MODELLING THE IMPACT OF SEROSORTING
AND SEROADAPTATION ON THE SPREAD OF
HIV IN MEN WHO HAVE SEX WITH MEN

by

Sarah Kok
B.Sc., Simon Fraser University, 2010

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Science

in the
Department of Mathematics
Faculty of Science

© Sarah Kok 2012
SIMON FRASER UNIVERSITY
Summer 2012

All rights reserved.
However, in accordance with the Copyright Act of Canada, this work may be
reproduced without authorization under the conditions for “Fair Dealing.”
Therefore, limited reproduction of this work for the purposes of private study,
research, criticism, review and news reporting is likely to be in accordance
with the law, particularly if cited appropriately.
APPROVAL

Name: Sarah Kok
Degree: Master of Science
Title of Thesis: Modelling the Impact of Serosorting and Seroadaptation on the Spread of HIV in Men who have Sex with Men
Examsining Committee: Dr. Nilima Nigam, Associate Professor
Chair

Dr. Paul Tupper, Associate Professor
Senior Supervisor

Dr. Alexander Rutherford, Adjunct Professor
Supervisor

Dr. Ralf Wittenberg, Associate Professor
Supervisory Committee

Dr. JF Williams, Associate Professor
Internal Examiner

Date Approved: July 6, 2012
Abstract

Serosorting – the practice of choosing sexual partners based on their perceived serostatus – is widely credited as a behavioural intervention that limits the transmission of HIV among men who have sex with men (MSM). However, if this assumption is false, the trend towards serosorting could potentially promote the spread of HIV infection. Here we present a deterministic compartmental model of ordinary differential equations and a subsequent network model of HIV transmission among an MSM population to study the impact of serosorting on HIV incidence and prevalence. Analysis of the compartmental model suggest that serosorting is an effective preventive measure at the population level only once a critical mean time to diagnosis has been achieved. The detrimental impacts of serosorting associated with longer times to diagnosis in the compartmental model are nearly eliminated in the subsequent network model, demonstrating the importance of considering network structure in models of this kind.
I would like to express many thanks to my supervisors, Dr. Paul Tupper and Dr. Alexander Rutherford, for their guidance and support of this work. I also thank Dr. Krisztina Vasarhelyi and everyone involved with the IMPACT-HIV working group, and a special thank you to Dr. Lukas Ahrenberg, who developed the software used for the network modelling simulations. I would also like to express my gratitude to the IRMACS Centre at SFU for their administrative and technical support of this project.
Contents

Approval ii
Abstract iii
Acknowledgments iv
Contents v
List of Figures vii

1 Introduction 1
  1.1 The HIV Epidemic ............................................. 1
    1.1.1 HIV among Men who have Sex with Men ............... 1
  1.2 Serosorting and Seroadaptation .................................. 2
  1.3 Project Overview .................................................. 2
  1.4 Existing Literature ............................................... 3

2 Compartmental Model 5
  2.1 Ensuring Positive-Sized Population Predictions ............... 7
  2.2 Calculating $R_0$ .................................................... 10
  2.3 Fixed Points and their Stability .................................. 11

3 Compartmental Model: Epidemiological Results 21
  3.1 Progression of the Epidemic .................................... 21
  3.2 Disease Prevalence ............................................... 22
  3.3 Instantaneous Incidence .......................................... 24
4 Network Model
  4.1 The Barabási-Albert Model ........................................ 30
  4.2 Process on the Network ............................................. 31
  4.3 Comparison to Compartmental Model ............................... 32
    4.3.1 Mean-Field Approximation ................................... 32
    4.3.2 Network Model Comparison .................................... 34

5 Serosorting on a Network .............................................. 36
  5.1 Method 1: Removing Serodiscordant Links ......................... 37
  5.2 Method 2: Network Rewiring ....................................... 38
    5.2.1 Generalized BA Graph ....................................... 40
    5.2.2 Instantaneous Incidence .................................... 41

6 Conclusion ..................................................................... 43
  6.1 Future Work ................................................................ 45
    6.1.1 Different Methods of Serosorting ............................. 46

Appendix A Model Parameters ........................................... 48

Appendix B Python Code .................................................. 50
  B.1 Configuration File .................................................... 50
  B.2 Module .................................................................... 52

Bibliography .................................................................... 61
List of Figures

2.1 Compartmental Model of an HIV Epidemic in an MSM Population
2.2 A Forward Invariant Region of the Compartmental Model
2.3 Bifurcation Diagram of the Compartmental Model with $\phi = 0.90$
2.4 Bifurcation Diagrams of the Compartmental Model with $\phi = 0.95, 1$
3.1 Percent of the Population in each Compartment over Time
3.2 Steady State HIV Prevalence with respect to $\sigma$ and $\gamma$
3.3 Contour Plot of Steady State HIV Prevalence with respect to $\sigma$
3.4 When Serosorting Increases/Decreases HIV Incidence, assuming $\alpha = 0.5$
3.5 When Serosorting Increases/Decreases HIV Incidence, assuming $\alpha = 0.75, 1$
3.6 When Seroadaptation Increases/Decreases HIV Incidence
4.1 Comparison of Compartmental Model and Mean-Field Approximation HIV Prevalence Predictions
4.2 Comparison of Compartmental Model and Network Model HIV Prevalence Predictions
5.1 A Network Representation of a Population
5.2 Network Model Predictions of Equilibrium HIV Prevalence with respect to $\sigma$
5.3 Process of Network Rewiring
5.4 Qualitative Impact of Serosorting on HIV Incidence
6.1 Compartmental and Network Model Predictions of Equilibrium HIV Prevalence with respect to $\sigma$
Chapter 1

Introduction

1.1 The HIV Epidemic

Human Immunodeficiency Virus (HIV) is a retrovirus that causes Acquired Immunodeficiency Syndrome (AIDS). HIV infection is characterized by the progressive failure of the immune system, predisposing affected individuals to life-threatening opportunistic infections. Left untreated, HIV infection typically progresses to AIDS and death within about 8 to 12 years. HIV is transmitted by exchange of bodily fluids, primarily through unprotected heterosexual or homosexual intercourse, syringe sharing among injection drug users, or from an infected mother to her infant. Since the first reports of unusual illnesses and deaths in 1981, HIV has killed more than 25 million people worldwide [27]. About 0.6% of the world’s entire population is infected with HIV. In Canada, about 65,000 people were living with HIV in 2008 [17]. Treatment of HIV was revolutionized in 1996 with the introduction of highly active antiretroviral therapy (HAART) – a combination of antiretroviral drugs which suppress viral replication, resulting in dramatic improvements in general health and life expectancy for most patients.

1.1.1 HIV among Men who have Sex with Men

Worldwide, men who have sex with men (MSM) continue to be at an elevated risk of HIV infection [6][28][22]. According to a 2009 review of the global epidemic of HIV among MSM [22], this risk group accounts for greatest number of new HIV cases in countries using
CHAPTER 1. INTRODUCTION

traditional surveillance systems. A study of surveillance data in low- and middle-income countries found MSM to be 19.3 times more likely than the general population to be HIV-positive [4]. In Canada, 51% of HIV infections are MSM [17]. Current HIV prevention efforts have been unable to contain or reduce the spread of HIV among MSM [22].

1.2 Serosorting and Seroadaptation

In response to the HIV epidemic, some MSM have adopted the practice of serosorting or seroadaptation to decrease risk of HIV transmission. Serosorting is the practice of restricting unprotected anal intercourse (UAI) to partners of the same perceived serostatus. Seroadaptation is the practice of strategic positioning, where the proportion of UAI acts that are insertive or receptive changes depending on the perceived knowledge of one’s partner’s HIV status.

Serosorting is a common risk reduction strategy [10]. Studies conducted in Vancouver [20] and Atlanta [9] estimate at least 30% of the HIV-negative MSM population practice serosorting. Another survey of self-reported HIV-negative MSM in the United States estimate over 50% practice serosorting, and over 50% practice seroadaptation [12].

1.3 Project Overview

Serosorting may decrease the risk of HIV infection for men who are HIV-negative by reducing the number of sexual contacts they have with men who are diagnosed positive. At the same time, serosorting and seroadaptation may support more frequent and increased risky behaviour with partners who are HIV-positive and undiagnosed, thereby increasing the risk of HIV transmission. As the use of serosorting has already been promoted by some public health and community-based organizations [18], it is essential to be clear on the implications of this practice. The goal of this work is to use mathematical modelling to see how serosorting and seroadaptation impact HIV prevalence and incidence in a hypothetical epidemic among MSM. Prevalence is defined as the total number of individuals who are HIV-positive, while incidence is the number of new HIV cases per unit time.

In Chapter 2, we introduce a compartmental model of the epidemic. We look at some basic
CHAPTER 1. INTRODUCTION

model characteristics including model fixed points and $R_0$ – the number of secondary infections caused by an initial infected individual in an otherwise wholly susceptible population. In Chapter 3 we present some epidemiological results of this model, including the progression of the epidemic, and HIV incidence and equilibrium prevalence predictions.

In Chapter 4, we transfer the model onto a network. This network model and its corresponding mean field approximation are then compared to the previous compartmental model. In Chapter 5, we look at different methods of serosorting on a network, and the resulting prevalence and incidence that these methods predict.

To conclude, in Chapter 6 we mention some implications of these results, as well as describe future work that could be done.

1.4 Existing Literature

Existing literature offers a mixed opinion on the impact of serosorting. Cassels et al. [8] used a deterministic mathematical model of HIV transmission to suggest that serosorting can be an effective harm reduction strategy under realistic amounts of testing frequency and sexual contact rates. The model they developed was a detailed compartmental model of HIV transmission, parameterized with data from a 2003 study of MSM in Seattle. Their result is supported by analysis of data from the EXPLORE cohort, which was a study to test a behavioural intervention in HIV-negative MSM in six U.S. cities from 1999-2003 [15].

Conversely, using a static model which calculated the risk of HIV transmission from MSM who disclosed that they were HIV-positive versus HIV-negative, Butler and Smith [7] concluded that due to the high infectivity during the acute and often undiagnosed phase that occurs at the beginning of an infection, serosorting may cause increased HIV transmission. In another study Wilson et al. [25] used a mathematical model to estimate the relative risk of HIV acquisition associated with serosorting compared to not serosorting, and concluded that serosorting can be expected to increase risk of HIV acquisition in many settings. They estimated the relative risk of serosorting by calculating the probability of forming a relationship with an HIV-positive partner when serosorting does and does not occur, and the corresponding probabilities of HIV transmission.

All of the above mentioned models are deterministic models, and do not consider the impact of network structure. The work presented here differs from and improves upon the existing
literature by incorporating both serosorting and seroadaptation into a deterministic model. By contrast with the static models of Butler and Smith [7], and Wilson et al. [25] this is a dynamic model. This model is subsequently placed onto a network in an effort to capture more accurately how HIV is transmitted in the real world.
Chapter 2

Compartmental Model

Presented here is a compartmental model of an HIV epidemic in a MSM population. The model studies the effect of serosorting and seroadaptation, two different behavioural interventions used by MSM to prevent or decrease the probability of HIV transmission. Serosorting is the practice of choosing only sexual partners perceived to be seroconcordant, that is, partners of the same HIV status. Seroadaptation refers to the change in behaviour based on the knowledge of the HIV status of one’s partner. This is a constant population model with four compartments representing subpopulations of individuals who are susceptible (S), infected but not diagnosed (I), diagnosed (D), and on treatment (T). The total population $N$ is the sum of the populations in these four compartments, $N = S + I + D + T$.

![Diagram of compartmental model](image)

Figure 2.1: A compartmental model of an HIV epidemic in an MSM population. The rate at which individuals in the susceptible (S) class become infected (I) depends on the amount of serosorting and seroadaptation in the population. Infected individuals are diagnosed (D) at rate $\gamma$, and diagnosed individuals start treatment (T) at rate $\lambda$. 


In this model, we are only concerned with sexual partnerships that could potentially result in HIV transmission, which are sexual partnerships between an HIV-negative and HIV-positive partner. Therefore, the model only tracks decisions regarding serosorting and seroadapting that are made by susceptible (HIV-negative) individuals.

