1)...
182                                       + total_rec_population(t+1);
183       end
184       fprintf('%f %f %f %f\n',...
185           total_sus_population(t+1),...
186           total_inf_population(t+1),...
187           total_rec_population(t+1), ...
188           total_sus_population(t+1) + total_rec_population(t+1));
189   end
190
191   % Produce 2D Plots
192   %
193
194   x=1:1:n;
```

```
195
196 figure;
197 plot(x,total_sus_population(x));
198 title('Suseptable Population');
199 xlabel('Time: 3.36hrs/step - 24 weeks total');
200 ylabel('Population');
201 grid;
202
203 figure;
204 plot(x,total_inf_population(x));
205 axis([0 1200 0 10000]);
206 title('Infected Population');
207 xlabel('Time: 3.36hrs/step - 24 weeks total');
208 ylabel('Population');
209 grid;
210
211 figure;
212 plot(x,total_rec_population(x));
213 axis([0 1200 0 10000]);
214 title('Recovered Population');
215 xlabel('Time: 3.36hrs/step - 24 weeks total');
216 ylabel('Population');
217 grid;
218
219 figure;
220 plot(x,total_population(x));
221 title('Total Population');
222 xlabel('Time: 3.36hrs/step - 24 weeks total');
223 ylabel('Population');
224 grid;
225
226 figure;
227 plot(x,total_BH(x));
228 title('BH Vibrio Population');
229 xlabel('Time: 3.36hrs/step - 24 weeks total');
230 ylabel('Population');
231 grid;
232
233 figure;
234 plot(x,total_BL(x));
235 title('BL Vibrio Population');
236 xlabel('Time: 3.36hrs/step - 24 weeks total');
237 ylabel('Population');
238 grid;
```

APPENDIX B

Matlab - 2nd Order Accurate Age-Structured Cholera Model - Jacobian Matrix

```
function [] = SIR_AgeStructured_2nd_order_Matrix_v10()
%%  Project - Age Structured Cholera Model
%  Author: John Szijjarto
%  Date:   10/27/12
%  Title:  2nd Order Accurate Epidemic Cholera Model - Matrix Method
%
%
%  Running the Program:
%       Reference Simulation - Variable Settings
%           reference_sim = true;
%       Hi Rate Shedding Simulation - Variable Settings
%           reference_sim = false;
%           xi           = 1000000000;
%       Low Rate Shedding Simulation - Variable Settings
%           reference_sim = false;
%           xi           = 100;
%

%% %%%%%% Matlab Environment Commands  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
clear all;  % Close/Delete all figures
close all;  % Free system memory
clc;        % Clear command window

outfile_1 = fopen('output.txt','w');


%% %%%%%% Simulation Control Variables  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

reference_sim = false;      % if true, run ref sim (ie. no infected pop)

%xi           = 1000000000; % Hi Rate of shedding of cholera vibrios from
xi            = 100;        % Lo Rate of shedding of cholera vibrios from
                            % infected human of age a.

%% %%%%%% Model Variables %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
k     = 1/50;           % detla_t = 1/50 of a week
h     = 1;              % delta_a = 1 week
alpha = 1;              % Proportionality factor (wave speed)
b     = (1/50)*(1/52);  % normal mortality rate in deaths per week

% Declare the S, I, R Arrays and zero out all values.
%
% need to double the age parameter to handle RHS boundary
m = 3744*2;               % cols -> total age = 72 * 52 = 3744 weeks * 2
n = 24/k;                % rows -> total time = 24 weeks

S = zeros(m,n);       % Suseptable population
I = zeros(m,n);       % Infected population
R = zeros(m,n);       % Recovered population

BH  = zeros(n,1);       % Concentration of HI vibrio at time t
BL  = zeros(n,1);       % Concentration of non-HI vibrio at time t

A       = 72;           % Upper bound on human age

```

```
kappa_L  = 1000000;      % Saturation Constant of non-Hi vibrios at age a
kappa_H  = kappa_L/700; % Saturation Constant of Hi vibrios at age a
beta_L   = 1.5/7*7;      % Ingestion rate of non-Hi vibrios at age a
beta_H   = 1.5/7*7;      % Ingestion rate of Hi vibrios at age a
lambda   = 0*7;          % Human recruitment rate (non-newborns entering pop)
gamma_1  = 1/5*7;        % recovery rate of untreated cholera
gamma_2  = 1/3*7;        % recovery rate of treated cholera
u        = 0*7;          % antiboitic treatment rate - humans age a at time t
H2O      = 0.9;          % hydration therapy related mortality.(h(a,t) in eq)
                         % This is a percentage value.
delta_L  = 1/30*7;       % Death rate of vibrio in the environment.

eta      = 0.1;          % Relative amount of stool per unit time - no units

chi      = 1/5*24*7;     % Rate of vibrio moving from HI to non-HI state.


%% %%%%% Initial Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
eighteen_years_old = 936;   % Age in weeks
nineteen_years_old = 988;   % Age in weeks

% Initial Conditions for I (Infected)
%
if (reference_sim == false) % reference sim contains no infected people
    % Include one 18 year old infected human
    for a=(m-1):-1:0
        if (a>=eighteen_years_old) && (a<nineteen_years_old)
            I(a+1,0+1) = 1/52.0;
        else
            I(a+1,0+1) = 0.0;
        end
    end
end

% Initial Conditions for R (Recovered) and S (Susceptible)
one_year_old = 52;   % Age in weeks
for a=(m-1):-1:0
    % Susceptable and Recovered Population
    % 2.67094017 * 52wks * 72yrs = 10000 people
    %
    if (a<=one_year_old) % if age<1 year-old then immune (ie. in R group)
        R(a+1,0+1) = 2.67094017;
        S(a+1,0+1) = 0.0;
    else
        R(a+1,0+1)  = 0.0;
        S(a+1,0+1) = 2.67094017;
        if (reference_sim == false)  % Hi or Low rate shedding scenario
            if (a>=eighteen_years_old) && (a<nineteen_years_old)
                S(a+1,0+1) = 2.67094017 - 1/52.0; %minus 1 infected person
            end
        end
    end
end




%% %%%%% Boundary Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%       Note: Boundary conditions for R (Recovered) array are created
%             within the main simulation loop.
%
for t = 0:(n-1)
    % Susceptable and Infected Population
    S(0+1,t+1) = 0.0;
    I(0+1,t+1) = 0.0;
end

```

```
127
128 %% %%%%% Run the Simulation %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
129 %
130 % Generate All Other Interior Grid Points
131 %
132 for t = 0:(n-1)
133
134     % Calculate current hyperinfective (BH) and non-hyperinfective (BL)
135     % cholera bacteria populations
136     dBH = get_dBH(I,t,BH,xi,chi,eta,A,k,h);
137     BH(t+1+1) = BH(t+1) + dBH;
138     dBL = get_dBL(chi,BH,delta_L,BL,t,k,h);
139     BL(t+1+1) = BL(t+1) + dBL;
140
141     % Include new born babies into the recovered population this week
142     fecundity = 0;
143     for age=779:1:2339
144         fecundity = fecundity...
145             + (S(age+1,t+1) + I(age+1,t+1) + R(age+1,t+1))...
146             * (1/5) * (sin((age-780)/1560*3.14159))^2;
147     end
148     R(0+1,t+1)= fecundity/52; %Divided by 52 weeks/year (~8 born per week)
149
150     % Main Loop
151     vibrio = (beta_L*BL(t+1+1)/(kappa_L+BL(t+1+1)) ...
152                 + beta_H*BH(t+1+1)/(kappa_H+BH(t+1+1)));
153     recovery = gamma_1*(1-u)+gamma_2*u;
154     for a=1:m-2
155
156         c1 = 1/k + 1/2*(vibrio + b);
157         c2 = -1/2*omega(a+1);
158         c3 = 1/k + 1/2*(b + (1-H2O)*delta(a+1) + recovery);
159         c4 = -1/2*vibrio;
160         c5 = 1/k + 1/2*(b + omega(a+1));
161         c6 = -1/2*recovery;
162
163         d1 =      1/k*S(a+1,t+1) + 1/2*lambda...
164                 - (alpha/(2*h))*(S(a+1+1,t+1) - S(a-1+1,t+1))...
165                 + (alpha^2*k/(2*h^2))*(S(a+1+1,t+1)...
166                   -2*S(a+1,t+1)+ S(a-1+1,t+1))...
167                 + 1/2*lambda...
168                 - 1/2*(vibrio + b)* S(a+1,t+1)...
169                 + 1/2*omega(a+1)* R(a+1,t+1)...
170                 - (k*alpha/(4*h))*lambda...
171                 + (k*alpha/(4*h))*(vibrio + b) * S(a+1+1,t+1)...
172                 - (k*alpha/(4*h))*omega(a+1+1)*R(a+1+1,t+1)...
173                 + (k*alpha/(4*h))*lambda...
174                 - (k*alpha/(4*h))*(vibrio + b)* S(a-1+1,t+1)...
175                 + (k*alpha/(4*h))*omega(a-1+1)*R(a-1+1,t+1);
176
177         d2 =      1/k*I(a+1,t+1)...
178                 - (alpha/(2*h))*(I(a+1+1,t+1)-I(a-1+1,t+1))...
179                 + (k*alpha^2/(2*h^2))*(I(a+1+1,t+1) - 2*I(a+1,t+1) ...
180                   + I(a-1+1,t+1))...
181                 + 1/2*vibrio*S(a+1,t+1)...
182                 - 1/2*(b+(1-H2O)*delta(a+1) + recovery)*I(a+1,t+1)...
183                 - (alpha*k/(4*h))*vibrio*S(a+1+1,t+1)...
184                 + (alpha*k/(4*h))*(b + (1-H2O)*delta(a+1+1) ...
185                   + recovery)*I(a+1+1,t+1)...
186                 + (alpha*k/(4*h))* vibrio*S(a-1+1,t+1)...
187                 - (alpha*k/(4*h))* (b + (1-H2O)*delta(a-1+1) ...
188                   + recovery)*I(a-1+1,t+1);
189
190         d3 =      1/k*R(a+1,t+1)...
191                 - (alpha/(2*h))*(R(a+1+1,t+1)-R(a-1+1,t+1))...
192                 + (k*alpha^2/(2*h^2))*(R(a+1+1,t+1)...
193                   -2*R(a+1,t+1)+R(a-1+1,t+1))...
194                 + 1/2*recovery*I(a+1,t+1)...
```

```
                - 1/2*(b+ omega(a+1))*R(a+1,t+1)...
                - (alpha*k/(4*h))* recovery*I(a+1+1,t+1)...
                + (alpha*k/(4*h))* (b+ omega(a+1+1))*R(a+1+1,t+1)...
                + (alpha*k/(4*h))* recovery * I(a-1+1,t+1)...
                - (alpha*k/(4*h))* (b + omega(a-1+1))* R(a-1+1,t+1);

        A = [c1   0  c2;
             c4  c3   0;
              0  c6  c5];

        B = [d1;
             d2;
             d3];

        X = A\B;

        %x=[S(i,j+1);  I(i,j+1);  R(i,j+1)]
        S(a+1,t+1+1) = X(1);
        I(a+1,t+1+1) = X(2);
        R(a+1,t+1+1) = X(3);

    end;

%    total_I_mortalities(t+1+1) = total_I_mortalities(t+1) + I_mortalities;
    if (mod(t,100) == 0)
        fprintf('t = %d\n', t);   % This is just a program status output
    end
end;