All susceptible individuals are assumed to have an average of $C$ sexual acts per year. For each of these $C$ acts, a susceptible individual will serosort with probability $\sigma$, and a fraction $\frac{1}{1+\sigma}$ of these partnerships will be serodiscordant. Individuals who do not serosort will query the status of their partner with probability $\rho$. When not serosorting, the probability of forming a sexual relationship with someone in a specific state is proportional to the number of individuals in that state. For example, the probability of forming a partnership with an undiagnosed, HIV-positive person is $\frac{1}{N}$.

This model uses three different transmission probabilities, $\beta_{ND}$, $\beta_{D}$, and $\beta_{U}$, corresponding to the probability of HIV transmission per act when the susceptible individual falsely believes their partner is HIV-negative (not diagnosed), when the susceptible individual knows their partner is HIV-positive (diagnosed), and when the susceptible individual has not asked the HIV status of their partner (unknown), respectively [26]. The lowest transmission probability is $\beta_{D}$, when the partner is known to be HIV-positive. The probability of transmission is slightly higher when the HIV status of the infected partner is unknown ($\beta_{U}$), and is greatest when the infected partner is not diagnosed ($\beta_{ND}$). The transmission probability per act for individuals who practice serosorting is always $\beta_{ND}$, and the transmission probability per act for individuals who practice neither serosorting nor seroadaptation is always $\beta_{U}$.

Treatment is assumed to be $\phi$-percent effective in reducing infectiousness, resulting in a $(1 - \phi)$ reduction in probability of transmission.

Infected individuals are diagnosed with rate $\gamma$, and diagnosed individuals are put on treatment at rate $\lambda$. Individuals in all compartments have a natural death rate $\mu$, and there are two HIV-related death rates, $d_1$ and $d_2$, corresponding to HIV-positive individuals before and after they start treatment, respectively. All new arrivals into the population are susceptible, and enter the population at a rate equal to the total death rate of all individuals, maintaining a constant population.
The ODEs describing the model discussed above are shown in equations (2.1a) - (2.1d).

\[
\begin{align*}
\frac{dS}{dt} &= \mu N + d_1 (I + D) + d_2 T - \mu S - \sigma \beta_{ND} C \left( \frac{I}{I+S} \right) S \\
&\quad - (1 - \sigma) \rho \left[ \beta_{ND} C \left( \frac{I}{N} \right) S + \beta_{DC} \left( \frac{D + (1 - \phi) T}{N} \right) S \right] \\
&\quad - (1 - \sigma)(1 - \rho) \beta_{UC} \left( \frac{I + D + (1 - \phi) T}{N} \right) S , \\
\frac{dI}{dt} &= \sigma \beta_{ND} C \left( \frac{I}{I+S} \right) S + (1 - \sigma) \rho \left[ \beta_{ND} C \left( \frac{I}{N} \right) S + \beta_{DC} \left( \frac{D + (1 - \phi) T}{N} \right) S \right] \quad (2.1b) \\
&\quad + (1 - \sigma)(1 - \rho) \beta_{UC} \left( \frac{I + D + (1 - \phi) T}{N} \right) S - (\mu + d_1 + \gamma) I , \\
\frac{dD}{dt} &= \gamma I - (\mu + d_1 + \lambda) D , \\
\frac{dT}{dt} &= \lambda D - (\mu + d_2) T . \quad (2.1d)
\end{align*}
\]

Note that the first three terms in equation (2.1a) correspond to all new arrivals into the population.

### 2.1 Ensuring Positive-Sized Population Predictions

Any realistic population model must satisfy the condition that if the system begins in the physical region where the populations of all compartments are greater than or equal to zero, the system will remain in the physical region. That is, we would like to show that starting with realistic initial conditions, the model will never predict a negative population size in any of the compartments. To prove this, we will construct a forward invariant region. A region \( \Lambda \) is forward invariant if \( \vartheta_t(\Lambda) \subset \Lambda \) for all \( t > 0 \), where \( \vartheta_t(\Lambda) \) is the set of all flows at time \( t \) starting in \( \Lambda \) at time 0.

Since equations (2.1a)-(2.1d) describe a constant population model, we only have three independent equations. To avoid working with actual population sizes, we can use proportions. Let

\[
\tilde{S} = \frac{S}{N} , \quad \tilde{I} = \frac{I}{N} , \quad \tilde{D} = \frac{D}{N} , \quad \tilde{T} = \frac{T}{N} ,
\]
so that
\[ \tilde{S} + \tilde{I} + \tilde{D} + \tilde{T} = \tilde{N} = 1. \]  

Using the original model equations (2.1a)-(2.1d) and equation (2.2), we can write a system of three independent equations in variables \( \tilde{I}, \tilde{D}, \text{ and } \tilde{T}. \) These equations are given below:

\[
\begin{align*}
\frac{d\tilde{I}}{dt} & = \sigma \beta_{ND} C \left( \frac{\tilde{I}}{1 - \tilde{D} - \tilde{T}} \right) (1 - \tilde{I} - \tilde{D} - \tilde{T}) \\
& \quad + (1 - \sigma) \rho \beta_{ND} C \cdot \tilde{I} (1 - \tilde{I} - \tilde{D} - \tilde{T}) + (1 - \phi) \beta_{D} C \left( \tilde{D} + (1 - \phi) \tilde{T} \right) (1 - \tilde{I} - \tilde{D} - \tilde{T}) \\
& \quad + (1 - \sigma)(1 - \rho) \beta_{U} C \left( \tilde{I} + \tilde{D} + (1 - \phi) \tilde{T} \right) (1 - \tilde{I} - \tilde{D} - \tilde{T}) - (\mu + d_1 + \gamma) \tilde{I} , \\
\frac{d\tilde{D}}{dt} & = \gamma \tilde{I} - (\mu + d_1 + \lambda) \tilde{D} , \\
\frac{d\tilde{T}}{dt} & = \lambda \tilde{D} - (\mu + d_2) \tilde{T} .
\end{align*}
\]

We now construct a forward invariant region for the model described by equations (2.3a)-(2.3c), chosen to be the 3-simplex defined by \( \tilde{I} + \tilde{D} + \tilde{T} \leq 1 \) and \( \tilde{I}, \tilde{D}, \tilde{T} \geq 0, \) shown in Figure 2.2. We want to show that the vector fields at the boundaries of the region are either zero, tangent to the boundary, or point inwards. This guarantees that no trajectory starting in this region will escape.
CHAPTER 2. COMPARTMENTAL MODEL

Figure 2.2: A forward invariant region of the model described by equations (2.3a)-(2.3c). The vector field on the boundaries of this region is either zero, tangent to the boundary, or pointing inwards. This proves that starting with realistic initial conditions, the model will never predict negative values for $\tilde{I}$, $\tilde{D}$, or $\tilde{T}$.

To show that the vector field on the surface where $\tilde{I} = 0$ (the front-left plane of the simplex) points inwards, we have that,

$$\tilde{I} = 0 \implies \frac{d\tilde{I}}{dt} = (1 - \sigma)\rho \left( \beta_D C (\tilde{D} + (1 - \phi)\tilde{T}) \cdot (1 - \tilde{D} - \tilde{T}) \right) + (1 - \sigma)(1 - \rho)\beta_U C (\tilde{D} + (1 - \phi)\tilde{T})(1 - \tilde{D} - \tilde{T}) \geq 0,$$

since $0 \leq \sigma, \rho, \phi \leq 1$, $\beta_D, \beta_U, C > 0$, and $\tilde{D} + \tilde{T} \leq 1$.

Similarly, to show that the vector field on the surfaces where $\tilde{D} = 0$ and $\tilde{T} = 0$ (the front-right and bottom planes of the simplex, respectively) point inwards, we have that,

$$\tilde{D} = 0 \implies \frac{d\tilde{D}}{dt} = \gamma \tilde{I} \geq 0, \text{ since } \gamma > 0,$$

$$\tilde{T} = 0 \implies \frac{d\tilde{T}}{dt} = \gamma \tilde{I} \geq 0, \text{ since } \gamma > 0,$$
and

\[ \tilde{T} = 0 \implies \frac{d\tilde{T}}{dt} = \lambda \tilde{D} \geq 0, \quad \text{since } \lambda > 0. \]

A direction vector on the surface \( \tilde{I} + \tilde{D} + \tilde{T} = 1 \) (the back plane of the simplex) is

\[
\begin{pmatrix}
\tilde{I}' \\
\tilde{D}' \\
\tilde{T}'
\end{pmatrix} =
\begin{pmatrix}
-(\mu + d_1 + \gamma)\tilde{I} \\
\gamma \tilde{I} - (\mu + d_1 + \lambda)\tilde{D} \\
\lambda \tilde{D} - (\mu + d_2)\tilde{T}
\end{pmatrix}.
\]

If we take the dot product of this vector with the outward pointing vector \((1,1,1)\), we get

\[
\begin{pmatrix}
-(\mu + d_1 + \gamma)\tilde{I} \\
\gamma \tilde{I} - (\mu + d_1 + \lambda)\tilde{D} \\
\lambda \tilde{D} - (\mu + d_2)\tilde{T}
\end{pmatrix} \cdot 
\begin{pmatrix}
1 \\
1 \\
1
\end{pmatrix}
\]

\[
= -(\mu + d_1 + \gamma)\tilde{I} + \gamma \tilde{I} - (\mu + d_1 + \lambda)\tilde{D} + \lambda \tilde{D} - (\mu + d_2)\tilde{T}
\]

\[
= -(\mu + d_1)\tilde{I} - (\mu + d_1)\tilde{D} - (\mu + d_2)\tilde{T}
\]

\[\leq 0.\]

Therefore any direction vector on this surface must be pointing inwards. So, we have shown that the direction field on all four surfaces of the simplex in Figure 2.2 points inwards (or are zero). It can also be shown the direction field on all the edges and vertices of the simplex also points inwards (or are zero). Therefore we have constructed a forward invariant region, and with proper initial conditions, the model described in equations (2.3a)-(2.3c) will never predict a negative population. Additionally, since equations (2.3a)-(2.3c) along with equation (2.2) describe a model equivalent to the model described by (2.1a)-(2.1d), we can draw the same conclusion for the original model.

### 2.2 Calculating \( R_0 \)

One quantity of interest that we can calculate is the basic reproduction number \( R_0 \), the number of secondary infections caused by an initial infected individual in an otherwise
wholly susceptible population. The value of $R_0$ provides information on the stability of the disease-free fixed point. When $R_0 < 1$, the disease-free fixed point is stable as the epidemic is unable to sustain itself. Conversely, if $R_0 > 1$ a single infection will lead to multiple others, spurring an epidemic and making the disease-free fixed point unstable.

The quantity $R_0$ can be expressed as,

$$R_0 = \frac{\beta_N D C + (1 - \sigma)\rho \beta_N D C + (1 - \sigma)(1 - \rho)\beta_U C}{(\mu + d_1 + \gamma)} \times \frac{(1 - \sigma)\rho \beta_D C + (1 - \sigma)(1 - \rho)\beta_U C}{(\mu + d_1 + \lambda)} \times \frac{(1 - \sigma)\rho \beta_D C (1 - \phi) + (1 - \sigma)(1 - \rho)\beta_U C (1 - \phi)}{(\mu + d_2)}.$$

The first term in (2.4) is the expected number of secondary infections caused by the initial infected individual while they are undiagnosed. This term is the product of the expected number of infections caused per unit time, and the average time spent undiagnosed. The expected number of infections caused per unit time is found using (2.1b) with $I = 1$, $D = T = 0$, and $S = N - 1 \approx N$ for $N$ large. Individuals leave compartment $I$ if they die from natural or HIV-related causes, or if they are diagnosed. Thus, the average time spent undiagnosed is $\frac{1}{\mu + d + \gamma}$.

The second term in (2.4) is the probability that the infected person becomes diagnosed before he dies from natural or HIV-related causes, multiplied by the expected number of infections caused after he is diagnosed and before he is put on treatment. The expected number of infections caused while diagnosed is found in a method similar as previously described, with the only difference being that now $D = 1$ and $I = T = 0$. Similarly, the third term is the probability of becoming diagnosed and starting treatment before he dies from natural or HIV-related causes, multiplied by the expected number of infections caused while he is on treatment.