%% %%%%% Output Results %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

% %%%%% Print Population Totals %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
total_sus_population = zeros(n,1); % column vector of length n
total_inf_population = zeros(n,1); % column vector of length n
total_rec_population = zeros(n,1); % column vector of length n
total_population = zeros(n,1);     % column vector of length n

fprintf(outfile_1,...
    '\nSUSEPTABLE  INFECTED RECOVERED  TOTAL-POPULATION  BIRTHS\r\n\r\n');

for t=0:(n-1)
    for a=0:(m/2-1)
        total_sus_population(t+1) = total_sus_population(t+1) ...
                                  + S(a+1,t+1);
        total_inf_population(t+1) = total_inf_population(t+1) ...
                                  + I(a+1,t+1);
        total_rec_population(t+1) = total_rec_population(t+1) ...
                                  + R(a+1,t+1);
    end
    total_population(t+1)         = total_sus_population(t+1) ...
                                  + total_inf_population(t+1)...
                                  + total_rec_population(t+1);

    fprintf(outfile_1,'%f %f %f %f %f\r\n',...
        total_sus_population(t+1),...
        total_inf_population(t+1),...
        total_rec_population(t+1), ...
        total_population(t+1),...
        R(0+1,t+1));  % Print Births for each week
end
fclose(outfile_1);


% Produce 2D Plots
%
```

```
263
264 x=1:1:n-1;
265
266 figure;
267 plot(x,total_sus_population(x),'b',x,total_inf_population(x),'r',...
268     x,total_rec_population(x),'g');
269 title('Lo Rate Shedding - Human Populations');
270 xlabel('Time: 3.36 hrs/step');
271 ylabel('Population');
272 legend('Susceptible', 'Infected', 'Recovered');
273 grid;
274
275 figure;
276 plot(x,BH(x),'b',x,BL(x),'r');
277 title('Lo Rate Shedding - Vibrio Populations');
278 xlabel('Time: 3.36 hrs/step');
279 ylabel('Vibrio Population');
280 legend('BH Vibrio', 'BL Vibrio');
281 grid;
282
283 figure;
284 plot(x,total_sus_population(x));
285 title('Susceptible Population');
286 xlabel('Time: 3.36hrs/step - 24 weeks total');
287 ylabel('Population');
288 grid;
289
290 figure;
291 plot(x,total_inf_population(x));
292 title('Infected Population');
293 xlabel('Time: 3.36hrs/step - 24 weeks total');
294 ylabel('Population');
295 grid;
296
297 figure;
298 plot(x,total_rec_population(x));
299 title('Recovered Population');
300 xlabel('Time: 3.36hrs/step - 24 weeks total');
301 ylabel('Population');
302 grid;
303
304 figure;
305 plot(x,total_population(x));
306 title('Total Population');
307 xlabel('Time: 3.36hrs/step - 24 weeks total');
308 ylabel('Population');
309 grid;
310
311 figure;
312 plot(x,BH(x));
313 title('BH Vibrio Population');
314 xlabel('Time: 3.36hrs/step - 24 weeks total');
315 ylabel('Population');
316 grid;
317
318 figure;
319 plot(x,BL(x));
320 title('BL Vibrio Population');
321 xlabel('Time: 3.36hrs/step - 24 weeks total');
322 ylabel('Population');
323 grid;
324
325 end % Main Program
326
327 %% %%%%% Sub Functions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
328 % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
329 %
330 function db_h = get_dBH(I_,t_,BH_,xi_,chi_,eta_,A_,k_,h_)
```

```
% Return change in HI vibrio population
    sum = 0;
    for age = 0:(A_*52-2)  % age is in weeks
        sum = sum + I_(age+1+1,t_+1) + I_(age+1,t_+1);
    end;
    sum = xi_*eta_*h_*(1/2)*sum;
    dt  = k_;
    db_h = dt*h_*(sum - chi_*BH_(t_+1));
end

function db_l = get_dBL(chi_,BH_,delta_L_,BL_,t_,k_,h_)
% Return change in non-HI vibrio population
%   second order accuracy midpoint scheme used here.
    k1 = chi_*BH_(t_+1) - delta_L_*BL_(t_+1);
    k2 = chi_* (BH_(t_+1)+BH_(t_+1+1))/2 - delta_L_* (BL_(t_+1)+1/2*k1*k_);
    db_l = k2 * k_;
end

function  w = omega(a_)
% Return age specific rate of waning immunity
    ten_years_old = 520;        % Age in weeks
    if a_ <= ten_years_old
        w = (1/365); % days
    else
        w = 1/(2*365); % days
    end;
    w = w * 7;  % convert to weeks
end