### 2.3 Fixed Points and their Stability

An HIV epidemic is a widespread outbreak of infections affecting many members of a population simultaneously. When HIV remains constantly present in a population, that population is said to be HIV endemic.
Setting equations (2.1a)-(2.1d) to zero, we find the system has two fixed points \((S^*, I^*, D^*, T^*)\). There is the trivial, disease-free equilibrium (DFE) where \(I^* = D^* = T^* = 0\), and an endemic fixed point.

We now prove there is local asymptotic stability of the DFE if and only if \(R_0 < 1\). In this proof we use the equivalent system of three independent equations in variables \(\tilde{I}, \tilde{D}\) and \(\tilde{T}\) described in Equations (2.3a)-(2.3c).

The Jacobian of this system evaluated at the DFE is

\[
J_{DFE} = \begin{pmatrix}
D_1 & D_2 & D_3 \\
\gamma & -y_2 & 0 \\
0 & \lambda & -y_3
\end{pmatrix},
\]

where

\[
y_1 = \mu + d_1 + \gamma,
\]
\[
y_2 = \mu + d_1 + \lambda,
\]
\[
y_3 = \mu + d_2,
\]

and

\[
D_1 = \sigma \beta_{ND}C + (1 - \sigma)\rho \beta_{ND}C + (1 - \sigma)(1 - \rho)\beta_UC - y_1,
\]
\[
D_2 = (1 - \sigma)\rho \beta_{D}C + (1 - \sigma)(1 - \rho)\beta_UC,
\]
\[
D_3 = (1 - \sigma)\rho \beta_{D}C(1 - \phi) + (1 - \sigma)(1 - \rho)\beta_UC(1 - \phi).
\]

Note that using the above definitions, \(R_0\) can be rewritten as

\[
R_0 = \frac{D_1 + y_1}{y_1} + \frac{\gamma}{y_1} \cdot \frac{D_2}{y_2} + \frac{\gamma \lambda}{y_1 y_2} \cdot \frac{D_3}{y_3},
\]

(2.5)
CHAPTER 2. COMPARTMENTAL MODEL

The corresponding characteristic equation \(|J_{DFE} - xI| = 0\) can be expressed as

\[
0 = \begin{vmatrix}
D_1 - x & D_2 & D_3 \\
\gamma & -y_2 - x & 0 \\
0 & \lambda & -y_3 - x \\
\end{vmatrix} - \gamma \begin{vmatrix}
D_2 & D_3 \\
\lambda & -y_3 - x \\
\end{vmatrix}
\]

or equivalently,

\[
0 = x^3 - [D_1 - y_2 - y_3]x^2 + [-D_1y_2 - D_1y_3 + y_2y_3 - \gamma D_2]x \\
- [D_1y_2y_3 + \gamma D_2y_3 + \gamma \lambda D_3].
\]

The Routh-Hurwitz Criterion [3] for cubic polynomials states that all three complex roots of a cubic polynomial \(x^3 - ax^2 + bx - c = 0\) have strictly negative real parts if and only if

\[
\begin{align*}
a &< 0 \\
c &< 0, \text{ and} \\
ab &< c.
\end{align*}
\]

Applying this criterion to the characteristic polynomial of \(J_{DFE}\), so that

\[
\begin{align*}
a &= D_1 - y_2 - y_3, \\
b &= -D_1y_2 - D_1y_3 + y_2y_3 - \gamma D_2, \\
c &= D_1y_2y_3 + \gamma D_2y_3 + \gamma \lambda D_3,
\end{align*}
\]

we will now show that all three conditions (2.6a)-(2.6c) hold if and only if \(R_0 < 1\).
Since $\gamma, \lambda, y_2, y_3, D_2, D_3 > 0$, from (2.5) we know,

\[
R_0 < 1 \implies D_1 < 0 \\
\implies D_1 - y_2 - y_3 < 0 \\
\implies a < 0.
\]

Additionally, from (2.5) we know,

\[
R_0 < 1 \iff D_1 y_2 y_3 + \gamma D_2 y_3 + \gamma \lambda D_3 < 0 \\
\iff c < 0.
\]

It remains to show condition (2.6c). To do this, first let

\[
E_1 = \sigma \beta_{ND} + (1 - \sigma)\rho \beta_{ND} + (1 - \sigma)(1 - \rho)\beta_U \\
E_2 = (1 - \sigma)(\rho \beta_D + (1 - \rho)\beta_U) \\
E_3 = (1 - \sigma)(\rho \beta_D + (1 - \rho)\beta_U)(1 - \phi).
\]

Then

\[
R_0 = C \left[ \frac{E_1}{y_1} + \frac{\gamma E_2}{y_1 y_2 y_3} + \frac{\gamma \lambda E_3}{y_1 y_2 y_3} \right] \\
= C \left[ \frac{y_2 y_3 E_1 + y_3 \gamma E_2 + \gamma \lambda E_3}{y_1 y_2 y_3} \right],
\]

and

\[
R_0 < 1 \implies C < \frac{y_1 y_2 y_3}{y_2 y_3 E_1 + y_3 \gamma E_2 + \gamma \lambda E_3} = C_R.
\]

Now consider the function $F(C) = ab - c$, which is a quadratic function in $C$. To prove condition (2.6c) we show that the maximum value of $F(C)$ in the interval $0 < C < C_R$ is negative. We show this with the following five arguments.

1. The function $F(C)$ is negative on the left endpoint of the interval:

\[
F(0) = -(y_1 + y_2 + y_3)(y_1 y_2 + y_1 y_3 + y_2 y_3) + y_1 y_2 y_3 \\
= -(y_1 + y_2)(y_1 y_2 + y_1 y_3 + y_2 y_3) - y_3(y_1 y_3 + y_2 y_3) \\
< 0.
\]
2. The function $F(C)$ is negative on the right endpoint of the interval. Note that on the right endpoint of the interval $R_0 = 1$. Using Equation (2.5) it is straightforward to show that $a < 0$ and $c = 0$ when $R_0 = 1$. Thus to prove $F(C_R) = ab - c < 0$, it is sufficient to prove that $b > 0$ when $C = C_R$.

$$b = -D_1y_2 - D_1y_3 + y_2y_3 - \gamma D_2$$
$$= -(C_R E_1 - y_1)y_2 - (C_R E_1 - y_1)y_3 + y_2y_3 - \gamma C_R E_2$$
$$= -C_R E_1 y_2 + y_1 y_2 - C_R E_1 y_3 + y_1 y_3 + y_2 y_3 - \gamma C_R E_2$$
$$= -C_R(E_1 y_2 + E_1 y_3 + \gamma E_2) + y_1 y_2 + y_1 y_3 + y_2 y_3$$
$$= -(E_1 y_2 + E_1 y_3 + \gamma E_2) \cdot \frac{y_2 y_3 E_1 + y_3 y_2 E_2 + \gamma y_2 y_3 E_3}{y_2 y_3 E_1 + y_3 y_2 E_2 + \gamma y_2 y_3 E_3} + y_1 y_2 + y_1 y_3 + y_2 y_3$$
$$= y_1 y_2 \left(\frac{y_3 \gamma E_2 + \gamma \lambda E_3}{y_2 y_3 E_1 + y_3 y_2 E_2 + \gamma \lambda E_3}\right) + y_1 y_3 \left(\frac{y_3 \gamma E_2 + \gamma \lambda E_3}{y_2 y_3 E_1 + y_3 y_2 E_2 + \gamma \lambda E_3}\right)$$
$$- y_1 y_2 \left(\frac{\gamma \lambda E_3}{y_2 y_3 E_1 + y_3 y_2 E_2 + \gamma \lambda E_3}\right) + y_2 y_3$$
$$> 0,$$

and so $F(C_R) < 0$.

3. The function $F(C)$ has positive derivative at the left endpoint of the interval:

$$F'(0) = E_1(y_1 y_2 + y_1 y_3 + y_2 y_3) + (y_1 + y_2 + y_3)(E_1 y_2 + E_1 y_3 + \gamma E_2)$$
$$- E_1 y_2 y_3 - \gamma E_2 y_3 - \gamma \lambda E_3$$
$$= E_1(y_1 y_2 + y_1 y_3) + (y_1 + y_2)(E_1 y_2 + E_1 y_3 + \gamma E_2) + y_3(E_1 y_2 + E_1 y_3)$$
$$- \gamma \lambda E_3$$
$$> E_1(y_1 y_2 + y_1 y_3) + y_1(E_1 y_2 + E_1 y_3 + \gamma E_2) + y_2(E_1 y_2 + E_1 y_3)$$
$$+ y_3(E_1 y_2 + E_1 y_3)$$
$$> 0.$$
4. The function $F(C)$ is concave down:

$$F''(C) = E_1(-E_1 y_2 - E_1 y_3 - \gamma E_2) + E_1(-E_1 y_2 - E_1 y_3 - \gamma E_2)$$
$$= -2E_1(E_1 y_2 + E_1 y_3 + \gamma E_2)$$
$$< 0.$$ 

5. The function $F(C)$ does not reach its maximum in the interval $[0, C_R]$. To prove this, we show $F'(C^*) = 0 \implies C^* > C_R$.

$$F'(C) = E_1(-D_1 y_2 - D_1 y_3 + y_2 y_3 - \gamma D_2) + (D_1 - y_2 - y_3)(-E_1 y_2 - E_1 y_3 - \gamma E_2)$$
$$- E_1 y_2 y_3 - \gamma E_2 y_3 - \gamma \lambda E_3$$
$$= E_1(-(CE_1 - y_1) y_2 - (CE_1 - y_1) y_3 + y_2 y_3 - \gamma D_2)$$
$$+ (CE_1 - y_1 - y_2 - y_3)(-E_1 y_2 - E_1 y_3 - \gamma E_2) - E_1 y_2 y_3 - \gamma E_2 y_3 - \gamma \lambda E_3$$
$$= E_1(-CE_1 y_2 + y_1 y_3 - CE_1 y_3 + y_1 y_3 + y_2 y_3 - \gamma CE_2)$$
$$- CE_1(E_1 y_2 + E_1 y_3 + \gamma E_2) + (y_1 + y_2 + y_3)(E_1 y_2 + E_1 y_3 + \gamma E_2)$$
$$- E_1 y_2 y_3 - \gamma E_2 y_3 - \gamma \lambda E_3$$
$$= 2C[-E_1^2 y_2 - E_1^2 y_3 - \gamma E_1 E_2] + E_1(y_1 y_2 + y_1 y_3 + y_2 y_3)$$
$$+ (y_1 + y_2 + y_3)(E_1 y_2 + E_1 y_3 + \gamma E_2) - E_1 y_2 y_3 - \gamma E_2 y_3 - \gamma \lambda E_3$$
$$= 2C[-E_1^2 y_2 - E_1^2 y_3 - \gamma E_1 E_2] + E_1(y_1 y_2 + y_1 y_3 + y_2 y_3)$$
$$+ (y_1 + y_2)(E_1 y_2 + E_1 y_3 + \gamma E_2) + E_1 y_3^2 - \gamma \lambda E_3$$

Therefore $F'(C^*) = 0 \implies C^* = \frac{C_1^*}{C_2^*}$, where

$$C_1^* = E_1(y_1 y_2 + y_1 y_3 + y_2 y_3) + (y_1 + y_2)(E_1 y_2 + E_1 y_3 + \gamma E_2) + E_1 y_3^2 - \gamma \lambda E_3$$
$$C_2^* = 2(E_1 y_2 + E_1 y_3 + \gamma E_1 E_2).$$

Now to show that $C^*$ is outside of the interval $[0, C_R]$ we want to show that $C^* > C_R$. Note that

$$C^* > C_R \iff \frac{C_1^*}{C_2^*} > \frac{y_1 y_2 y_3}{y_2 y_3 E_1 + y_3 \gamma E_2 + \gamma \lambda E_3}$$
$$\iff C_1^*(y_2 y_3 E_1 + y_3 \gamma E_2 + \gamma \lambda E_3) - C_2^* (y_1 y_2 y_3) > 0.$$
Finally, we have,

\[ C_1^*(y_2y_3E_1 + y_3\gamma E_2 + \gamma \lambda E_3) - C_2^*(y_1y_2y_3) \]

\[ = \left[ \left( (E_1(y_1y_2 + y_1y_3 + y_2y_3) + (y_1 + y_2)(E_1y_2 + E_1y_3 + \gamma E_2) + E_1y_3^2 - \gamma \lambda E_3) \right) \right. \]

\[ \times \left. [y_2y_3E_1 + y_3\gamma E_2 + \gamma \lambda E_3] - 2(E_1^2y_2 + E_1^2y_3 + \gamma E_1E_2)(y_1y_2y_3) \right] \]

\[ = \left[ \left( (E_1y_2y_3 + y_2(E_1y_2 + E_1y_3 + \gamma E_2) + E_1y_3^2 - \gamma \lambda E_3) \right) \right. \]

\[ \times \left. [y_2y_3E_1 + y_3\gamma E_2 + \gamma \lambda E_3] + \gamma \lambda E_1E_3y_2y_2 \right] \]

\[ + \left[ (E_1(y_1y_2 + y_1y_3) + y_1(E_1y_3 + \gamma E_2)) \times [y_3\gamma E_2 + \gamma \lambda E_3] \right] \]

\[ > \left[ \left( (E_1y_2y_3 + y_2(E_1y_2 + E_1y_3) + E_1y_3^2) \right) \right. \]

\[ \times \left. [y_2y_3E_1 + y_3\gamma E_2 + \gamma \lambda E_3] + \gamma \lambda E_1E_3y_2y_2 \right] \]

\[ + \left[ (E_1(y_1y_2 + y_1y_3) + y_1(E_1y_3 + \gamma E_2)) \times [y_3\gamma E_2 + \gamma \lambda E_3] \right] \]

\[ > 0. \]

Therefore conditions (2.6a)-(2.6c) hold if and only if \( R_0 < 1 \), and so the DFE is stable if and only if \( R_0 < 1 \). Note that the only if part of the proof comes from the proof of criterion (2.6b). Numerical results in Maple confirm that a transcritical bifurcation occurs at \( R_0 = 1 \). When \( R_0 > 1 \) the endemic fixed point is stable.

Results from the stability analysis can be presented in the form of a bifurcation diagram. The bifurcation diagram shown in Figure 2.3 is a surface in the \((\rho, \sigma, \gamma)\)-parameter space which separates the regions where the DFE and endemic fixed points are stable.
CHAPTER 2. COMPARTMENTAL MODEL

Figure 2.3: A bifurcation diagram of the model described in equations (2.1a) - (2.1d). The diagram shows in which regions of the \((\rho, \sigma, \gamma)\)-parameter space the DFE and endemic fixed point are stable. The region below the surface corresponds to a stable DFE. Treatment is assumed to reduce the probability of transmission by 90%. All other parameter values are listed in Appendix A.

In Figure 2.3 we see that the amount of seroadaptation \((\rho)\) has little effect on determining which fixed point is stable. The model is, however, sensitive to changes in both the diagnosis rate \((\gamma)\) and the amount of serosorting in the population \((\sigma)\). Stability of the DFE requires some minimum \(\gamma_{\text{min}}\)-value such that if diagnosis takes longer than \(\frac{1}{\gamma_{\text{min}}}\) years, no amount of serosorting will be able to completely wipe out the endemic. As you decrease the average time until diagnosis by increasing \(\gamma\), smaller amounts of serosorting are needed to obtain a stable DFE.

Increasing the effectiveness of treatment eliminates the relationship between \(\gamma\) and \(\sigma\), and increases the range of parameters in which the DFE is stable (Figure 2.4). When treatment is highly effective in preventing transmission, serosorting has very little impact on which fixed point will be stable. In this scenario, the best way to obtain a steady DFE is to increase the diagnosis rate to minimize the time between infection and initiation of treatment. Figure 2.4 below can be compared to Figure 2.3 to see the impact of treatment efficiency on the stability of the fixed points.
Serosorting and seroadaptation start to become counterproductive when $\phi \approx 0.92$. As treatment becomes more effective, lower amounts of serosorting and seroadaptation increase the likelihood of a disease free equilibrium. This is because as treatment becomes highly effective, you are less likely to become infected with HIV by choosing partners who are known to be HIV-positive and on treatment, rather than taking a risk with someone who is assumed to be HIV-negative.
Note that the parameter $\phi$ represents treatment efficiency, but can also be used to capture treatment retention and adherence. For example, decreasing $\phi$ would decrease the effectiveness of treatment in reducing HIV transmission, which could be attributed to the biological properties of HAART or to a non-optimal adherence regimen.
Chapter 3

Compartmental Model: Epidemiological Results

The compartmental model described in (2.1a)-(2.1d) can be used to find some valuable epidemiological results. In Section 3.1 we start by looking at the progression of the epidemic over time. Sections 3.2 and 3.3 look at HIV prevalence and incidence predictions. All analyses of HIV prevalence are equilibrium analyses, and all HIV incidence analyses are non-equilibrium. Unless otherwise noted, the value of all model parameters are listed in Appendix A.

3.1 Progression of the Epidemic

The first diagnosis of AIDS occurred in 1981 [16]. The current treatment of HIV, highly active antiretroviral therapy (HAART), was introduced in 1996. We can use these dates to compare the progression of the disease over time with model predictions.

To see how long the model predicts it will take to reach an equilibrium state, we start with one index case of HIV in 1981. Then we let the epidemic run for 15 years without any diagnosis or treatment by setting the diagnosis rate $\gamma$ much greater than the death rate $\mu$, so $\gamma \gg \mu$. After 15 years, we turn on diagnosis and treatment. Turning off diagnosis and treatment for 15 years is a simplifying assumption made in the model, as there was diagnosis and early forms of treatment, such as azidothymidine (AZT), before 1996.
CHAPTER 3. COMPARTMENTAL MODEL: EPIDEMIOLOGICAL RESULTS

To quantify how close HIV prevalence is to equilibrium, we can calculate what percent the current susceptible population is of the equilibrium susceptible population. Note that initially, the susceptible population decreases towards its equilibrium value. With no serosorting, the population is at 124% of equilibrium after 30 years, 115% after 40 years, and 105% after 50 years. In Figure 3.1 we can see that after 50 years the susceptible population dips below the equilibrium value for a period of time, but never falls below 97% of the equilibrium value.

Figure 3.1: The percent of the population which is susceptible (blue), infected (green), diagnosed (red) and on treatment (teal) over time. Plot (a) corresponds to the beginning of the epidemic that started with a single index case of HIV. After 16 years, diagnosis and treatment are turned on, shown in plot (b). In these images we assume a mean time to diagnosis of 4 years, no serosorting ($\sigma = 0$) and 100% seroadaptation ($\rho = 1$). All other parameter values are given in Appendix A.

Higher levels of serosorting correspond to a longer time to equilibrium. If 50% of the population serosorts ($\sigma = 0.5$), then the population is at 136%, 127%, and 117% of equilibrium after 30, 40 and 50 years, respectively. With 100% serosorting ($\sigma = 1$), it takes between 85 and 90 years to reach 130% of equilibrium.

### 3.2 Disease Prevalence

One quantity of interest is the number of individuals who are HIV-positive when the population is at endemic equilibrium. These results can be illustrated using prevalence plots. Figure 3.2 shows the total number of HIV-positive individuals at the endemic steady state, as a function of the level of serosorting ($\sigma$) and the diagnosis rate ($\gamma$) in the population.
CHAPTER 3. COMPARTMENTAL MODEL: EPIDEMIOLOGICAL RESULTS

Figure 3.2: Steady state HIV prevalence with respect to the amount of serosorting ($\sigma$) and the diagnosis rate ($\gamma$) in a population assuming 100% seroadaptation. Treatment efficiency $\phi$ is taken to be 95%, and values for all other model parameters are given in Appendix A.

In Figure 3.2 we see that there is some critical rate of diagnosis $\gamma_c$ that determines whether or not serosorting is an effective behavioural intervention in limiting the spread of HIV. When $\gamma > \gamma_c$ higher rates of serosorting correspond to lower equilibrium prevalence, and for $\gamma < \gamma_c$, higher rates of serosorting correspond to higher equilibrium prevalence. The critical time until diagnosis is estimated to be between 3.5 and 3.75 years. Another way to view this result is presented in Figure 3.3. Here, instead of looking at HIV prevalence as a function of time to diagnosis ($\gamma$) and the amount of serosorting ($\sigma$), prevalence is given as a function of $\sigma$ for varying values of $\gamma$.

Figure 3.3: Steady state HIV prevalence with respect to the amount of serosorting ($\sigma$) in a population assuming 100% seroadaptation. The different curves represent varying times to diagnosis.
3.3 Instantaneous Incidence

Another useful epidemiological quantity is HIV incidence – the number of new HIV cases per unit time. Incidence is a better measure than prevalence when considering the risk of an individual being infected in a given time period, as a population can have high prevalence with a very low incidence, or a high incidence with a relatively low prevalence.

We can find out how serosorting and seroadaptation impact incidence by calculating instantaneous incidence with respect to both $\sigma$ and $\rho$, respectively. Let $Z$ represent incidence,

$$ Z = \text{Number of new HIV cases per year.} $$

Now we can use the rate of change of infected individuals, equation (2.1b), to find that

$$ Z = \sigma\beta_{ND}C \left( \frac{I}{I+S} \right) S + (1-\sigma)\rho \left[ \beta_{ND}C \left( \frac{I}{N} \right) S + \beta_{DC} \left( \frac{D+(1-\phi)T}{N} \right) S \right] $$

$$ + (1-\sigma)(1-\rho)\beta_{UC} \left( \frac{I+D+(1-\phi)T}{N} \right) S. $$

(3.1)

Note that since $Z$ is linear in $\sigma$, incidence is minimized by either $\sigma = 0$ or $\sigma = 1$. Now we can calculate the rate of change of incidence with respect to serosorting from (3.1) as

$$ \frac{dZ}{d\sigma} = \frac{SC}{N} \cdot \left( \frac{N}{I+S} \beta_{ND} - \rho\beta_{ND} - (1-\rho)\beta_{U} \right) I - [\rho\beta_{D} + (1-\rho)\beta_{U}] D $$

$$ - [\rho\beta_{D} + (1-\rho)\beta_{U}] (1-\phi)T \right) $$

$$ \times \frac{1}{N} \cdot \left[ \frac{N}{I+S} \beta_{ND} - \rho\beta_{ND} - (1-\rho)\beta_{U} \right] I - [\rho\beta_{D} + (1-\rho)\beta_{U}] D $$

$$ - [\rho\beta_{D} + (1-\rho)\beta_{U}] (1-\phi)T. $$

(3.2)

When $dZ/d\sigma < 0$, serosorting decreases HIV incidence. Conversely, when $dZ/d\sigma > 0$, increased levels of serosorting have a negative impact on HIV incidence. So, we are interested at the point at which (3.2) changes sign. This depends on the size of the compartments $S, I, D$ and $T$. To avoid working with actual population sizes, we can again use proportions, so that

$$ \tilde{S} + \tilde{I} + \tilde{D} + \tilde{T} = \tilde{N} = 1. $$
Thus equation (3.2) can be written as

\[
\frac{dZ}{d\sigma} \propto \left[ \frac{1}{I+S} \beta_{ND} - \rho \beta_{ND} - (1 - \rho) \beta_U \right] \bar{I} - [\rho \beta_D + (1 - \rho) \beta_U] \bar{D} - [\rho \beta_D + (1 - \rho) \beta_U] (1 - \phi) \bar{T} .
\]

(3.3)

We would like to see how serosorting influences incidence with respect to the total prevalence of HIV in the population \(P\) and the fraction of the HIV-positive population that is diagnosed \(F\), where

\[
P = \bar{I} + \bar{D} + \bar{T}
\]

\[
F = \frac{\bar{D} + \bar{T}}{\bar{I} + \bar{D} + \bar{T}} .
\]

Therefore the compartment proportions can be expressed as

\[
\bar{S} = 1 - P
\]

\[
\bar{I} = (1 - F) \cdot P
\]

\[
\bar{D} = F \cdot P - \bar{T}
\]

\[
\bar{T} = \alpha F \cdot P ,
\]

where \(\alpha\) is the fraction of the diagnosed population that is on treatment. Equation (3.3) can now be rewritten as

\[
\frac{dZ}{d\sigma} \propto \left[ \frac{1}{(1 - F \cdot P)} \beta_{ND} - \rho \beta_{ND} - (1 - \rho) \beta_U \right] \cdot P \left[ 1 - F \right] - [\rho \beta_D + (1 - \rho) \beta_U] (1 - \phi) F \cdot P .
\]

(3.4)

In Figure 3.4 we see the curve \(\frac{dZ}{d\sigma} = 0\). The three different curves correspond to 0\%, 50\% and 100\% seroadaptation. The area above the curves is the region where \(\frac{dZ}{d\sigma} < 0\), which is where serosorting decreases HIV incidence.
CHAPTER 3. COMPARTMENTAL MODEL: EPIDEMIOLOGICAL RESULTS

Figure 3.4: The curve $\frac{dS}{d\sigma} = 0$ with respect to total prevalence of HIV ($P$) and the fraction of the HIV-positive population that is diagnosed ($F$) assuming no seroadaptation (teal curve), 50% seroadaptation (blue curve) and 80% seroadaptation (black curve). The area above the curves is the region where serosorting decreases HIV incidence. We assume that 50% of the diagnosed population is on treatment ($\alpha = 0.5$) and treatment efficiency $\phi$ is 95%. All other parameter values are given in Appendix A.