function d = delta(a_)
%   Return age specific disease related mortality rate
    ten_years_old = 520;        % Age in weeks
    if (a_ <= ten_years_old)
        d = 0.032;
    else
        d = 0.007;
    end;
    d = d * 7;  % convert to weeks
end
```

APPENDIX C

Derivation of Explicit Equations for Second Order Accurate Age-Structured Model

The purpose of this section is to take the second order age-structured partial differential equations and convert them into their Lax Wendroff finite difference form. These equations are in an explicit form. The Matlab code in Appendix D refers to these equations.

The Gobbert model [9] PDE equations (ignoring the ODE's for now) are:

$$\begin{aligned}\frac{\partial S}{\partial t}+\alpha\frac{\partial S}{\partial a} &= \Lambda(a,t)+\omega(a)R(a,t)-\beta_L(a)\frac{B_L(t)}{K_L(a)+B_L(t)}S(a,t)\\ &\quad -\beta_H(a)\frac{B_H(t)}{K_H(a)+B_H(t)}S(a,t)-b(a)S(a,t),\end{aligned} \tag{C.1}$$

$$\begin{aligned}\frac{\partial I}{\partial t}+\alpha\frac{\partial I}{\partial a} &= \beta_L(a)\frac{B_L(t)}{K_L(a)+B_L(t)}S(a,t)+\beta_H(a)\frac{B_H(t)}{K_H(a)+B_H(t)}S(a,t)\\ &\quad -b(a)I(a,t)-(1-h(a,t))\Delta(a)I(a,t)\\ &\quad -\gamma_1(1-u(a,t))I(a,t)-\gamma_2u(a,t)I(a,t),\end{aligned} \tag{C.2}$$

$$\begin{aligned}\frac{\partial R}{\partial t}+\alpha\frac{\partial R}{\partial a} &= \gamma_1(1-u(a,t))I(a,t)+\gamma_2u(a,t)I(a,t)-b(a)R(a,t)\\ &\quad -\omega(a)R(a,t),\end{aligned} \tag{C.3}$$

To simplify these equations, we define the following functions:

$$D(a,t)=\beta_L(a)\frac{B_L(t)}{K_L(a)+B_L(t)}+\beta_H(a)\frac{B_H(t)}{K_H(a)+B_H(t)} \tag{C.4}$$

$$E(a,t)=b(a)+(1-h(a,t))\Delta(a)+\gamma_1(1-u(a,t))+\gamma_2u(a,t) \tag{C.5}$$

$$F(a,t) = \gamma_1(1 - u(a,t)) + \gamma_2 u(a,t) \tag{C.6}$$

$$G(a,t) = b(a) + \omega(a) \tag{C.7}$$

Equations C.1 through C.3 then simplify to:

$$\frac{\partial S}{\partial t} + \alpha\frac{\partial S}{\partial a} = \Lambda(a,t) + \omega(a)R(a,t) - (D(a,t) + b(a))S(a,t) \tag{C.8}$$

$$\frac{\partial I}{\partial t} + \alpha\frac{\partial I}{\partial a} = D(a,t)S(a,t) - E(a,t)I(a,t), \tag{C.9}$$

$$\frac{\partial R}{\partial t} + \alpha\frac{\partial R}{\partial a} = F(a,t)I(a,t) - G(a,t)R(a,t), \tag{C.10}$$

Using the Lax-Wendroff Scheme (Strikwerda [17], eq 3.1.1), the corresponding second order finite difference equations are presented below.

C.1 Infected Population

Converting equation C.9 into its Lax-Wendroff form, we get

$$\begin{aligned} I_{i,j+1} = I_{i,j} &- \frac{\alpha k}{2h}\left(I_{i+1,j} - I_{i-1,j}\right) + \frac{\alpha^2 k^2}{2h^2}\left(I_{i+1,j} - 2I_{i,j} + I_{i-1,j}\right) \\ &+ \frac{k}{2}\{D_{i,j+1}S_{i,j+1} - E_{i,j+1}I_{i,j+1}\} \\ &+ \frac{k}{2}\{D_{i,j}S_{i,j} - E_{i,j}I_{i,j}\} \\ &- \frac{\alpha k^2}{4h}\{D_{i+1,j}S_{i+1,j} - E_{i+1,j}I_{i+1,j}\} \\ &+ \frac{\alpha k^2}{4h}\{D_{i-1,j}S_{i-1,j} - E_{i-1,j}I_{i-1,j}\} \end{aligned}$$

Solving for $I_{i,j+1}$ we get

$$\begin{aligned} I_{i,j+1} =& \frac{2}{2+kE_{i,j+1}} \left[I_{i,j} - \frac{\alpha k}{2h}(I_{i+1,j} + I_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(I_{i+1,j} - 2I_{i,j} + I_{i-1,j}) \right] \\ &+ \frac{k}{2+kE_{i,j+1}} \left[D_{i,j+1}S_{i,j+1} + D_{i,j}S_{i,j} - E_{i,j}I_{i,j} \right] \\ &- \frac{\alpha k^2}{h(4+2kE_{i,j+1})} \left[D_{i+1,j}S_{i+1,j} - E_{i+1,j}I_{i+1,j} - D_{i-1,j}S_{i-1,j} + E_{i-1,j}I_{i-1,j} \right] \end{aligned} \tag{C.11}$$

Define the following functions:

$$V = \frac{2}{2+kE_{i,j+1}} \left[I_{i,j} - \frac{\alpha k}{2h}(I_{i+1,j} + I_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(I_{i+1,j} - 2I_{i,j} + I_{i-1,j}) \right]$$

$$X = \frac{\alpha k^2}{h(4+2kE_{i,j+1})} \left[D_{i+1,j}S_{i+1,j} - E_{i+1,j}I_{i+1,j} - D_{i-1,j}S_{i-1,j} + E_{i-1,j}I_{i-1,j} \right]$$

so that Equation C.11 simplifies to

$$I_{i,j+1} = V - X + \frac{k}{2+kE_{i,j+1}} \left[D_{i,j+1}S_{i,j+1} + D_{i,j}S_{i,j} - E_{i,j}I_{i,j} \right] \tag{C.12}$$

C.2 Recovered Population

Converting equation C.10 into its Lax-Wendroff form, we get

$$\begin{aligned} R_{i,j+1} = R_{i,j} &- \frac{\alpha k}{2h}\left(R_{i+1,j} - R_{i-1,j}\right) + \frac{\alpha^2 k^2}{2h^2}\left(R_{i+1,j} - 2R_{i,j} + R_{i-1,j}\right) \\ &+ \frac{k}{2}\{F_{i,j+1}I_{i,j+1} - G_i R_{i,j+1} + F_{i,j}I_{i,j} - G_i R_{i,j}\} \\ &- \frac{\alpha k^2}{4h}\{F_{i+1,j}I_{i+1,j} - G_{i+1}R_{i+1,j} - F_{i-1,j}I_{i-1,j} + G_{i-1}R_{i-1,j}\} \end{aligned}$$

Solving for $R_{i,j+1}$ we get

$$\begin{aligned} R_{i,j+1} =& \frac{2}{2+kG_i}\left[R_{i,j} - \frac{\alpha k}{2h}(R_{i+1,j} - R_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(R_{i+1,j} - 2R_{i,j} + R_{i-1,j})\right] \\ &+ \frac{k}{2+kG_i}\left[F_{i,j+1}I_{i,j+1} + F_{i,j}I_{i,j} - G_i R_{i,j}\right] \\ &- \frac{\alpha k^2}{h(4+2kG_i)}\left[F_{i+1,j}I_{i+1,j} - G_{i+1}R_{i+1,j} - F_{i-1,j}I_{i-1,j} + G_{i-1}R_{i-1,j}\right] \end{aligned} \tag{C.13}$$

Define the following functions:

$$Y = \frac{2}{2+kG_i}\left[R_{i,j} - \frac{\alpha k}{2h}(R_{i+1,j} - R_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(R_{i+1,j} - 2R_{i,j} + R_{i-1,j})\right]$$

$$Z = \frac{\alpha k^2}{h(4+2kG_i)}\left[F_{i+1,j}I_{i+1,j} - G_{i+1}R_{i+1,j} - F_{i-1,j}I_{i-1,j} + G_{i-1}R_{i-1,j}\right]$$

so that Equation C.13 simplifies to

$$R_{i,j+1} = Y - Z + \frac{k}{2+kG_i}\left[F_{i,j+1}I_{i,j+1} + F_{i,j}I_{i,j} - G_i R_{i,j}\right] \tag{C.14}$$

In Euqattion C.14, $R_{i,j+1}$ is a function of $I_{i,j+1}$. Plugging Equation C.12 into Equation C.14, we now have $R_{i,j+1}$ as a function of $S_{i,j+1}$

$$\begin{aligned} R_{i,j+1} =& \frac{k^2 F_{i,j+1}D_{i,j+1}}{(2+kG_i)(2+kE_{i,j+1})}S_{i,j+1} + Y - Z + \frac{k}{2+kG_i}\left[F_{i,j}I_{i,j} - G_i R_{i,j}\right] \\ &+ \frac{k}{2+kG_i}F_{i,j+1}\left[V - X + \frac{k}{2+kE_{i,j+1}}(D_{i,j}S_{i,j} - E_{i,j}I_{i,j})\right] \end{aligned} \tag{C.15}$$

C.3 Susceptible Population

Converting equation C.8 into its Lax-Wendroff form, we get

$$\begin{aligned} S_{i,j+1} = S_{i,j} &- \frac{\alpha k}{2h}\left(S_{i+1,j} - S_{i-1,j}\right) + \frac{\alpha^2 k^2}{2h^2}\left(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}\right) \\ &+ \frac{k}{2}\{\Lambda_{i,j+1} + \omega_i R_{i,j+1} - (D_{i,j+1} + b_i)S_{i,j+1}\} \\ &+ \frac{k}{2}\{\Lambda_{i,j} + \omega_i R_{i,j} - (D_{i,j} + b_i)S_{i,j}\} \\ &- \frac{\alpha k^2}{4h}\{\Lambda_{i+1,j} + \omega_{i+1} R_{i+1,j} - (D_{i+1,j} + b_{i+1})S_{i+1,j}\} \\ &- \frac{\alpha k^2}{4k}\{-\Lambda_{i-1,j} - \omega_{i-1} R_{i-1,j} + (D_{i-1,j} + b_{i-1})S_{i-1,j}\} \end{aligned} \tag{C.16}$$

In Euqattion C.16, $S_{i,j+1}$ is a function of $R_{i,j+1}$. Plugging Equation C.15 into Equation C.16, we now have $S_{i,j+1}$ as a function of $S_{i,j+1}$.

$$\begin{aligned} S_{i,j+1} =& \frac{k\omega_i}{2}\left(\frac{k^2 F_{i,j+1} D_{i,j+1}}{(2 + kG_i)(2 + kE_{i,j+1})}\right) S_{i,j+1} \\ &+ \frac{k\omega_i}{2}\left(Y - Z + \frac{k}{2 + kG_i}\left[F_{i,j} I_{i,j} - G_i R_{i,j}\right]\right) \\ &+ \frac{k\omega_i}{2}\left(\frac{k}{2 + kG_i} F_{i,j+1}\left[V - X + \frac{k}{2 + kE_{i,j+1}}(D_{i,j} S_{i,j} - E_{i,j} I_{i,j})\right]\right) \\ &+ \frac{k}{2}\left(\Lambda_{i,j+1} - (D_{i,j+1} + b_i)S_{i,j+1} + \Lambda_{i,j} + \omega_i R_{i,j} - (D_{i,j} + b_i)S_{i,j}\right) \\ &- \frac{\alpha k^2}{4h}\left(\Lambda_{i+1,j} + \omega_{i+1} R_{i+1,j} - (D_{i+1,j} + b_{i+1})S_{i+1,j}\right) \\ &- \frac{\alpha k^2}{4h}\left(-\Lambda_{i-1,j} - \omega_{i-1} R_{i-1,j} + (D_{i-1,j} + b_{i-1})S_{i-1,j}\right) \\ &+ S_{i,j} - \frac{\alpha k}{2h}\left(S_{i+1,j} - S_{i-1,j}\right) + \frac{\alpha^2 k^2}{2h^2}\left(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}\right) \end{aligned} \tag{C.17}$$

Solving for $S_{i,j+1}$ in Equation C.17 we get

$$
\begin{aligned}
S_{i,j+1} =& \frac{1}{1 - \frac{k\omega_i}{2}\frac{k^2 F_{i,j+1} D_{i,j+1}}{(2+kG_i)(2+kE_{i,j+1})} + \frac{k}{2}(D_{i,j+1} + b_i)} \\
& * [\\
& \frac{k\omega_i}{2}\left(Y - Z + \frac{k}{2+kG_i}[F_{i,j}I_{i,j} - G_i R_{i,j}]\right) \\
& + \frac{k\omega_i}{2}\left(\frac{k}{2+kG_i}F_{i,j+1}\left[V - X + \frac{k}{2+kE_{i,j+1}}(D_{i,j}S_{i,j} - E_{i,j}I_{i,j})\right]\right) \\
& + \frac{k}{2}(\Lambda_{i,j+1} + \Lambda_{i,j} + \omega_i R_{i,j} - (D_{i,j} + b_i)S_{i,j}) \\
& - \frac{\alpha k^2}{4h}(\Lambda_{i+1,j} + \omega_{i+1}R_{i+1,j} - (D_{i+1,j} + b_i)S_{i+1,j}) \\
& - \frac{\alpha k^2}{4h}(-\Lambda_{i-1,j} - \omega_{i-1}R_{i-1,j} + (D_{i-1,j} + b_{i-1})S_{i-1,j}) \\
& + S_{i,j} - \frac{\alpha k}{2h}(S_{i+1,j} - S_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}) \\
&]
\end{aligned}
\tag{C.18}
$$

C.4 The Second Order Accurate Model Algorithms

To summarize, below are the three algorithms (and their sub-functions) that represent the susceptible, infected, and recovered populations.

Susceptible Population

$$S_{i,j+1} = \frac{1}{1 - \frac{k\omega_i}{2}\frac{k^2 F_{i,j+1} D_{i,j+1}}{(2+kG_i)(2+kE_{i,j+1})} + \frac{k}{2}(D_{i,j+1} + b_i)}$$

$$\begin{aligned}
&* [\\
&\frac{k\omega_i}{2}\left(Y - Z + \frac{k}{2+kG_i}[F_{i,j}I_{i,j} - G_i R_{i,j}]\right) \\
&+ \frac{k\omega_i}{2}\left(\frac{k}{2+kG_i}F_{i,j+1}\left[V - X + \frac{k}{2+kE_{i,j+1}}(D_{i,j}S_{i,j} - E_{i,j}I_{i,j})\right]\right) \\
&+ \frac{k}{2}(\Lambda_{i,j+1} + \Lambda_{i,j} + \omega_i R_{i,j} - (D_{i,j} + b_i)S_{i,j}) \\
&- \frac{\alpha k^2}{4h}(\Lambda_{i+1,j} + \omega_{i+1}R_{i+1,j} - (D_{i+1,j} + b_i)S_{i+1,j}) \\
&- \frac{\alpha k^2}{4h}(-\Lambda_{i-1,j} - \omega_{i-1}R_{i-1,j} + (D_{i-1,j} + b_{i-1})S_{i-1,j}) \\
&+ S_{i,j} - \frac{\alpha k}{2h}(S_{i+1,j} - S_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(S_{i+1,j} - 2S_{i,j} + S_{i-1,j}) \\
&]
\end{aligned} \tag{C.19}$$

Infected Population

$$I_{i,j+1} = V - X + \frac{k}{2+kE_{i,j+1}}[D_{i,j+1}S_{i,j+1} + D_{i,j}S_{i,j} - E_{i,j}I_{i,j}] \tag{C.20}$$

Recovered Population

$$R_{i,j+1} = Y - Z + \frac{k}{2+kG_i}[F_{i,j+1}I_{i,j+1} + F_{i,j}I_{i,j} - G_i R_{i,j}] \tag{C.21}$$

With the following sub-functions:

$$D(a,t) = \beta_L(a)\frac{B_L(t)}{K_L(a) + B_L(t)} + \beta_H(a)\frac{B_H(t)}{K_H(a) + B_H(t)}$$

$$E(a,t) = b(a) + (1 - h(a,t))\Delta(a) + \gamma_1(1 - u(a,t)) + \gamma_2 u(a,t)$$

$$F(a,t) = \gamma_1(1 - u(a,t)) + \gamma_2 u(a,t)$$

$$G(a,t) = b(a) + \omega(a)$$

$$V = \frac{2}{2 + kE_{i,j+1}} \left[I_{i,j} - \frac{\alpha k}{2h}(I_{i+1,j} - I_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(I_{i+1,j} - 2I_{i,j} + I_{i-1,j}) \right]$$

$$X = \frac{\alpha k^2}{h(4 + 2kE_{i,j+1})} \left[D_{i+1,j}S_{i+1,j} - E_{i+1,j}I_{i+1,j} - D_{i-1,j}S_{i-1,j} + E_{i-1,j}I_{i-1,j} \right]$$

$$Y = \frac{2}{2 + kG_i} \left[R_{i,j} - \frac{\alpha k}{2h}(R_{i+1,j} - R_{i-1,j}) + \frac{\alpha^2 k^2}{2h^2}(R_{i+1,j} - 2R_{i,j} + R_{i-1,j}) \right]$$

$$Z = \frac{\alpha k^2}{h(4 + 2kG_i)} \left[F_{i+1,j}I_{i+1,j} - G_{i+1}R_{i+1,j} - F_{i-1,j}I_{i-1,j} + G_{i-1}R_{i-1,j} \right]$$

APPENDIX D

Matlab - 2nd Order Accurate Age-Structured Cholera Model - Explicit Equations

```
function [] = SIR_AgeStructured_2nd_order_explicit()
%%  Project - Age Structured Cholera Model
%  Author: John Szijjarto
%  Date:   10/27/11
%  Title:  (2,2) Order Accurate Lax-Wendroff Scheme Epidemic Cholera Model
%          (2) Order Accurate - Midpoint Scheme for dBL/dt ODE
%
%  Running the Program:
%       Reference Simulation - Variable Settings
%           reference_sim = true;
%       Hi Rate Shedding Simulation - Variable Settings
%           reference_sim = false;
%           xi            = 1000000000;
%       Low Rate Shedding Simulation - Variable Settings
%           reference_sim = false;
%           xi            = 100;
%

%% %%%%%% Matlab Environment Commands  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
clear all;  % Close/Delete all figures
close all;  % Free system memory
clc;        % Clear command window

outfile_1 = fopen('output.txt','w');


%% %%%%%% Simulation Control Variables  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

reference_sim = false;      % if true, run ref sim (ie. no infected pop)

%xi            = 1000000000; % Hi Rate of shedding of cholera vibrios from
xi             = 100;        % Lo Rate of shedding of cholera vibrios from
                             % infected human of age a.