From Figure 3.4 we see that if total prevalence of HIV is fixed, increasing the number of diagnosed individuals increases the likelihood that serosorting has a beneficial impact on HIV incidence, as expected. Additionally, if total prevalence increases while the fraction of the HIV-positive population that is diagnosed remains constant, then serosorting is more likely to increase incidence. This is because as the prevalence of HIV increases, the number of susceptible individuals decreases while the number of undiagnosed HIV-positive individuals either increases or remains constant. Since an individual who serosorts is assumed to form a serodiscordant relationship with probability $\frac{I}{1+S}$, increasing prevalence increases the chance of forming serodiscordant relationships and thus increases the probability of transmitting HIV. Finally, we note that increased seroadaptation increases the benefit of serosorting.

In Figure 3.4 we assume that 50% of the diagnosed population is on treatment. In Figure 3.5 we vary this assumption, and compare scenarios of 75% and 100% of the diagnosed population on treatment. All these scenarios assume 95% treatment efficiency ($\phi = 0.95$).
Figure 3.5: The curve $\frac{dZ}{d\sigma} = 0$ with respect to total prevalence of HIV ($P$) and the fraction of the HIV-positive population that is diagnosed ($F$) assuming (a) 75% and (b) 100% of the diagnosed population is on treatment. The area above the curves is the region where serosorting decreases HIV incidence. The different coloured curves correspond to different levels of seroadaptation (the teal, blue and black curves correspond to 0%, 50% and 80% seroadaptation, respectively). Treatment efficiency $\phi$ is assumed to be 95%.

In Figures 3.4 and 3.5 we see that increasing treatment coverage makes serosorting more likely to increase HIV incidence. As a greater proportion of the diagnosed population is put on treatment, it becomes safer to choose partners who are known to be HIV-positive and on treatment, rather than serosort and take a risk with someone who is assumed to be HIV-negative. Recall that a similar result was found with respect to the stability of the disease free equilibrium when we increased treatment efficiency.
We can perform similar calculations as above to look at the rate of change of incidence with respect to seroadaptation. From (3.1) we have that

\[
\frac{dZ}{d\rho} = \frac{(1 - \sigma)CS}{N} \cdot \left[ \beta_{ND}I + \beta_D(D + (1 - \phi)T) - \beta_U(I + D + (1 - \phi)T) \right]
\]

\[
= (1 - \sigma)CS \cdot \left[ (\beta_{ND} - \beta_U)I - (\beta_U - \beta_D)\bar{T} - (\beta_U - \beta_D)(1 - \phi)\bar{T} \right]
\]

\[
= (1 - \sigma)CS \cdot \left[ (\beta_{ND} - \beta_U)(1 - F) \cdot P - (\beta_U - \beta_D)(1 - \alpha)F \cdot P \right.
\]

\[
- (1 - \phi)(\beta_U - \beta_D)\alpha F \cdot P \left. \right]
\]

\[
= (1 - \sigma)CS \cdot P \cdot \left[ (\beta_{ND} - \beta_U) + (\beta_D - \beta_{ND} + \beta_U\phi_\alpha - \beta_D\phi_\alpha) \cdot F \right]
\]

\[
\propto (\beta_{ND} - \beta_U) + (\beta_D - \beta_{ND} + \beta_U\phi_\alpha - \beta_D\phi_\alpha) \cdot F \tag{3.5}
\]

To see if seroadaptation increases or decreases incidence, we are interested at the point where \(\frac{dZ}{d\rho} = 0\). If \(\sigma = 1\) then everyone in the population serosorts, and seroadaptation will have no impact on incidence. Assuming that HIV prevalence is strictly between 0% and 100%, we can use equation (3.5) to determine when seroadaptation has a positive impact on incidence. From (3.5), this depends on the fraction of the HIV-positive population that is diagnosed \((F)\), and the proportion of the diagnosed population that is on treatment \((\alpha)\).

Using (3.5) we can express \(F\) as a function of \(\alpha\) when \(\frac{dZ}{d\rho} = 0\) as follows,

\[
F(\alpha) = \frac{\beta_U - \beta_{ND}}{\beta_D - \beta_{ND} + \beta_U\phi_\alpha - \beta_D\phi_\alpha}.
\]

Figure 3.6 illustrates these results. The region above the curve is the area where \(\frac{dZ}{d\rho} < 0\), which is where seroadaptation decreases HIV incidence.
CHAPTER 3. COMPARTMENTAL MODEL: EPIDEMIOLOGICAL RESULTS

Figure 3.6: The curve $\frac{d\rho}{d\alpha} = 0$ as a function of the fraction of the diagnosed population on treatment ($\alpha$), and the fraction of the HIV-positive population that is diagnosed ($F$). The area above the curve is the region where seroadaptation decreases HIV incidence. All parameter values are given in Appendix A.

In Figure 3.6 we see that a minimum of 80% of the HIV-positive population needs to be diagnosed in order for seroadaptation to decrease incidence. The benefit of seroadaptation on HIV incidence decreases as a larger proportion of the diagnosed population is put on treatment.
Chapter 4

Network Model

In Chapters 2 and 3 we discussed a compartmental model of an HIV epidemic. In this Chapter, we present a network model of the same epidemic.

Network model simulations were performed using NepidemiX – a Python software package for simulating contact processes on a network [1]. Each simulation requires a configuration file, which provides details including the number of iterations, what time-step to use, the type of network on which to run the simulations, model parameters, and initial node state distributions. The configuration file also specifies which process to run on the network. Processes specify the rules for node transitions, and multiple processes can be written in a single module. NepidemiX uses the Python software package NetworkX to generate the underlying network on which the simulations are run. The configuration file and module used in these simulations are given in Appendix B.

4.1 The Barabási-Albert Model

All network model simulations we consider take place on a Barabási-Albert (BA) graph [14]. A BA graph is a random, undirected graph built using the concept of preferential attachment. Generation of a BA graph begins with a small number of connected nodes. Subsequent nodes are added one at a time, and are initially attached to a fixed number of existing nodes.
The probability $p_i$ of a new node being connected to an existing node $i$ is

$$p_i = \frac{k_i}{\sum_j k_j},$$

where $k_i$ is the degree of node $i$. The probability that a randomly selected node in the network has degree $k$ is denoted as $p(k)$. In the limit of large network size and large $k$, BA graphs generate a power law degree distribution with exponent $3$ [14], that is,

$$p(k) \propto k^{-3}.$$

Networks with power law degree distributions are also called *scale free networks*. Power law degree distributions have been observed in sexual contact networks [14], and one estimate suggests the network of British MSM is scale free with an exponent between $1.5$ and $2$ [19].

### 4.2 Process on the Network

In the network model individuals are represented by nodes, and a link between two nodes indicates a sexual partnership. Nodes can be in one of four states, either susceptible (S), infected but not diagnosed (I), diagnosed (D), or on treatment (T), and this state may change over the course of the epidemic. All links between nodes can be classified as either *seroconcordant* or *serodiscordant*, depending on whether the link is between individuals of the same or opposing serostatus, respectively.

Each simulation begins with the generation of a new BA graph. Once the graph is generated, individuals are randomly assigned a status of $S, I, D$ or $T$, according to specified initial conditions indicating the proportion of individuals in each class.

Transmission of HIV may only occur along serodiscordant links, and the probability of a node becoming infected with HIV is proportional to the number of their nearest neighbours who are HIV-positive. Each susceptible individual is assumed to have an average of $C$ sexual acts per year, distributed equally among all of their nearest neighbours. Treatment is assumed to carry a $(1 - \phi)$ reduction in probability of transmission.

As a simplification from the compartmental model, we assume $100\%$ seroadaptation ($\rho = 1$). Consequently, the network model only uses two different transmission probabilities, $\beta_{ND}$.
and $\beta_D$, corresponding to the probability of HIV transmission per act when the susceptible individual falsely believes their partner is HIV-negative, and when the susceptible individual knows their partner is HIV-positive, respectively.

As in the compartmental model, infected individuals are diagnosed with rate $\gamma$, and diagnosed individuals are put on treatment at rate $\lambda$. There are three death rates, $\mu, d_1$ and $d_2$, corresponding to a natural death rate, and death rate due to HIV before and after starting treatment, respectively.

To maintain a constant population, an individual who dies in the model is replaced by a new susceptible individual. This new node is placed in the same position in the network with the same nearest neighbours. Computationally, this is equivalent to a node’s state changing from its current status to susceptible when it dies. When individuals die due to HIV-related causes, this placement of new arrivals may introduce a bias towards HIV infection since the new arrival may begin with many neighbours who are already HIV-positive.

### 4.3 Comparison to Compartmental Model

#### 4.3.1 Mean-Field Approximation

Prior to looking at how HIV spreads through a network structure, we first look at the mean-field approximation to the network model, and compare this to the compartmental model.

In the network model, the probability of a node in state $S$ with $k$ nearest neighbours transitioning to state $I$ in a given time period $dt$ is,

$$
P(S \rightarrow I|k) = \left[ \beta_{ND} \text{NN}(I) + \beta_D (1 - \phi) \text{NN}(T) + \beta_D \text{NN}(D) \right] 
\cdot \frac{C}{[\text{NN}(S) + \text{NN}(I) + \text{NN}(D) + \text{NN}(T)]} 
\cdot dt,
$$

(4.1)

where $\text{NN}(X)$ is the number of nearest neighbours in state $X$.

The mean-field approximation states that the probability for each node to be in state $X$ is $p_X = X/N$ [5]. Letting $x$ represent the proportion of nodes in state $X$ (so that $p_X = x$), a node with $k$ nearest neighbours will on average have $kx$ nearest neighbours in state $X$.
Therefore, in the mean-field approximation Equation (4.1) becomes,

\[ P(S \rightarrow I|k) = \left[ \beta_{ND}ki + \beta_D(1 - \phi)kt + \beta_D kd \right] \cdot \left[ \frac{C}{k(s + i + d + t)} \right] \cdot dt \]

\[ = \left[ \beta_{ND}ki + \beta_D(1 - \phi)kt + \beta_D kd \right] \cdot \frac{C}{k} \cdot dt \]

\[ = \left[ \beta_{ND}i + \beta_D(1 - \phi)t + \beta_D d \right] \cdot C \cdot dt \]

(4.2)

Note that the probability of transitioning from \( S \) to \( I \) is independent of a node’s degree \( k \), so that,

\[ P(S \rightarrow I|k) = P(S \rightarrow I) \]

Equation (4.2) is the probability of a single susceptible node becoming infected in time \( dt \). Since in the mean-field approximation the state of nodes are independent, we can write the change of the number of nodes in state \( S \) as,

\[ dS = \mu N \cdot dt + d_1(I + D) \cdot dt + d_2T \cdot dt - \mu S \cdot dt \]

\[ - S \left[ \beta_{ND} \left( \frac{I}{N} \right) + \beta_D(1 - \phi) \frac{T}{N} + \beta_D \frac{D}{N} \right] \cdot C \cdot dt, \]

where the first four terms account for new arrivals to the population and the death of susceptible nodes at each time step. From above, we have,

\[ \frac{dS}{dt} = \mu N + d_1(I + D) + d_2T - \mu S \]

\[ - S \left[ \beta_{ND} \left( \frac{I}{N} \right) + \beta_D(1 - \phi) \frac{T}{N} + \beta_D \frac{D}{N} \right] \cdot C \]

\[ = \mu N + d_1(I + D) + d_2T - \mu S \]

\[ - \left[ \beta_{ND}C \left( \frac{I}{N} \right) S + \beta_D C \left( \frac{D + (1 - \phi)T}{N} \right) S \right]. \]

(4.3)

With 0% serosorting \( (\sigma = 0) \) and 100% seroadaptation \( (\rho = 1) \), the rate of change of susceptible individuals in the mean-field approximation, given above in Equation (4.3), is equivalent to the rate of change of susceptible individuals in the compartmental model, given in Equation (2.1a). Therefore, in the limit of large population size, we would expect the network mean-field predictions to be the same as the compartmental model predictions.
Predicted equilibrium prevalence from numerical simulations of the mean-field approximation using NepidemiX is compared to predictions from the compartmental model in Figure 4.1 below. For these comparisons we assume 95% treatment efficiency ($\phi = 0.95$). The mean-field approximation results are an average of 20 independent simulations.

Figure 4.1: A comparison of the equilibrium prevalence predictions from the mean-field approximation to the network, and the compartmental model, assuming 0% serosorting and a population size of 10,000.

We can see in Figure 4.1 that the compartmental model and the mean-field approximation have very similar predictions for equilibrium prevalence, as expected. The minor discrepancies that occur, most notably at $\gamma^{-1} = 2$ and $\gamma^{-1} = 5$, can most likely be attributed to a too small population size and statistical error, as these results are an average of only 20 independent simulations.

4.3.2 Network Model Comparison

In this section we compare the true network model – a contact process on a network – and the compartmental model.

For these simulations we use a BA graph with mean degree $\approx 4$. We assume 0% serosorting ($\sigma = 0$) and 95% treatment efficiency ($\phi = 0.95$). A comparison of the predicted equilibrium prevalence given by the two models is shown in Figure 4.2. The network model results are an average of 20 independent simulations.
As shown in Figure 4.2, the network model predicts significantly lower equilibrium HIV prevalence than the compartmental model. Additionally, the critical time to diagnosis that determines whether or not an epidemic will be sustained in a population is greater in the network model. One would expect these results since compartmental models make a key assumption that each susceptible individual interacts with all infected individuals in the population, thereby overestimating the amount of new infections. In contrast, susceptible individuals in the network model are restricted to interactions with their nearest neighbours in the network structure, making infection less likely.
Chapter 5

Serosorting on a Network

To implement serosorting on a network, we make some additional changes to the model. Two new states are added for individuals who serosort – one for individuals who are susceptible \((S_S)\) and one for individuals who are infected but not diagnosed \((I_S)\).

Note that in the compartmental model we were only concerned with susceptible individuals who serosort. In the network model, however, we must also keep track of individuals who are undiagnosed and HIV-positive who serosort, as serosorting done by individuals in this subpopulation change the network in such a way that can influence the epidemic. Therefore to implement serosorting, a fraction \(\sigma\) of all susceptible individuals, as well as a fraction \(\sigma\) of all undiagnosed HIV-positive individuals, will serosort.

Serosorting is embedded in the network structure. Individuals who serosort will rewire their local network structure so that they have no links with anyone who is diagnosed as HIV-positive, including those on treatment. This has the potential to increase HIV incidence since nodes in state \(I_S\) will seek to form links with uninfected individuals, increasing their number of serodiscordant links. See Figure 5.1 below.
CHAPTER 5. SEROSORTING ON A NETWORK

VZ

migure XQTa h network representation of a population. The black, red, and purple nodes represent individuals who are HIV-negative, HIV-positive and diagnosed, and HIV-positive and undiagnosed, respectively. The figure shows the same population before (left) and after (right) 100% serosorting occurs. Note that each node has the same degree before and after serosorting.

We now present two different methods of serosorting.

5.1 Method 1: Removing Serodiscordant Links

In this first method of serosorting, all susceptible and undiagnosed HIV-positive nodes serosort with probability $\sigma$. At each time step, nodes in state $S_S$ and $I_S$ remove all of their links with anyone who is diagnosed HIV-positive. The removal of a link may represent the end of a sexual partnership, or the beginning of a partnership involving no acts of unprotected anal intercourse (UAI). After the removal of links, $C$ acts of UAI are equally distributed among the node's remaining neighbours. Note that this method does not preserve degree distribution, as nodes never acquire new links.

The above method was used to predict equilibrium HIV prevalence for 25%, 50%, 75% and 100% serosorting, for varying rates of diagnosis ($\gamma$) in the population. These results are shown in Figure 5.2 below.
CHAPTER 5. SEROSORTING ON A NETWORK

We can compare these results to the compartmental model predictions shown in Figure 3.3. Here, serosorting decreases HIV prevalence at equilibrium when time to diagnosis is less than 8 years, and has little impact if time to diagnosis is longer. Serosorting does not increase prevalence with longer times to diagnosis, as we saw with the compartmental model. Serosorting in the compartmental model and the network model can eliminate the epidemic as long as time to diagnosis is less than 3.5 and 7 years, respectively.

Overall, serosorting is more likely to lower equilibrium HIV prevalence in this network model compared to the compartmental model. This is because in the network model nodes can become isolated, after which there is no way for them to re-attach to other nodes, eliminating the chance of any HIV transmission. In comparison, the compartmental model assumes all nodes are part of one fully connected network.

5.2 Method 2: Network Rewiring

The second method of serosorting aims to preserve degree distribution, allowing individuals who serosort the possibility of maintaining a constant number of sexual partners. As in the previous method, all susceptible and undiagnosed HIV-positive nodes serosort with probability $\sigma$. At each time step, nodes in state $S_S$ and $I_S$ remove all of their links with anyone who is diagnosed HIV-positive. Now however, nodes seek to acquire a new link for every link that they remove, although this is not always possible. If two different nodes
serosort in the same time step, a link will be formed between the two individuals who serosort, and a second link will be formed between their previous partners. See Figure 5.3 below. As before, $C$ acts of UAI are equally distributed among the node’s remaining neighbours.

![Diagram](image)

**Figure 5.3**: Process of network rewiring. (a) If two individuals are serosorting, (b) they will both remove their discordant link. (c) Then a new link is formed between the two individuals who are serosorting, and the two diagnosed HIV-positive individuals. Note the new link between individuals who serosort may be serodiscordant.

The above method was used to predict equilibrium HIV prevalence for 25%, 50%, 75% and 100% serosorting, for varying rates of diagnosis ($\gamma$) in the population. Each of these scenarios resulted in an eliminated epidemic, that is, 0% HIV prevalence at equilibrium. This indicates that this second method of serosorting is more likely to eliminate an epidemic than the first method of serosorting. The reason for this is explained in the scenario below.

Suppose a node in state $S_S$ has four nearest neighbours in state $S, I, D$ and $T$, respectively. In the first method of serosorting the node in state $S_S$ will remove links with the nodes in state $D$ and $T$, so that his $C$ acts of UAI will be evenly distributed among the two remaining neighbours. In the second method of serosorting the node in state $S_S$ will remove links with the nodes in state $D$ and $T$, and is able to connect with another node in state $S_S$, so that his $C$ acts of UAI will be evenly distributed among the three remaining neighbours. Therefore the node in state $S_S$ is more likely to become infected using the first method of serosorting.
5.2.1 Generalized BA Graph

The standard BA graph generates a power-law degree distribution with exponent $\gamma = 3$. For basic susceptible-infected-recovered (SIR) models on scale free networks with a power-law exponent $\gamma < 3$, there is no minimum transmission probability $\lambda_c$ required to sustain an epidemic [13]. That is, there is an epidemic for any $\lambda_c > 0$. When $\gamma > 3$, there is a critical transmission probability $\lambda_c > 0$.

Using an algorithm described by Albert and Barabási [2] it is possible to construct a generalized BA graph with exponents other than 3. The first step in the algorithm is to begin with a small number of isolated nodes. Then at each time step, do one of the following:

(i) With probability $p$ add $m$ new links to the network. The starting node for each new link is chosen at random, and this node is linked to a second node selected with probability

$$\Pi(k_i) = \frac{k_i + 1}{\sum_j (k_j + 1)}.$$

(ii) With probability $q$ rewire $m$ links. To rewire a link, randomly chose a node $i$ and a link $l_{ij}$ connected to it. Remove the link $l_{ij}$ and add a new link $l_{ij'}$, where node $j'$ is chosen with probability $\Pi(k_{j'})$.

(iii) With probability $(1 - p - q)$ add a new node with $m$ new links connected to existing nodes in the network. These $m$ new links are connected to nodes chosen with probability $\Pi(k_i)$.

The network parameters $p$ and $q$ are chosen in the interval $0 \leq p < 1$ and $0 \leq q < 1 - p$. Selecting values for $p$ and $q$ within a certain range will produce a scale-free network with a specified power-law exponent [2]. Note that setting $p = q = 0$ produces the original BA network with power-law exponent $\gamma = 3$.

Network model simulations using the second method of serosorting described above were repeated using a generalized BA network with exponent $\gamma = 2.5$ ($p = 0, q = 2/5, m = 2$). Simulations testing 25%, 50%, 75% and 100% serosorting all predicted 0% prevalence at equilibrium. These results are the same as results from simulations on the standard BA graph with exponent 3, suggesting that the critical power-law exponent for this model is less than that of the SIR model. The generation of the generalized BA network was implemented by the author of the network software NepediX [1].
5.2.2 Instantaneous Incidence

This network model that uses network rewiring was also used to see how serosorting impacts HIV incidence – the number of new HIV cases per unit time. This was done on a standard BA graph, and incidence was recorded for 25%, 50%, 75% and 100% serosorting for varying rates of diagnosis ($\gamma$).

At the beginning of an epidemic with 20% initial prevalence, of which 80% is diagnosed and 20% is undiagnosed, increasing serosorting corresponds to an increase in incidence. As the epidemic progresses, serosorting has a decreasing impact on incidence until a point where incidence does not depend on the amount of serosorting. After this point, increasing serosorting corresponds to a decrease in incidence. This result is the same for all values of $\gamma$, and is comparable to the result from the compartmental model that found the rate of change in incidence is linear with respect to $\sigma$. This qualitative behaviour can be seen in Figure 5.4 below when mean time to diagnosis is 8 years.

![Figure 5.4: The qualitative impact of serosorting on HIV incidence. At the beginning of the epidemic with 20% initial prevalence (80% of which is diagnosed), increased serosorting corresponds to an increase in incidence. Over time, serosorting has a decreasing impact on incidence until a point where it does not impact incidence at all. After this point, increasing serosorting corresponds to a decrease in incidence. This figure assumes a population of 10,000 and a mean time to diagnosis of 8 years, however this qualitative behaviour is the same for all diagnosis rates $\gamma$. These results are an average of 10 independent simulations.](image)

The magnitude of the initial spike in incidence seen in Figure 5.4 diminishes with mean time to diagnosis, and is not present for any amount of serosorting when mean time to diagnosis
is 6 years or less. When mean time to diagnosis increases, there is also an initial spike in incidence for lower amounts of serosorting, however the size of the initial increase is always greater for larger amounts of serosorting. Additionally, the time for incidence to reach zero also decreases with mean time to diagnosis, and is less than 200 years when mean time to diagnosis is four years or less.