%% %%%%%% Model Variables %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
k     = 1/50;           % detla_t = 1/50 of a week
h     = 1;              % delta_a = 1 week
alpha = 1;              % Proportionality factor (wave speed)
b     = (1/50)*(1/52);  % normal mortality rate in deaths per week

% Declare the S, I, R Arrays and zero out all values.
%
% Extend the age dimension (m) out twice as far to handle RHS boundary
m = 3744*2;             % cols -> total age = 72 * 52 = 3744 weeks

n = 24/k;               % rows -> total time = 24 weeks

S = zeros(m+2,n);       % Suseptable population
I = zeros(m+2,n);       % Infected population
R = zeros(m+2,n);       % Recovered population

BH  = zeros(n,1);       % Concentration of HI vibrio at time t
BL  = zeros(n,1);       % Concentration of non-HI vibrio at time t

A       = 72;           % Upper bound on human age
```

```
59
60  kappa_L  = 1000000;     % Saturation Constant of non-Hi vibrios at age a
61  kappa_H  = kappa_L/700; % Saturation Constant of Hi vibrios at age a
62  beta_L   = 1.5/7*7;     % Ingestion rate of non-Hi vibrios at age a
63  beta_H   = 1.5/7*7;     % Ingestion rate of Hi vibrios at age a
64  lambda   = 0*7;         % Human recruitment rate (non-newborns entering pop)
65  gamma_1  = 1/5*7;       % recovery rate of untreated cholera
66  gamma_2  = 1/3*7;       % recovery rate of treated cholera
67  u        = 0*7;         % antiboitic treatment rate - humans age a at time t
68  H2O      = 0.9;         % hydration therapy related mortality.(h(a,t) in eq)
69                          % This is a percentage value.
70  delta_L  = 1/30*7;      % Death rate of vibrio in the environment.
71
72  eta      = 0.1;         % Relative amount of stool per unit time - no units
73
74  chi      = 1/5*24*7;    % Rate of vibrio moving from HI to non-HI state.
75
76
77  %% %%%%% Initial Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
78  %
79  eighteen_years_old = 936;   % Age in weeks
80  nineteen_years_old = 988;   % Age in weeks
81
82  % Initial Conditions for I (Infected)
83  %
84  if (reference_sim == false) % reference sim contains no infected people
85      % Include one 18 year old infected human
86      for a=(m-1):-1:0
87          if (a>=eighteen_years_old) && (a<nineteen_years_old)
88              I(a+1,0+1) = 1/52.0;
89          else
90              I(a+1,0+1) = 0.0;
91          end
92      end
93  end
94
95  % Initial Conditions for R (Recovered) and S (Susceptible)
96  one_year_old = 52;   % Age in weeks
97  for a=(m-1):-1:0
98      % Susceptable and Recovered Population
99      % 2.67094017 * 52wks * 72yrs = 10000 people
100     %
101     if (a<=one_year_old) % if age<1 year-old then immune (ie. in R group)
102         R(a+1,0+1) = 2.67094017;
103         S(a+1,0+1) = 0.0;
104     else
105         R(a+1,0+1)  = 0.0;
106         S(a+1,0+1) = 2.67094017;
107         if (reference_sim == false)  % Hi or Low rate shedding scenario
108             if (a>=eighteen_years_old) && (a<nineteen_years_old)
109                 S(a+1,0+1) = 2.67094017 - 1/52.0; %minus 1 infected person
110             end
111         end
112     end
113 end
114
115
116
117
118 %% %%%%% Boundary Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
119 %       Note: Boundary conditions for R (Recovered) array are created
120 %             within the main simulation loop.
121 %
122 for t = 0:(n-1)
123     % Susceptable and Infected Population
124     S(0+1,t+1) = 0.0;
125     I(0+1,t+1) = 0.0;
126 end
```

```
127
128
129 %% %%%%% Run the Simulation %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
130 %
131 % Generate All Other Interior Grid Points
132 %
133 for t = 0:(n-1)
134     % Calculate current hyperinfective (BH) and non-hyperinfective (BL)
135     % cholera bacteria populations
136     dBH = get_dBH(I,t,BH,xi,chi,eta,A,k,h);
137     BH(t+1+1) = BH(t+1) + dBH;
138     dBL = get_dBL(chi,BH,delta_L,BL,t,k,h);
139     BL(t+1+1) = BL(t+1) + dBL;
140
141     % Suseptable Population - Generate one row %%%%%%%%%%%%%%%%%%%%%%%%%%%
142      for a=1:(m-1)
143         S(a+1,t+1+1) = ...
144           1/(     1 ...
145                 - k/2*omega(a+1)*k^2*(gamma_1*(1-u)+gamma_2*u)...
146                     *(beta_L*BL(t+1+1)/(kappa_L+BL(t+1+1)) ...
147                         + beta_H*BH(t+1+1)/(kappa_H+BH(t+1+1)))...
148                             /((2+k*(b + omega(a+1)))...
149                             *(2+k*(b + (1-H2O)*delta(a+1) ...
150                         + gamma_1*(1-u) + gamma_2*u)) )...
151                 + k/2*((beta_L*BL(t+1+1)/(kappa_L+BL(t+1+1)) ...
152                     + beta_H*BH(t+1+1)/(kappa_H+BH(t+1+1)))+ b)...
153             )...
154           * (...
155                 k/2*omega(a+1) ...
156                   * (...
157                           2/(2+k*(b + omega(a+1))) ...
158                             * (   R(a+1,t+1) ...
159                                 - alpha*k/(2*h)*(R(a+1+1,t+1)...
160                                     -R(a+1-1,t+1)) ...
161                                 + (alpha*k)^2/(2*h^2)* (R(a+1+1,t+1)...
162                                     -2*R(a+1,t+1)+R(a+1-1,t+1))...
163                               )...
164                         - alpha*k^2/(h*(4+2*k*(b + omega(a+1))))...
165                             * (  (gamma_1*(1-u)+gamma_2*u)*I(a+1+1,t+1) ...
166                                 - (b + omega(a+1+1))*R(a+1+1,t+1) ...
167                                 -(gamma_1*(1-u)+gamma_2*u)*I(a+1-1,t+1) ...
168                                 + (b + omega(a+1-1))*R(a+1-1,t+1) ...
169                                )...
170                         + k/(2+k*(b + omega(a+1)))*((gamma_1*(1-u)...
171                             +gamma_2*u)*I(a+1,t+1)...
172                                 - (b + omega(a+1))*R(a+1,t+1))...
173                         + k/(2+k*(b + omega(a+1)))...
174                             *(gamma_1*(1-u)+gamma_2*u)...
175                             * (...
176                                    2/(2+k*(b + (1-H2O)*delta(a+1) ...
177                                     + gamma_1*(1-u) + gamma_2*u)) ...
178                                     * (   I(a+1,t+1) ...
179                                         - alpha*k/(2*h)*( I(a+1+1,t+1) ...
180                                             - I(a+1-1,t+1)) ...
181                                         + (alpha*k)^2/(2*h^2)...
182                                             *(I(a+1+1,t+1)...
183                                               - 2*I(a+1,t+1)...
184                                                 + I(a+1-1,t+1)...
185                                            )...
186                                       )...
187                                 - alpha*k^2/(h*(4+2*k*(b + (1-H2O)...
188                                    *delta(a+1) + gamma_1*(1-u)...
189                                        +gamma_2*u)))...
190                                    * ((beta_L*BL(t+1)/(kappa_L+BL(t+1))...
191                                     + beta_H*BH(t+1)/(kappa_H+BH(t+1)))...
192                                      *S(a+1+1,t+1) ...
193                                        - (b + (1-H2O)*delta(a+1+1)...
194                                        + gamma_1*(1-u) + gamma_2*u)...
```

```
                                        *I(a+1+1,t+1) ...
                                        - (beta_L*BL(t+1)/...
                                       (kappa_L+BL(t+1))+ beta_H*BH(t+1)...
                                       /(kappa_H+BH(t+1)))*S(a+1-1,t+1) ...
                                        + (b + (1-H2O)*delta(a+1-1)...
                                        + gamma_1*(1-u) + gamma_2*u)...
                                        *I(a+1-1,t+1) ...
                                       )...
                                  + k/(2+k*(b + (1-H2O)*delta(a+1)...
                                     + gamma_1*(1-u) + gamma_2*u))...
                                     * ((beta_L*BL(t+1)/(kappa_L+BL(t+1)))...
                                     + beta_H*BH(t+1)/(kappa_H+BH(t+1)))...
                                     *S(a+1,t+1)...
                                     -(b + (1-H2O)*delta(a+1)...
                                     + gamma_1*(1-u) + gamma_2*u)...
                                     *I(a+1,t+1))...
                               )...
                    ) ...
                  + k/2 ...
                    * ( ...
                        omega(a+1)*R(a+1,t+1)-((beta_L*BL(t+1)...
                            /(kappa_L+BL(t+1)) + beta_H*BH(t+1)...
                            /(kappa_H+BH(t+1)))+b)*S(a+1,t+1)...
                      )...
                  - alpha*k^2/(4*h) ...
                    * (...
                          omega(a+1+1)*R(a+1+1,t+1)...
                        - ((beta_L*BL(t+1)/(kappa_L+BL(t+1))...
                            + beta_H*BH(t+1)/(kappa_H+BH(t+1)))+b)...
                                *S(a+1+1,t+1)...
                        - omega(a+1-1)*R(a+1-1,t+1)...
                        + ((beta_L*BL(t+1)/(kappa_L+BL(t+1))...
                            + beta_H*BH(t+1)/(kappa_H+BH(t+1)))+b)...
                                *S(a+1-1,t+1)...
                      )...
              + S(a+1,t+1)...
              - alpha*k/(2*h)* (S(a+1+1,t+1)-S(a+1-1,t+1))...
              + alpha^2*k^2/(2*h^2) * (S(a+1+1,t+1)-2*S(a+1,t+1)...
                +S(a+1-1,t+1))...
        );
    end; % Suseptable population loop %%%%%%%%%%%%%%%%%%%%%%%%%%

    % Infected Population - Generate one row
    for a=1:(m-1)
       I(a+1,t+1+1) = ...
           2/(2+k*(b + (1-H2O)*delta(a+1) + gamma_1*(1-u)+ gamma_2*u)) ...
               * (   I(a+1,t+1) ...
                   - alpha*k/(2*h)*( I(a+1+1,t+1) - I(a+1-1,t+1)) ...
                   + (alpha*k)^2/(2*h^2)*(I(a+1+1,t+1)...
                       - 2*I(a+1,t+1) + I(a+1-1,t+1)...
                     )...
                )...
         + k/(2+k*(b +(1-H2O)*delta(a+1) + gamma_1*(1-u) + gamma_2*u)) ...
               * (   (beta_L*BL(t+1+1)/(kappa_L+BL(t+1+1))...
                  + beta_H*BH(t+1+1)/(kappa_H+BH(t+1+1)))*S(a+1,t+1+1) ...
                   + (beta_L*BL(t+1)/(kappa_L+BL(t+1)) + beta_H*BH(t+1)...
                       /(kappa_H+BH(t+1)))*S(a+1,t+1) ...
                   - (b + (1-H2O)*delta(a+1) + gamma_1*(1-u)...
                       + gamma_2*u)*I(a+1,t+1) ...
                  )...
         - alpha*k^2/(h*(4+2*k*(b + (1-H2O)*delta(a+1) + gamma_1*(1-u)...
           + gamma_2*u)))...
               * (   (beta_L*BL(t+1)/(kappa_L+BL(t+1)) + beta_H*BH(t+1)...
                   /(kappa_H+BH(t+1)))*S(a+1+1,t+1) ...
                   - (b + (1-H2O)*delta(a+1+1) + gamma_1*(1-u)...
                       + gamma_2*u)*I(a+1+1,t+1) ...
                   - (beta_L*BL(t+1)/(kappa_L+BL(t+1)) + beta_H*BH(t+1)...
                       /(kappa_H+BH(t+1)))*S(a+1-1,t+1) ...
```

```
                    + (b + (1-H20)*delta(a+1-1) + gamma_1*(1-u)...
                        + gamma_2*u)*I(a+1-1,t+1) ...
                    );
     end; % Infected population loop


    % Recovered Population %%%%%%%%%%%%%%%%%%%%%%%%%%
    %
    % Include new born babies into the recovered population this week
    fecundity = 0;
    for age=779:1:2339
        fecundity = fecundity...
            + (S(age+1,t+1) + I(age+1,t+1) + R(age+1,t+1))...
            * (1/5) * (sin((age-780)/1560*3.14159))^2;
    end
    R(0+1,t+1)= fecundity/52; %Divided by 52 weeks/year (~8 born per week)


    % Recovered Population - Generate one row
    for a=1:(m-1)
        R(a+1,t+1+1)= ...
            2/(2+k*(b + omega(a+1))) ...
                * (   R(a+1,t+1) ...
                    - alpha*k/(2*h)*(R(a+1+1,t+1) - R(a+1-1,t+1)) ...
                    + (alpha*k)^2/(2*h^2)* (R(a+1+1,t+1)-2*R(a+1,t+1)...
                        +R(a+1-1,t+1))...
                  )...
          + k/(2+k*(b + omega(a+1))) ...
                * (   (gamma_1*(1-u)+gamma_2*u)*I(a+1,t+1+1) ...
                    + (gamma_1*(1-u)+gamma_2*u)*I(a+1,t+1) ...
                    - (b + omega(a+1))*R(a+1,t+1) ...
                  )...
          - alpha*k^2/(h*(4+2*k*(b + omega(a+1))))...
                * (   (gamma_1*(1-u)+gamma_2*u)*I(a+1+1,t+1) ...
                    - (b + omega(a+1+1))*R(a+1+1,t+1) ...
                    - (gamma_1*(1-u)+gamma_2*u)*I(a+1-1,t+1) ...
                    + (b + omega(a+1-1))*R(a+1-1,t+1) ...
                  );
    end % Recovered population loop

    if (mod(t,100) == 0)
        fprintf('t = %d\n', t);   % This is just a program status output
    end
end % Main time loop

%% %%%%% Output Results %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%

% %%%%% Print Population Totals %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
total_sus_population = zeros(n,1); % column vector of length n
total_inf_population = zeros(n,1); % column vector of length n
total_rec_population = zeros(n,1); % column vector of length n
total_population = zeros(n,1);     % column vector of length n

fprintf(outfile_1,...
    '\nSUSEPTABLE  INFECTED RECOVERED  TOTAL-POPULATION  BIRTHS\r\n\r\n');

for t=0:(n-1)
    for a=0:(m/2-1)
        total_sus_population(t+1) = total_sus_population(t+1) ...
                                    + S(a+1,t+1);
        total_inf_population(t+1) = total_inf_population(t+1) ...
                                    + I(a+1,t+1);
        total_rec_population(t+1) = total_rec_population(t+1) ...
                                    + R(a+1,t+1);
    end
    total_population(t+1)         = total_sus_population(t+1) ...
                                    + total_inf_population(t+1)...
```

```
                                    + total_rec_population(t+1);

    fprintf(outfile_1,'%f %f %f %f %f\r\n',...
        total_sus_population(t+1),...
        total_inf_population(t+1),...
        total_rec_population(t+1), ...
        total_population(t+1),...
        R(0+1,t+1));  % Print Births for each week
end
fclose(outfile_1);


% Produce 2D Plots
%

x=1:1:n-1;

figure;
plot(x,total_sus_population(x),'b',x,total_inf_population(x),'r',...
    x,total_rec_population(x),'g');
title('Lo Rate Shedding - Human Populations');
xlabel('Time: 3.36 hrs/step');
ylabel('Population');
legend('Susceptible', 'Infected', 'Recovered');
grid;

figure;
plot(x,BH(x),'b',x,BL(x),'r');
title('Lo Rate Shedding - Vibrio Populations');
xlabel('Time: 3.36 hrs/step');
ylabel('Vibrio Population');
legend('BH Vibrio', 'BL Vibrio');
grid;

figure;
plot(x,total_sus_population(x));
title('Susceptible Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;

figure;
plot(x,total_inf_population(x));
title('Infected Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;

figure;
plot(x,total_rec_population(x));
title('Recovered Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;

figure;
plot(x,total_population(x));
title('Total Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;

figure;
plot(x,BH(x));
title('BH Vibrio Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;
```