Serosorting corresponds to a decrease in incidence only after population prevalence decreases. This is because a randomly selected HIV-negative individual will have fewer known HIV-positive neighbours in populations with lower HIV prevalence. Consequently, after the removal of links with these diagnosed HIV-positive individuals, $C$ acts of UAI will be distributed among a greater number of remaining links. This decreases the probability of infection from an undiagnosed HIV-positive individual.

It is important to note that the initial conditions used to produce Figure 5.4 are not from data. Therefore, the initial transient behaviour and timescale may not be relevant to an actual epidemic. The key result is that all the incidence curves cross at a single point, indicating that there is some state of the epidemic and the network at which serosorting does not impact incidence. It is possible for serosorting to increase or decrease incidence, depending on the current state of the epidemic and the network configuration.
Chapter 6

Conclusion

Worldwide, men who have sex with men are disproportionately affected by HIV [6][28][22]. Current HIV prevention efforts have been unable to contain or reduce this growing epidemic [22], and there is an increasing need to evaluate and understand possible interventions.

Serosorting and seroadaptation are two such interventions that have the potential to make an impact on this expanding epidemic. These are behavioural interventions made at the individual level that govern partner selection and sexual behaviour, which are assumed to reduce the chance of HIV infection. Whether these interventions have a positive or negative impact depends on the context of the local epidemic in which they are applied. Specifically, the effect of serosorting and seroadaptation depend on the mean time to diagnosis in a population, HAART efficiency, and the extent of HAART coverage.

Using a deterministic compartmental model (Chapters 2-3) and a stochastic network model (Chapters 4-5) of HIV transmission among MSM, we looked at the impact of serosorting and seroadaptation on HIV prevalence at equilibrium and HIV incidence. These models took into account the rate of diagnosis, frequency of sexual acts, prevalence of the undiagnosed HIV-positive population, treatment coverage and efficiency, and change in transmission probability due to behaviour changes stemming from assumed knowledge of one’s partner’s HIV status. The network model also looked at the impact of network structure on an HIV epidemic.

Serosorting and seroadaptation are most beneficial in populations with low mean time to diagnosis and low levels of HAART coverage. With a low mean time to diagnosis, serosorting
decreases equilibrium HIV prevalence. This is seen in Figure 6.1(a) for the compartmental model, and Figure 6.1(b) for the network model. The critical time to diagnosis in these two models is approximately 3.6 years and 8 years, respectively, both of which are potentially achievable by the public health field.

Figure 6.1: Compartmental (a) and network (b) model predictions of steady state HIV prevalence with respect to the amount of serosorting ($\sigma$) in a population. Both models assume 100% seroadaptation, and the different curves correspond to varying times to diagnosis. In the network model, a fraction $\sigma$ of individuals are chosen to serosort at random by removing all of their serodiscordant links. Parameter values are given in Appendix A.

In Figure 6.1 we see that the detrimental impacts of serosorting associated with longer times to diagnosis in the compartmental model are nearly eliminated in the subsequent network model. Additionally, in Section 5.2 we saw that equilibrium prevalence was always zero when using the second method of serosorting, in which nodes aim to acquire a new link for every link that they remove.

Populations having a low mean time to diagnosis will have a larger proportion of the infected population diagnosed. Compartmental model results suggest that serosorting will
lower HIV incidence if at least 60% of the infected population is diagnosed, assuming 50% seroadaptation and 50% HAART coverage (Figure 3.4). Seroadaptation requires at least 80% of the infected population to be diagnosed in order to decrease incidence (Figure 3.6). As total HIV prevalence increases, greater proportions of the infected population must be diagnosed for serosorting and seroadaptation to continue to decrease HIV incidence.

In comparison, serosorting is more likely to increase incidence in the network model than in the compartmental model, as individuals who serosort are restricted to their local network structure. Looking at model predictions of incidence over time we are able to see the short and long term behaviour. In the long term, incidence always goes to zero, although this may take hundreds of years. More epidemiologically relevant is the short term behaviour, where increased serosorting corresponds to an increase in incidence.

The models also suggest that serosorting and seroadaptation are most beneficial in populations with low levels of treatment efficiency. This is because as treatment efficiency increases, it becomes less risky to choose a partner who is known to be HIV-positive and on treatment, rather than to take a chance with someone who is assumed to be HIV-negative. Additionally, serosorting is more likely to increase HIV incidence in populations with high treatment coverage.

This work has potential implications for public health policy. Based on the results mentioned above, serosorting should only be credited as an effective measure in populations which have achieved a critical time to diagnosis. If this target is not met, or if the mean time to diagnosis is unknown, promoting serosorting may be counterproductive. In addition to refraining from promoting serosorting, pro-active efforts could be made to warn people of the potential risks of these practices. These results suggest that emphasis should be placed on shortening the time to diagnosis, which involves an effective testing strategy.

## 6.1 Future Work

There are a few simplifying assumptions that were made in the models presented in the previous chapters which could be refined in future work. Presently, we do not consider the short yet highly infectious acute phase that occurs at the beginning of an HIV infection. Similarly, we do not account for variability in infectivity in the AIDS phase of infection. This could be addressed by introducing additional compartments in the deterministic model, and by creating additional states for individuals in the network model.
The models presented here only consider transmission of HIV due to acts of UAI. Future refinement of the model could incorporate levels of condom use in the population, allowing for potential additional HIV transmission due to condom failure. These models also assume honest status disclosure, and do not consider the possibility of purposefully hiding one’s positive HIV status. Further analysis could be done on the network model by eliminating the assumption of 100% seroadaptation.

At present it is difficult to obtain real-world sexual network data. If and when more data becomes available, this could be used to construct a network on which to run the epidemic. This would provide more accurate results, and could potentially change the conclusions presented in this thesis. Until real network data becomes available, the analyses done with these models could be repeated on different types of graphs, including Watts-Strogatz and other small-world graphs, in order to study further the impact of network structure. The results of these analyses along with results from the different serosorting scenarios could perhaps guide questions used in future cohort studies of MSM.

The models here suggest that serosorting will be beneficial in some settings while being detrimental in others. Therefore, it would be interesting and perhaps useful to parameterize these models with data from a variety of specific epidemics to see in which cases serosorting should be promoted or cautioned against.

### 6.1.1 Different Methods of Serosorting

It is not known exactly how real-world serosorting takes place. So far, we have considered only two different possible methods. Until more real-world information becomes available we can continue to test different methods, including serosorting based on node degree. In this method, individuals with low or high degree could be made to be more likely to serosort.

The probability of serosorting could be linked to a node’s degree so that nodes with higher degree will be less likely to serosort. This idea is supported by results of a nationwide survey of MSM in France [24], which found that HIV-negative men who practice serosorting had fewer male sexual partners during the past year than those who did not serosort.

To implement this method, a susceptible or undiagnosed node with degree $k$ will serosort
with probability $e^{-k\alpha}$, where the constant $\alpha$ is chosen so that at equilibrium,

$$\frac{S_S + I_S}{S + S_S + I + I_S} = \sigma.$$ 

Theoretically, a value for $\alpha$ could be found by solving the probability equation,

$$P(\text{serosorting}) = \sum_k P(\text{serosorting} \mid \text{deg}(n) = k) \cdot P(\text{deg}(n) = k)$$

$$\sigma = \sum_k e^{-k\alpha} \cdot P(\text{deg}(n) = k).$$

However, since the degree distribution of the network changes over time due to network rewiring, there is no closed form solution for $P(\text{deg}(n) = k)$. In fact, a different $\alpha$ is needed for different rates of diagnosis. To solve this problem, $\alpha$ values can be found experimentally.

So far, we have found that with 50% serosorting, the epidemic was always eliminated. We leave further analysis of this serosorting method as future work.
# Appendix A

## Model Parameters

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma$</td>
<td>The proportion of the population which serosorts</td>
<td>0–1</td>
<td>—</td>
</tr>
<tr>
<td>$\rho$</td>
<td>The proportion of the population practicing seroadaptation</td>
<td>0–1</td>
<td>—</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Diagnosis rate</td>
<td>1 – 10 years$^{-1}$</td>
<td>—</td>
</tr>
<tr>
<td>$B$</td>
<td>Rate of new arrivals into the population</td>
<td>Set to maintain a constant population</td>
<td>—</td>
</tr>
<tr>
<td>$\beta_{ND}$</td>
<td>Transmission probability per act of UAI when the infected partner is assumed to be HIV-negative</td>
<td>0.0069</td>
<td>[26]</td>
</tr>
<tr>
<td>$\beta_D$</td>
<td>Transmission probability per act of UAI when the infected partner is known to be HIV-positive</td>
<td>0.0023</td>
<td>[26]</td>
</tr>
<tr>
<td>$\beta_U$</td>
<td>Transmission probability per act of UAI when the infected partner’s status is unknown</td>
<td>0.0032</td>
<td>[26]</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Rate of removal of individuals from sexually active population</td>
<td>35 years$^{-1}$</td>
<td>[25]</td>
</tr>
</tbody>
</table>
## APPENDIX A. MODEL PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>$d_1$</td>
<td>Death rate due to HIV when individual is untreated</td>
<td>$11 \text{ years}^{-1}$</td>
<td>[21]</td>
</tr>
<tr>
<td>$d_2$</td>
<td>Death rate due to HIV when individual is on HAART</td>
<td>$0 \text{ years}^{-1}$</td>
<td>Modelling assumption</td>
</tr>
<tr>
<td>$\lambda$</td>
<td>Rate of initiating treatment after diagnosis</td>
<td>$2 \text{ years}^{-1}$</td>
<td>[11]</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Treatment efficiency</td>
<td>0.95 ((0.9-1)^1)</td>
<td>—</td>
</tr>
<tr>
<td>$C$</td>
<td>Average number of UAI acts per year</td>
<td>60</td>
<td>[23]</td>
</tr>
</tbody>
</table>

1Unless otherwise stated, $\phi = 0.95$. At times $\phi$ take values in the range of 0.9-1.
Appendix B

Python Code

Network model simulations were performed using NepidemiX – a Python software package for simulating contact processes on a network. This software was developed by Lukas Ahrenberg, a member of the IMPACT-HIV working group that is a collaboration of HIV researchers of the BC Centre for Excellence in HIV and AIDS and mathematicians of the Complex Systems Modelling Group (CSMG) at the IRMACS Centre of Simon Fraser University. The software is available for download at: https://github.com/impact-hiv/NepidemiX.

B.1 Configuration File

The configuration file contains details about individual simulations, including the number of iterations in the simulation and values for model parameters. A configuration file also specifies which process is to be used. This can either be a predefined process, such as the SIS process, or a scripted process that is contained in a module.