```
399
400 figure;
401 plot(x,BL(x));
402 title('BL Vibrio Population');
403 xlabel('Time: 3.36hrs/step - 24 weeks total');
404 ylabel('Population');
405 grid;
406
407 end % Main Program
408
409 %% %%%%%% Sub Functions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
410 % %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
411 %
412 function db_h = get_dBH(I_,t_,BH_,xi_,chi_,eta_,A_,k_,h_)
413 % Return change in HI vibrio population
414     sum = 0;
415     for age = 0:(A_*52-2)  % age is in weeks
416         sum = sum + I_(age+1+1,t_+1) + I_(age+1,t_+1);
417     end;
418     sum = xi_*eta_*h_*(1/2)*sum;
419     dt  = k_;
420     db_h = dt*h_*(sum - chi_*BH_(t_+1));
421 end
422
423 function db_l = get_dBL(chi_,BH_,delta_L_,BL_,t_,k_,h_)
424 % Return change in non-HI vibrio population
425 %   second order accuracy midpoint scheme used here.
426     k1 = chi_*BH_(t_+1) - delta_L_*BL_(t_+1);
427     k2 = chi_* (BH_(t_+1)+BH_(t_+1+1))/2 - delta_L_*(BL_(t_+1)+ 1/2*k1*k_);
428     db_l = k2 * k_;
429 end
430
431 function  w = omega(a_)
432 % Return age specific rate of waning immunity
433     ten_years_old = 520;        % Age in weeks
434     if a_ <= ten_years_old
435         w = (1/365); % days
436     else
437         w = 1/(2*365); % days
438     end;
439     w = w * 7;  % convert to weeks
440 end
441
442 function d = delta(a_)
443 %   Return age specific disease related mortality rate
444     ten_years_old = 520;        % Age in weeks
445     if (a_ <= ten_years_old)
446         d = 0.032;
447     else
448         d = 0.007;
449     end;
450     d = d * 7;  % convert to weeks
451 end
```

APPENDIX E

Plot Results of Matlab Code for the Second Order Accurate Age-Structured Cholera Model

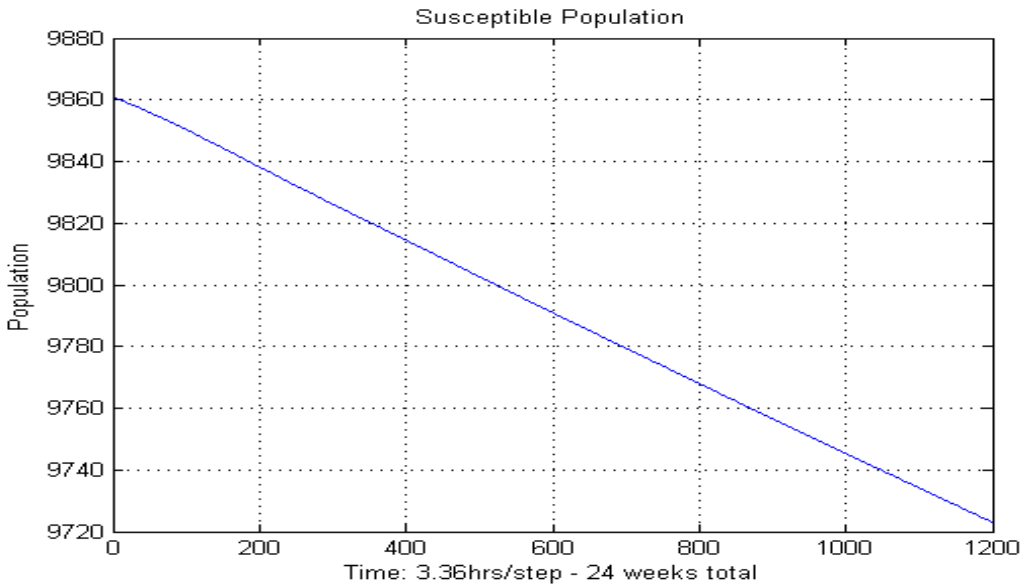

Figure E.1: Reference Simulation - Suseptable Population - 2nd Order Accuracy

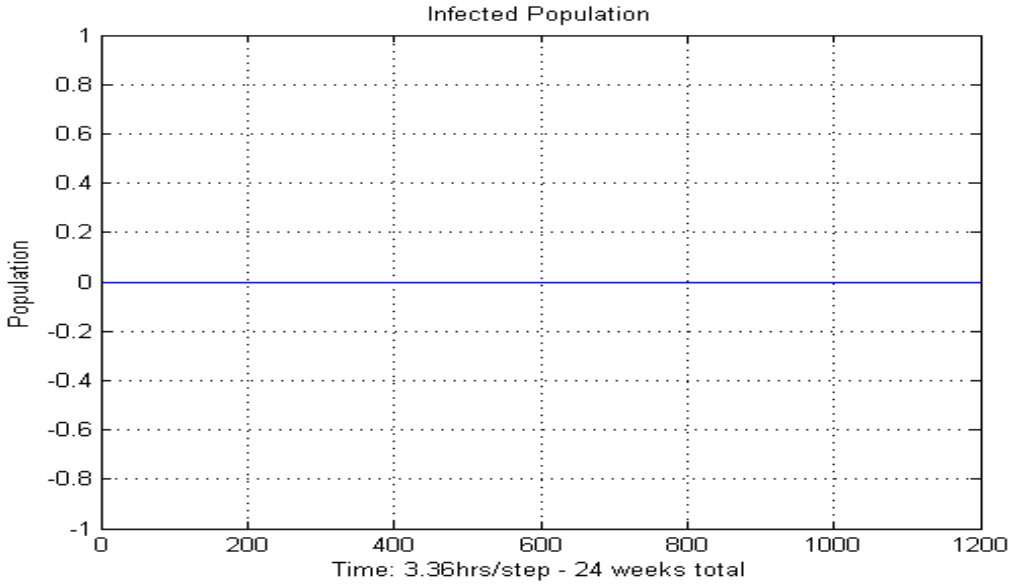

Figure E.2: Reference Simulation - Infected Population - 2nd Order Accuracy

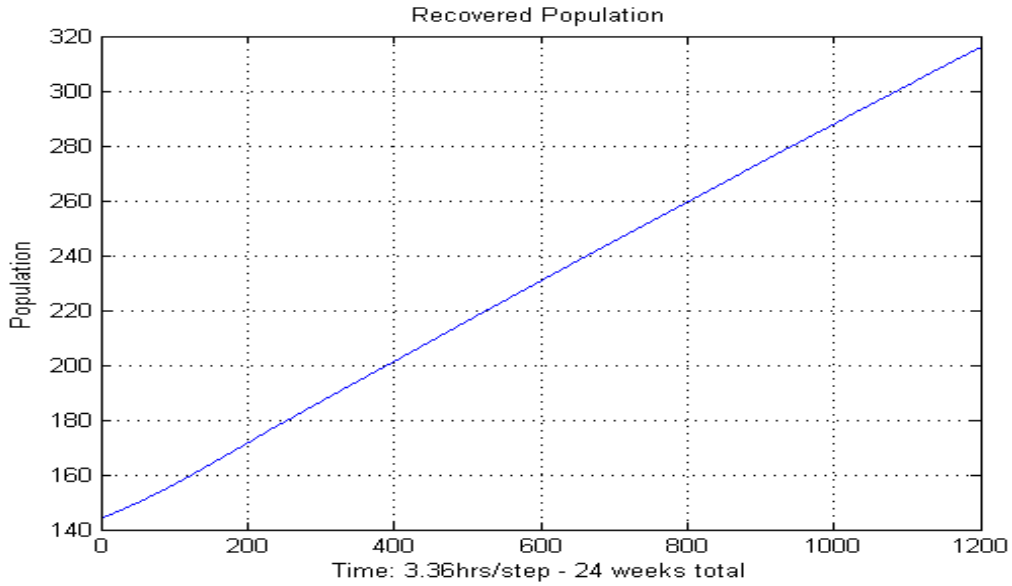

Figure E.3: Reference Simulation - Recovered Population - 2nd Order Accuracy

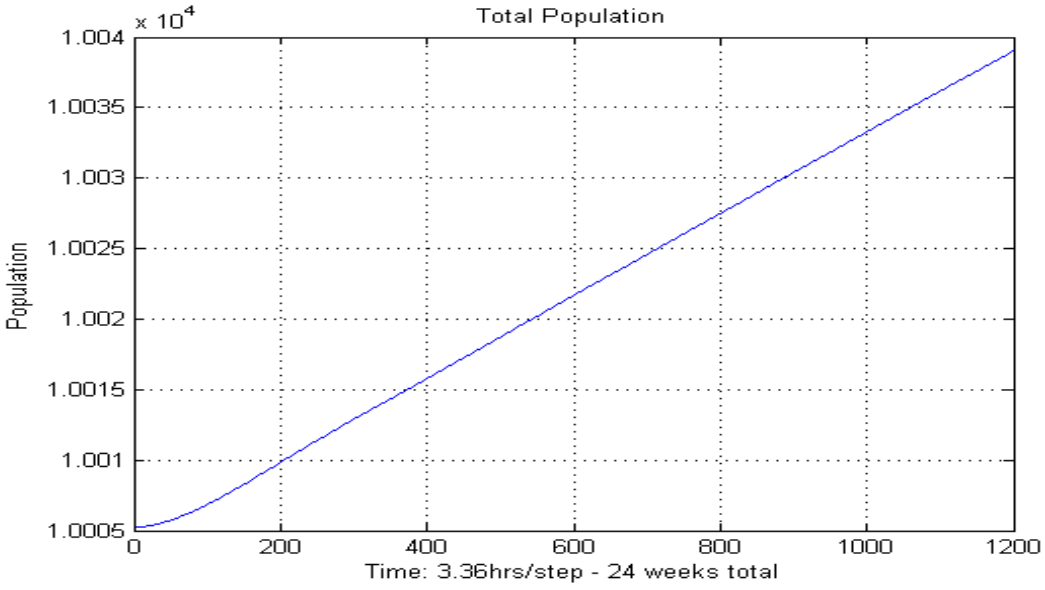

Figure E.4: Reference Simulation - Total Population - 2nd Order Accuracy

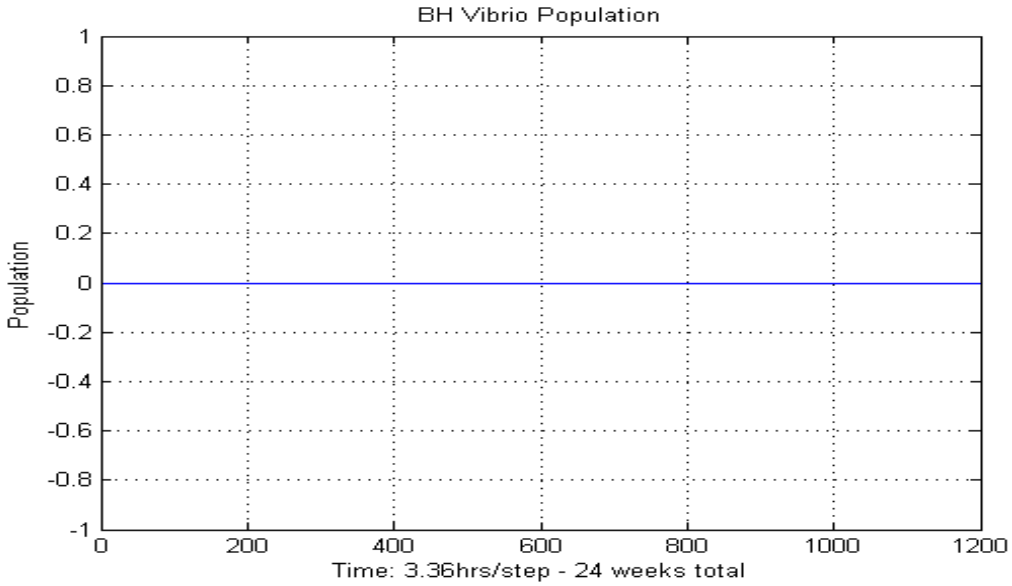

Figure E.5: Reference Simulation - BH Vibrio Population - 2nd Order Accuracy

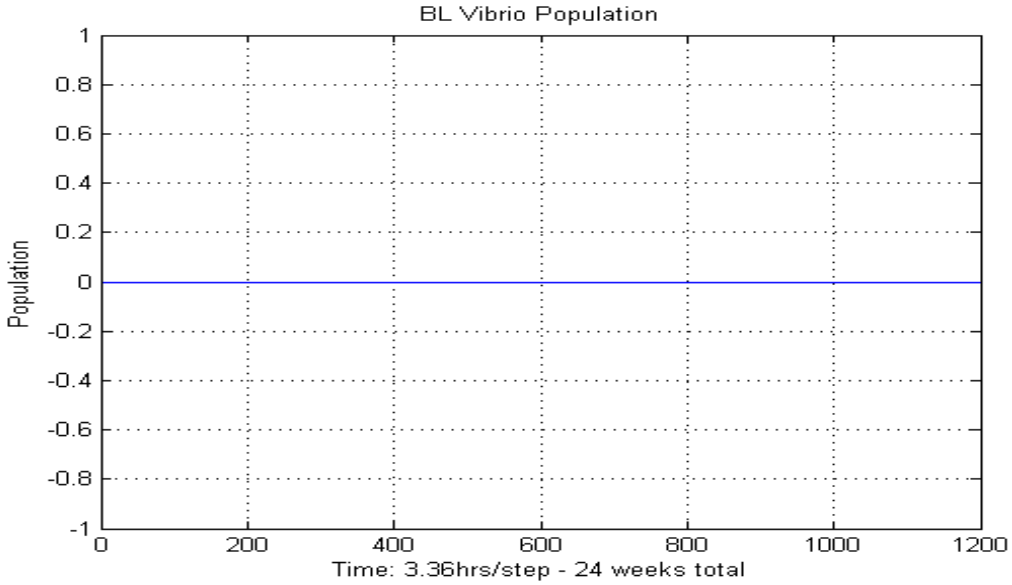

Figure E.6: Reference Simulation - BL Vibrio Population - 2nd Order Accuracy

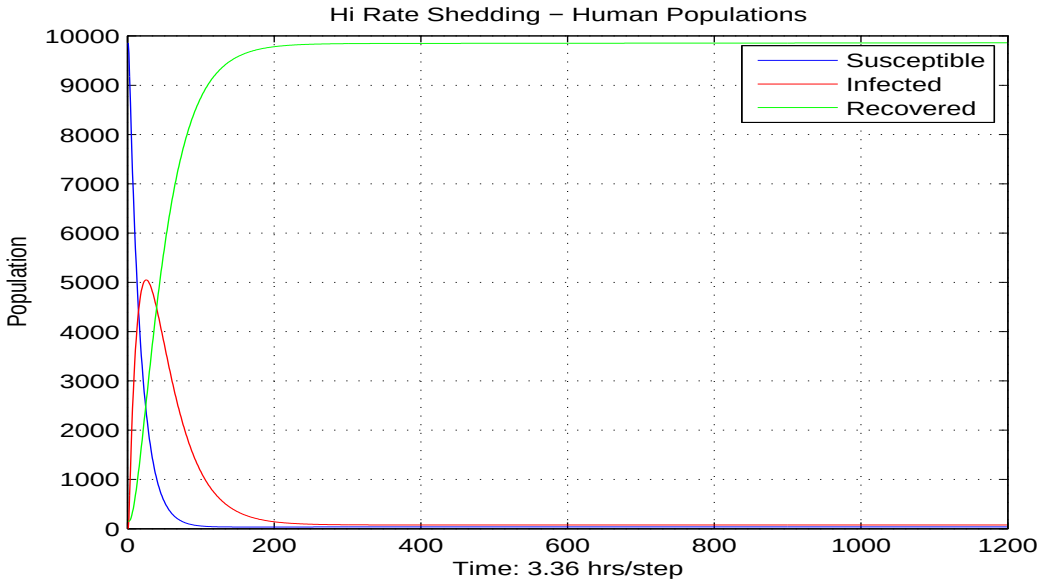

Figure E.7: High Rate Shedding Sim - Human Populations - 2nd Order Accuracy

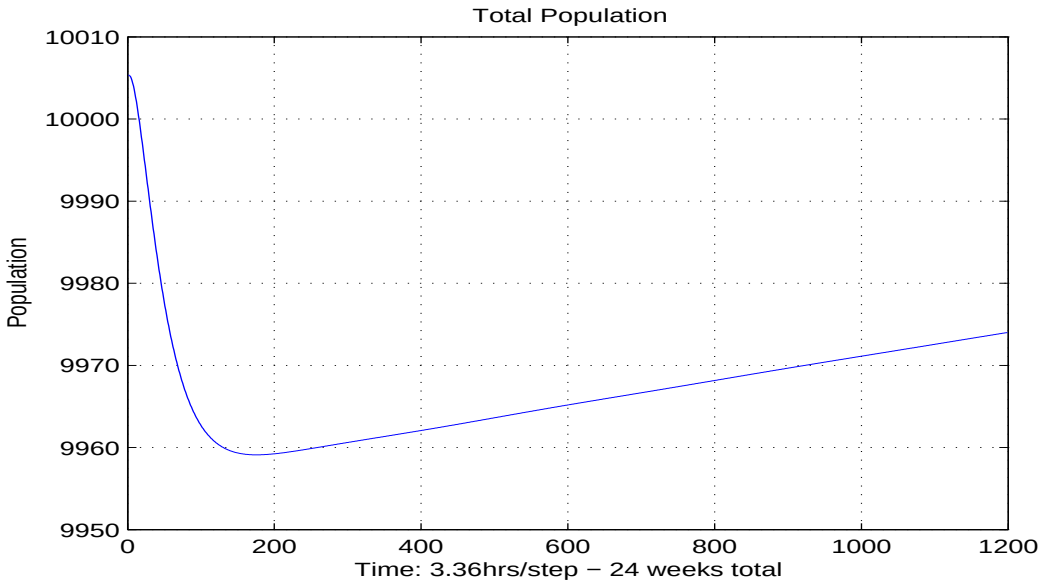

Figure E.8: High Rate Shedding Sim - Total Pop - 2nd Order Accuracy

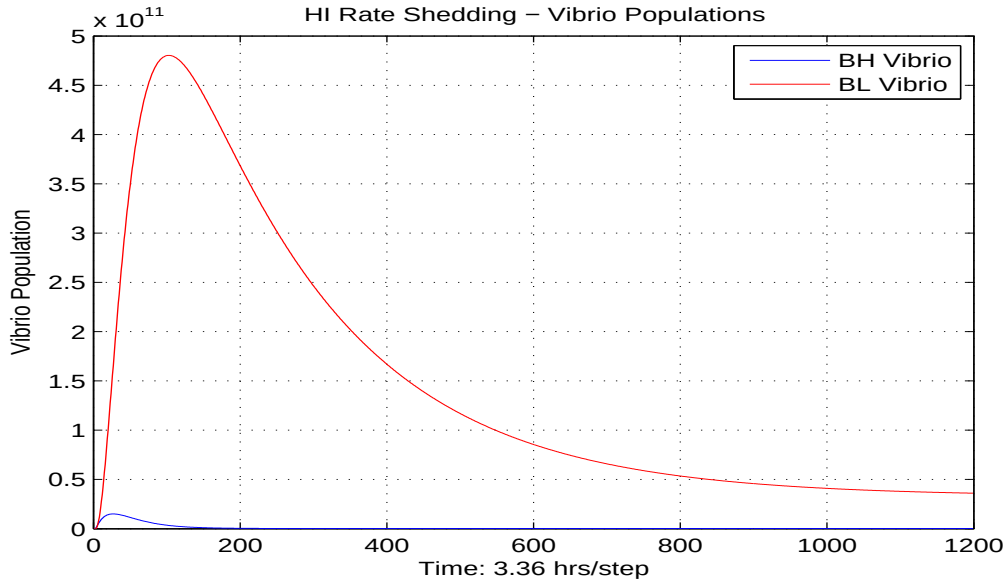

Figure E.9: High Rate Shedding Sim - Vibrio Populations - 2nd Order Accuracy

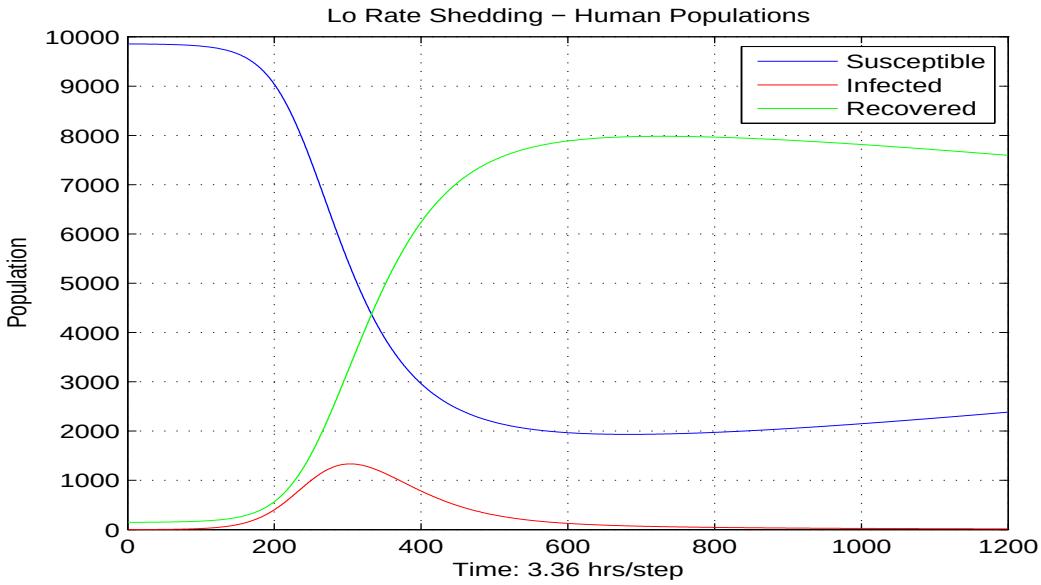

Figure E.10: Low Rate Shedding Sim - Human Populations - 2nd Order Accuracy

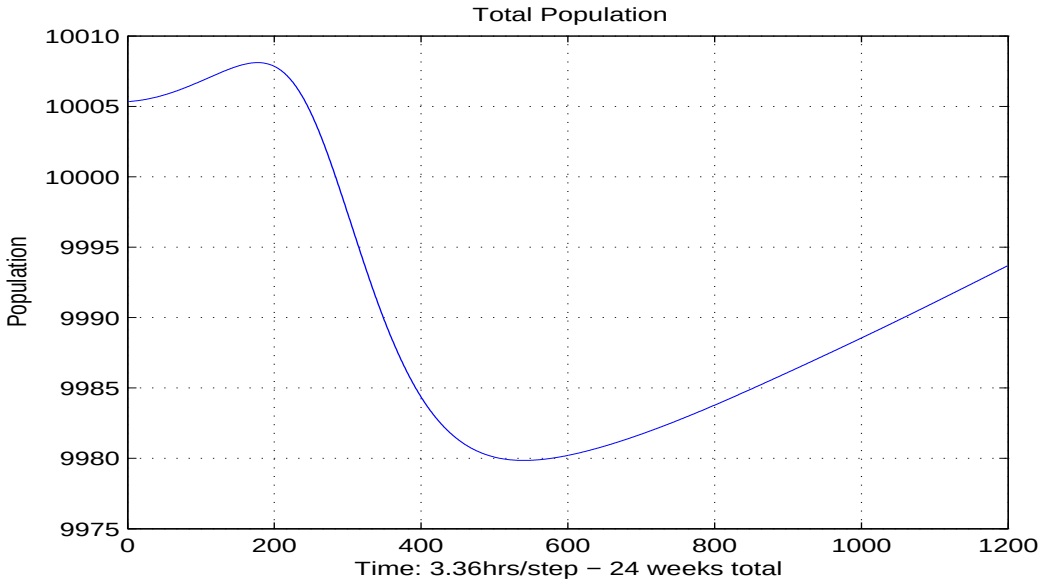

Figure E.11: Low Rate Shedding Sim - Total Population - 2nd Order Accuracy

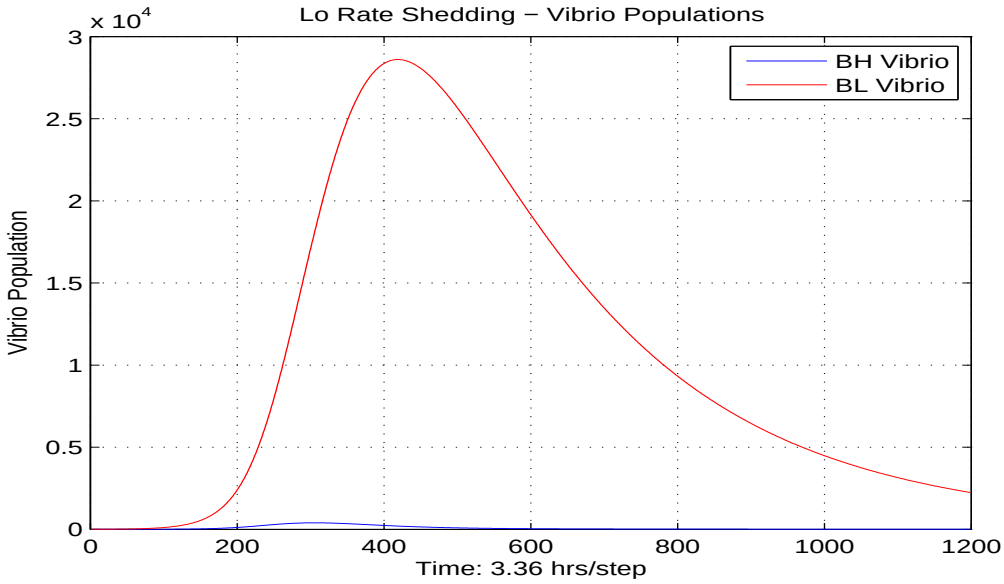

Figure E.12: Low Rate Shedding Sim - Vibrio Populations - 2nd Order Accuracy

APPENDIX F

Matlab Code - The Effects of Cholera on Children Versus Adults

```
function [] = SIR_AgeStructured_Children_V3()
%%  Project - Age Structured Cholera Model
%  Author: John Szijjarto
%  Date:   09/17/12
%  Title:  Study the Effects of Cholera on Children Using Gobbert's Model
%           First Order Accurate Epidemic Cholera Model
%
%  Immunity Waning function is enabled only for people < 1 year old. This
%  makes for more realistic output when the time frame is only 24 weeks.
%  In other words the time frame is much less than the 1 year required
%  for humans to begin loosing immunity to the current outbreak. This
%  prevents newly recovered humans from unrealistically getting reinfected
%  within the 24 week timeframe.
%
%
%% %%%%%% Matlab Environment Commands  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
clear all;  % Close/Delete all figures
close all;  % Free system memory
clc;        % Clear command window

if ~exist('./temp', 'dir')  % create './temp' dir if doesn't already exist
  mkdir('./temp');
end
outfile_1 = fopen('./temp/output.txt','w');

%% %%%%%% Simulation Control Variables  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

reference_sim = false;     % if true, run ref sim (ie. no infected pop)
xi            = 1000000000; % Hi Rate of shedding of cholera vibrios from
%xi           = 100;        % Lo Rate of shedding of cholera vibrios from
                           % infected human of age a.

%% %%%%%% Model Variables %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

k     = 1/50;           % detla_t = 1/50 of a week
h     = 1;              % delta_a = 1 week
alpha = 1;              % Proportionality factor (wave speed)
b     = (1/50)*(1/52);  % normal mortality rate in deaths per week

% Population Parameters
N  = 10000;            % Population Sice

% Declare the S, I, R Arrays and zero out all values.
%
m   = 3744;          % cols -> total age = 72 * 52 = 3744 weeks
n   = 24/k;          % rows -> total sim time = 24 weeks or ~6 months

S   = zeros(m,n);  % Suseptable population
I   = zeros(m,n);  % Infected population
R   = zeros(m,n);  % Recovered population
BL  = zeros(n,1);  % Less-infective vibrio population
BH  = zeros(n,1);  % highly-infective vibrio population

total_I_mortalities_adults   = zeros(n,1);
total_I_mortalities_children = zeros(n,1);

total_sus_population_adults = zeros(n,1); % col vect of length n
```

```
total_inf_population_adults = zeros(n,1); % col vect of length n
total_rec_population_adults = zeros(n,1); % col vect of length n
total_population_adults     = zeros(n,1); % col vect of length n

total_sus_population_children = zeros(n,1); % col vect of length n
total_inf_population_children = zeros(n,1); % col vect of length n
total_rec_population_children = zeros(n,1); % col vect of length n
total_population_children     = zeros(n,1); % col vect of length n

percent_I_mortalities_children = zeros(n,1);
percent_I_mortalities_adults   = zeros(n,1);
percent_I_children             = zeros(n,1);
percent_I_adults               = zeros(n,1);


A        = 72;            % Upper bound on human age : 72 years
kappa_L  = 1000000;       % Half saturation constant (Less-infective)
kappa_H  = kappa_L/700;   % Half saturation constant (Hyper-infective)
beta_L   = 1.5/7*7;       % Ingestion rate (Less-infective)
beta_H   = 1.5/7*7;       % Ingestion rate (Hyper-infective)

lambda  = 0.0*7;  % Human recruitment rate (non-newborns entering pop)
gamma_1 = 1/5*7;  % recovery rate of untreated cholera
gamma_2 = 1/3*7;  % recovery rate of treated cholera
u       = 0.0;    % antiboitic treatment rate for humans of age a at time t
H2O     = 0.0;    % hydration therapy related mortality. (h(a,t) in model)
                  % This is a percentage value.

delta_L = 1/30*7;   % Death rate of vibrio in the environment.
eta     = 0.1;      % Relative amount of stool per unit time - no units
chi     = 1/5*24*7; % Rate of vibrio moving from HI to non-HI state.

ten_years_old = 520;

%% %%%%% Boundary Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%       Note: Boundary conditions for R array are created
%             within the main simulation loop.
%
for t = 0:(n-1)
    S(0+1,t+1) = 0.0; % Susceptable Population
    I(0+1,t+1) = 0.0; % Infected Population
end


%% %%%%% Initial Conditions %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
eighteen_years_old = 936;   % Age in weeks
nineteen_years_old = 988;   % Age in weeks

% Initial Conditions for I (Infected)
%
if (reference_sim == false) % reference sim contains no infected people
    % Include one 18 year old infected human
    for a=(m-1):-1:0
        if (a>=eighteen_years_old) && (a<nineteen_years_old)
            I(a+1,0+1) = 1/52.0;
        else
            I(a+1,0+1) = 0.0;
        end
    end
end

fifteen_years_old     = 780;  % age in weeks
sixty_five_years_old  = 3380; % age in weeks
seventy_two_years_old = 3744; % age in weeks

% Pop distribution 40.1\% aged 0-14, 56.1\% aged 15-64, and 3.8\% aged 65+.
```

```
N1 = (N*.401)/(15-0)/52;  % #humans < 15 years old per week age segment
N2 = (N*.561)/(65-15)/52; % #humans >= 15 & < 65 years old per week age seg
N3 = (N*.038)/(72-65)/52; % #humans >= 65 & < 72 years old per week age seg

% Initial Conditions for R (Recovered) and S (Susceptible)
one_year_old = 52;   % Age in weeks

for a=(m-1):-1:0
    % Susceptable and Recovered Population
    if (a<one_year_old) % if age<1 year-old then immune (ie. in R group)
        R(a+1,0+1) = N1;
        S(a+1,0+1) = 0.0;
    end
    if (a>=one_year_old && a<fifteen_years_old)
        R(a+1,0+1)  = 0.0;
        S(a+1,0+1) = N1;
    end
    if (a>=fifteen_years_old && a<sixty_five_years_old)
        R(a+1,0+1)  = 0.0;
        S(a+1,0+1) = N2;
    end
    if (a>=sixty_five_years_old && a<seventy_two_years_old)
        R(a+1,0+1)  = 0.0;
        S(a+1,0+1) = N3;
    end
end


%% %%%%% Run the Simulation %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Generate All Other Interior Grid Points
%
births                 = 0;  % not currently utilized
I_mortalities_adults   = 0;  % infected adult mortalities
I_mortalities_children = 0;  % infected children mortalities

for t = 0:(n-1)

    % Suseptable Population - Generate one row
     for a=1:(m-1)
        S(a+1,t+1+1) = ...
          (1-k*alpha/h)* S(a+1,t+1) ...
          + k*alpha/h* S(a-1+1,t+1) ...
          + k*lambda ...                                % recruitment
          - k*beta_L*BL(t+1)/(kappa_L+BL(t+1))*S(a+1,t+1)... % BL infected
          - k*beta_H*BH(t+1)/(kappa_H+BH(t+1))*S(a+1,t+1)... % BH infected
          - k*b * S(a+1,t+1) ...                        % natural mortalities
          + k*omega(a+1,t+1,k)* R(a+1,t+1);         % pop losing immunity
     end

    % Infected Children Adults - Generate one row
     for a=1:ten_years_old
        I(a+1,t+1+1) = ...
          (1-k*alpha/h)* I(a+1,t+1) ...
          + k*alpha/h* I(a-1+1,t+1) ...
          + k*beta_L*BL(t+1)/(kappa_L+BL(t+1))*S(a+1,t+1)... % BL infected
          + k*beta_H*BH(t+1)/(kappa_H+BH(t+1))*S(a+1,t+1)... % BH infected
          - k*b * I(a+1,t+1) ...                        % natural mortalities
          - k*(1-H2O)*delta(a)*I(a+1,t+1) ...       % infected mortalities
          - k*gamma_1*(1-u)*I(a+1,t+1) ... % pop recovering w/o antibiotics
          - k*gamma_2*u*I(a+1,t+1);         % pop recovering with antibiotic
        I_mortalities_children = I_mortalities_children ...
            + k*(1-H2O)*delta(a)*I(a+1,t+1);
     end

     % Infected Population Adults - Generate one row
     for a=(ten_years_old+1):(m-1)
        I(a+1,t+1+1) = ...
```

```
195             (1-k*alpha/h)* I(a+1,t+1) ...
196             + k*alpha/h* I(a-1+1,t+1) ...
197             + k*beta_L*BL(t+1)/(kappa_L+BL(t+1))*S(a+1,t+1)... % BL infected
198             + k*beta_H*BH(t+1)/(kappa_H+BH(t+1))*S(a+1,t+1)... % BH infected
199             - k*b * I(a+1,t+1) ...                    % natural mortalities
200             - k*(1-H2O)*delta(a)*I(a+1,t+1) ...       % infected mortalities
201             - k*gamma_1*(1-u)*I(a+1,t+1) ... % pop recovering w/o antibiotics
202             - k*gamma_2*u*I(a+1,t+1);        % pop recovering with antibiotic
203           I_mortalities_adults = I_mortalities_adults...
204               + k*(1-H2O)*delta(a)*I(a+1,t+1);
205       end
206
207       % Include new born babies into the recovered population this week
208       fecundity = 0;
209       for age=779:1:2339
210           fecundity = fecundity...
211               + (S(age+1,t+1) + I(age+1,t+1) + R(age+1,t+1))...
212               * (1/5) * (sin((age-780)/1560*3.14159))^2;
213       end
214       R(0+1,t+1)= fecundity/52; %Divided by 52 weeks/year
215       births = births + k*fecundity/52;
216
217       % Recovered Population - Generate one row
218       for a=1:(m-1)
219          R(a+1,t+1+1) = ...
220            (1-k*alpha/h)* R(a+1,t+1) ...
221            + k*alpha/h * R(a-1+1,t+1) ...
222            + k*gamma_1*(1-u)*I(a+1,t+1) ... % pop recovering w/o antibiotics
223            + k*gamma_2*u*I(a+1,t+1) ...     % pop recovering with antibiotics
224            - k*b * R(a+1,t+1) ...           % natural mortalities
225            - k*omega(a+1,t+1,k)* R(a+1,t+1);   % recovered - loosing immunity
226       end
227
228       % Calculate current hyperinfective (BH) and non-hyperinfective (BL)
229       % cholera bacteria populations
230       dBH = get_dBH(I,t,BH,xi,chi,eta,A,k,h);
231       dBL = get_dBL(chi,BH,delta_L,BL,t,k,h);
232       BH(t+1+1)  = BH(t+1) + dBH;
233       BL(t+1+1)  = BL(t+1) + dBL;
234
235       total_I_mortalities_adults(t+1) = I_mortalities_adults;
236       total_I_mortalities_children(t+1) = I_mortalities_children;
237
238       if (mod(t,100) == 0)
239           fprintf('t = %d\n', t);   % This is just a program status output
240       end
241    end
242
243   %% %%%%% Output Results %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
244   %
245
246   % Print Population Totals
247   fprintf(outfile_1,...
248   '\nSUSEPTABLE  INFECTED-Adults INFECTED-Children RECOVERED  TOT-POP\r\n');
249   for t = 0:1:(n-1)
250       for a=0:ten_years_old
251           total_sus_population_children(t+1) = ...
252               total_sus_population_children(t+1) + S(a+1,t+1);
253           total_inf_population_children(t+1) = ...
254               total_inf_population_children(t+1) + I(a+1,t+1);
255           total_rec_population_children(t+1) = ...
256               total_rec_population_children(t+1) + R(a+1,t+1);
257           total_population_children(t+1)     = ...
258               total_sus_population_children(t+1) ...
259               + total_inf_population_children(t+1)...
260               + total_rec_population_children(t+1);
261
262           percent_I_children(t+1) = ...
```

```
            total_inf_population_children(t+1)...
            /total_population_children(t+1)*100;
        percent_I_mortalities_children(t+1) = ...
            total_I_mortalities_children(t+1)...
            /total_population_children(t+1)*100;
    end
    for a=(ten_years_old+1):(m-1)
        total_sus_population_adults(t+1) = ...
            total_sus_population_adults(t+1) + S(a+1,t+1);
        total_inf_population_adults(t+1) = ...
            total_inf_population_adults(t+1) + I(a+1,t+1);
        total_rec_population_adults(t+1) = ...
            total_rec_population_adults(t+1) + R(a+1,t+1);
        total_population_adults(t+1)     = ...
            total_sus_population_adults(t+1)...
            + total_inf_population_adults(t+1)...
            + total_rec_population_adults(t+1);
        percent_I_adults(t+1) = ...
            total_inf_population_adults(t+1)...
            /total_population_adults(t+1)*100;
        percent_I_mortalities_adults(t+1) = ...
            total_I_mortalities_adults(t+1)...
            /total_population_adults(t+1)*100;
    end
    fprintf(outfile_1, '%f %f %f %f %f\r\n',...
        total_sus_population_adults(t+1),...
        total_inf_population_adults(t+1),...
        total_inf_population_children(t+1),...
        total_rec_population_adults(t+1),...
        total_population_adults(t+1));
end
fclose(outfile_1);

% Produce 2D Plots
%

x=1:1:n;

figure;
plot(x,total_sus_population_adults(x),'r',...
    x,total_sus_population_children(x),'b');
title('Susceptible Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/SUSCEPTABLE_POPULATION.fig');

figure;
plot(x,total_inf_population_adults(x),'r',...
    x,total_inf_population_children(x),'b');
%axis([0 1200 0 10000]);
title('Infected Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/INFECTED_POPULATION.fig');

figure;
plot(x,total_rec_population_adults(x),'r',...
    x,total_rec_population_children(x),'b');
%axis([0 1200 0 10000]);
title('Recovered Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
```

```
saveas(gcf,'./temp/RECOVERED_POPULATION.fig');

figure;
plot(x,total_population_adults(x),'r',...
    x,total_population_children(x),'b');
title('Total Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/TOTAL_POPULATION.fig');

figure;
plot(x,BH(x));
title('BH Vibrio Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;
saveas(gcf,'./temp/BH_POPULATION.fig');

figure;
plot(x,BL(x));
title('BL Vibrio Population');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
grid;
saveas(gcf,'./temp/BL_POPULATION.fig');

figure;
plot(x,percent_I_adults(x),'r',x,percent_I_children(x),'b');
title('Percentage Infected');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/Percentage_Infected.fig');


figure;
plot(x,percent_I_mortalities_adults(x),'r',...
    x,percent_I_mortalities_children(x),'b');
title('Percentage Infected Mortalities');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/Percentage_Infected_Mortalities.fig');

figure;
plot(x,total_I_mortalities_adults(x),'r',...
    x,total_I_mortalities_children(x),'b');
title('Total Infected Mortalities');
xlabel('Time: 3.36hrs/step - 24 weeks total');
ylabel('Population');
legend('Adults','Children');
grid;
saveas(gcf,'./temp/Total_Infected_Mortalities.fig');


fclose('all');

end % Main Program

%% ****** Sub Functions ***********************************************
% ********************************************************************
%
function db_h = get_dBH(I_,t_,BH_,xi_,chi_,eta_,A_,k_,h_)
% Return change in HI vibrio population
```

```
399      sum = 0;
400      for age = 0:(A_*52-2)  % age is in weeks
401          sum = sum + I_(age+1+1,t_+1) + I_(age+1,t_+1);
402      end;
403      sum = xi_*eta_*h_*(1/2)*sum;
404      dt  = k_;
405      db_h = dt*h_*(sum - chi_*BH_(t_+1));
406  end
407
408  function db_l = get_dBL(chi_,BH_,delta_L_,BL_,t_,k_,h_)
409  % Return change in non-HI vibrio population
410      dt    = k_;
411      db_l  = dt*h_*(chi_*BH_(t_+1) - delta_L_*BL_(t_+1));
412  end
413
414  function  w = omega(a_,t_,k_)
415  % Return age specific rate of waning immunity
416      one_year_old = 52;
417      ten_years_old = 520;        % Age in weeks
418      w = 0;
419      if a_ <= one_year_old
420          w = (1/365); % days
421      elseif ((a_ > ten_years_old) &&(t_>104/k_)) % 10+ lose immunIty in 2yrs
422          w = 1;
423      elseif ((a_ <= ten_years_old) &&(t_>52/k_)) % <10 lose immunity in 1 yr
424          w = 1;
425      end;
426  end
427
428
429  function d = delta(a_)
430  %   Return age specific disease related mortality rate
431      ten_years_old = 520;        % Age in weeks
432      if (a_ <= ten_years_old)
433          d = 0.032;
434      else
435          d = 0.007;
436      end;
437      d = d * 7;  % convert to weeks
438  end
```