All network model simulations in this work use a population size of 10,000. Simulations used 12 time steps per year, and were run to equilibrium. The initial network for each simulation had a mean degree of approximately four, and all results presented are an average of 10 independent simulations.
APPENDIX B. PYTHON CODE

```python
[Simulation]
iterations = 3000  # number of iterations
dt = .1  # length of each iteration
process_class = SIDTProcessMethod  # specify which process class to use
process_class_module = extended_SIDT  # . and the module it’s in
module_paths = ../modules/

network_func = BA_networkx  # specify what kind of network to use
network_func_module = nepidemix.utilities.networkgeneratorwrappers

[NetworkParameters]
n = 10000  # number of nodes
m = 2  # initial number of links for each node

[ProcessParameters]  # set parameter values
mu = 0.028571
d1 = 0.09090909
d2 = 0
betaND = 0.0069
betaD = 0.0023
phi = 0.95
gamma = 1/8.0
eta = 0.5
sigma = 0.25
C = 60.0

[NodeStateDistribution]  # specify initial conditions for
{status:S} = 0.6  # proportion of nodes in each
{status:Ss} = 0.2  # state
{status:I} = 0.03  # note – the split between S/Ss and
{status:Is} = 0.01  # 1/Is depends on sigma
{status:D} = 0.16
{status:T} = 0.0

[Output]
output_dir = ../output/  # base name for all output files
base_name = M1sig25
unique = yes
save_config = yes

save_node_state = yes  # save node states at every time step
save_node_state_interval = 1
save_network = yes  # save initial and final network configuration
save_network_interval = 0
save_network_compress_file = no
```

Configuration file used for network model simulations.
APPENDIX B. PYTHON CODE

B.2 Module

Modules contain process classes that can be called by the configuration file. The following module contains three different process classes. The first, SIDTProcess, is the process described in Chapter 4, which involves no serosorting. This process was used to compare the network model to the compartmental model. The second and third processes, SIDTProcessMethod1 and SIDTProcessMethod2, describe the two methods of serosorting described in Chapter 5.

```python
import nepidemix.process as nwmp
import nepidemix.utilities.networkxtra as nwmx
import numpy

class SIDTProcess(nwmp.ExplicitStateProcess):
    r""
    SIDT process,
    Base Case - No serosorting
    For each susceptible node (S and Ss), C acts of UAI are distributed evenly among
    the number of nearest neighbours
    No network updates
    ""
    def __init__(self, mu, d1, d2, betaND, betaD, phi, gamma, eta, C, sigma):
        super(SIDTProcess, self).__init__(['{status:S}', '{status:I}', '{status:D}', '{status:T}', '{status:Ss}', '{status:Is}'],
                                         [],
                                         runNodeUpdate = True,
                                         runEdgeUpdate = False,
                                         runNetworkUpdate = False,
                                         constantTopology = True)

        self.mu = float(mu)
        self.d1 = float(d1)
        self.d2 = float(d2)
        self.betaND = float(betaND)
        self.betaD = float(betaD)
        self.phi = float(phi)
        self.gamma = float(gamma)
        self.eta = float(eta)
        self.C = float(C)
        self.sigma = float(sigma)

    def nodeUpdateRule(self, node, srcNetwork, dt):
        # Read original node state.
```
APPENDIX B. PYTHON CODE

srcState = node[1][self.STATE_ATTR_NAME]
# By default we have not changed states, so set
# the destination state to be the same as the source state.
dstState = srcState

# Start out with a dictionary of zero neighbors in each state.
nNSt = dict(zip(self.nodeStateIds,[0]∗len(self.nodeStateIds)))
# Calculate the actual numbers and update dictionary.
nNSt.update(nmex.utils.attributeCount(
    nmex.utils.neighbors_data_iter(srcNetwork, node[0]),
    self.STATE_ATTR_NAME))

# Pick a random number.
eventp = numpy.random.random_sample()
# Go through each state name, and chose an action.
if srcState == '{status:S}':
    if eventp < self.mu∗dt:
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
    elif ( nNSt['{status:S}'] + nNSt['{status:Ss}'] + nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}'] ) != 0 and eventp - self.mu∗dt < ( (self.betaND+(nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}'] + self.betaD∗nNSt['{status:D}'])∗(self.C/(nNSt['{status:S}'] + nNSt['{status:Ss}'] + nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}']))) )∗dt :
        dstState = '{status:I}'
    if srcState == '{status:Ss}':
        if eventp < self.mu∗dt:
            eventq = numpy.random.random_sample()
            if eventq < self.sigma:
                dstState = '{status:Ss}'
            else:
                dstState = '{status:S}'
        elif ( nNSt['{status:S}'] + nNSt['{status:Ss}'] + nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}'] ) != 0 and eventp - self.mu∗dt < ( (self.betaND+(nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}'] + self.betaD∗nNSt['{status:I}'] + self.betaD∗nNSt['{status:D}'])∗(self.C/(nNSt['{status:S}'] + nNSt['{status:Ss}'] + nNSt['{status:I}'] + nNSt['{status:Is}'] + nNSt['{status:D}'] + nNSt['{status:T}']))) )∗dt :
            dstState = '{status:I}'
    if srcState == '{status:I}':
        if eventp < (self.mu∗dt + self.d1∗dt):
            eventq = numpy.random.random_sample()
if eventq < self.sigma:
    dstState = '{status:Ss}'
else:
    dstState = '{status:S}'
elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status:D}'

if srcState == '{status:Is}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
    elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
        dstState = '{status:D}'

if srcState == '{status:D}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
    elif eventp - (self.mu*dt + self.d1*dt) < self.eta*dt:
        dstState = '{status:T}'

if srcState == '{status:T}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
    elif eventp - (self.mu*dt + self.d1*dt) < self.eta*dt:
        dstState = '{status:T}'

node[1][self.STATE_ATTR_NAME] = dstState

return node

#############################################################

class SIDTProcessMethod1(nwmp.ExplicitStateProcess):
    """
    S I D T process,

    Serosorting Method 1
    """

    For each susceptible node in S, C acts of UAI are distributed evenly among the number of nearest neighbours
APPENDIX B. PYTHON CODE

For each susceptible node in $S_s$, $C$ acts of UAI are distributed evenly among the number of 'susceptible' nearest neighbours ($S$, $S_s$, $I$, or $I_s$)

No network updates

For config file:

- set $\sigma$ in $[0,1]$
- set $S_s(0)$ as a fraction $\sigma$ of $S(0) + S_s(0)$
- set $I_s(0)$ as a fraction $\sigma$ of $I(0) + I_s(0)$

```python
""

def __init__(self, mu, d1, d2, betaND, betaD, phi, gamma, eta, C, sigma):
    super(SIDTProcessMethod1, self).__init__(
        ['', '{status:S}', '{status:I}', '{status:D}', '{status:T}', '{status:Ss}', '{status:Is}'],
        [],
        runNodeUpdate = True,
        runEdgeUpdate = False,
        runNetworkUpdate = False,
        constantTopology = True)

    self.mu = float(mu)
    self.d1 = float(d1)
    self.d2 = float(d2)
    self.betaND = float(betaND)
    self.betaD = float(betaD)
    self.phi = float(phi)
    self.gamma = float(gamma)
    self.eta = float(eta)
    self.C = float(C)
    self.sigma = float(sigma)

def nodeUpdateRule(self, node, srcNetwork, dt):
    # Read original node state.
    srcState = node[1][self.STATE_ATTR_NAME]
    # By default we have not changed states, so set
    # the destination state to be the same as the source state.
    dstState = srcState

    # Start out with a dictionary of zero neighbors in each state.
    nNSt = dict(zip(self.nodeStateIds, [0]*len(self.nodeStateIds)))
    # Calculate the actual numbers and update dictionary.
    nNSt.update(nwmx.util.attributeCount(
        nwmx.util.neighbors_data_iter(srcNetwork, node[0]),
        self.STATE_ATTR_NAME))

    # Pick a random number.
    eventp = numpy.random.random()
    # Go through each state name, and chose an action.
    if srcState == '{status:S}':
        if eventp < self.mu*dt:
            eventq = numpy.random.random()
```

if eventq < self.sigma:
    dstState = '{status:Ss}'
else:
    dstState = '{status:S}'

elif ( (nNSt['{status:S}'])+nNSt['{status:Ss}'] +nNSt['{status:D}'] ) != 0 and eventp - 
    self.mu*dt < ( (self.betaN*(nNSt['{status:I}'])+nNSt['{status:Is}'] "+nNSt['{status:D}'])* 
    (self.C/(nNSt['{status:S}'])+nNSt['{status:Ss}']"+nNSt['{status:Is}']") +nNSt['{status:R}']"+nNSt['{status:T}']") ) *dt :
    dstState = '{status:D}'

if srcState == '{status:S}':
    if eventp < self.mu*dt:
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
    else:
        dstState = '{status:S}''

elif ( (nNSt['{status:S}'])+nNSt['{status:Ss}'] +nNSt['{status:D}'] ) != 0 and eventp - 
    self.mu*dt < ( (self.betaN*(nNSt['{status:I}'])+nNSt['{status:Is}']")*(self.C/(nNSt['{status:S}']")+nNSt['{status:Ss}']") +nNSt['{status:R}']"+nNSt['{status:T}']") ) *dt :
    dstState = '{status:I}''

if srcState == '{status:I}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}''
    else:
        dstState = '{status:S}''

elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status:D}'

if srcState == '{status:R}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}''

elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status:D}''

if srcState == '{status:D}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
APPENDIX B. PYTHON CODE

```python
XZ
dstState = '{status:Ss}'
else:
dstState = '{status:S}'
elif eventp - (self.mu*dt + self.d1*dt) < self.eta*dt:
dstState = '{status:T}'
elif srcState == '{status:T}':
    if eventp < (self.mu*dt + self.d2*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status:Ss}'
        else:
            dstState = '{status:S}'
node[1][self.STATE_ATTR_NAME] = dstState
return node

class SIDTProcessMethod2(nwmp.ExplicitStateProcess):
    ""
    S I D T process,
    Serosorting Method 2
    ________________

    Node Updates:
    For each susceptible node in $S$, $C$ acts of UAI are distributed evenly among the
    number of nearest neighbours
    For each susceptible node in $Ss$, $C$ acts of UAI are distributed evenly among the
    number of nearest neighbours

    Network updates:
    At each timestep, for each 'serodiscordant' link involving a 'susceptible' in $Ss$
    or $I$:
    - all other serodiscordant links involving a 'susceptible' in $Ss$ or $I$ are
      checked
    - if possible, a second serodiscordant link is choosen, and rewiring occurs
    - otherwise, the original serodiscordant link is simply removed, with no
      rewiring (this changes the degree distribution)

    For config file:
    - set sigma in [0,1]
    - set $Ss(0)$ as a fraction sigma of $S(0) + Ss(0)$
    - set $Is(0)$ as a fraction sigma of $I(0) + Is(0)$
    ""
def __init__(self, mu, d1, d2, betaND, betaD, phi, gamma, eta, C, sigma):
```

super(SIDTProcessMethod2, self).__init__('[{status:S}', '{status:I}', '{status:D}', '{status:T}', '{status:Ss}', '{status:Is}'],
[],
runNodeUpdate = True,
runEdgeUpdate = False,
runNetworkUpdate = True,
)

self.mu = float(mu)
self.d1 = float(d1)
self.d2 = float(d2)
self.betaND = float(betaND)
self.betaD = float(betaD)
self.phi = float(phi)
self.gamma = float(gamma)
self.eta = float(eta)
self.C = float(C)
self.sigma = float(sigma)

def nodeUpdateRule(self, node, srcNetwork, dt):
    # Read original node state.
    srcState = node[1][self.STATE_ATTR_NAME]
    # By default we have not changed states, so set
    # the destination state to be the same as the source state.
    dstState = srcState

    # Start out with a dictionary of zero neighbors in each state.
    nNSt = dict(zip(self.nodeStateIds, [0]*len(self.nodeStateIds)))
    # Calculate the actual numbers and update dictionary.
    nNSt.update(nwmx.util.neighbors.attributeCount(
        nwmx.util.neighbors.data_iter(srcNetwork, node[0]),
        self.STATE_ATTR_NAME))

    # Pick a random number.
    eventp = numpy.random.random()
    # Go through each state name, and chose an action.
    if srcState == '{status:S}':
        if eventp < self.mu*dt:
            eventq = numpy.random.random()
            if eventq < self.sigma:
                dstState = '{status:Ss}'
            else:
                dstState = '{status:S}'
        elif ((nNSt['{status:S}']+nNSt['{status:Ss}']+nNSt['{status:I}'] +nNSt['{status:Ts}'] +nNSt['{status:T}']) != 0 and eventp -
            self.mu*dt < (self.betaND*(nNSt['{status:I}'] +nNSt['{status:Is}'])
            +self.betaD*(1-self.phi)*nNSt['{status:T}'] +self.betaD*nNSt['{status: D}'])*self.C/(nNSt['{status:S}'] +nNSt['{status:Ss}'] +nNSt['{status: I}'] +nNSt['{status:Is}'] +nNSt['{status:T}']) +nNSt['{status:D}'] +nNSt['{status:T}'] ) *dt ) :


dstState = '{status: I}"

if srcState == '{status: Ss}':
    if eventp < self.mu*dt:
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status: Ss}'
        else :
            dstState = '{status: S}'
    else:
        dstState = (nNST['{status: S}'] + nNST['{status: D}'] + nNST['{status: T}'])
else:
    dstState = '{status: S}"

if eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status: D}"

if srcState == '{status: I}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status: Ss}'
        else :
            dstState = '{status: S}"
    else:
        dstState = '{status: S}"

if eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status: D}"

if srcState == '{status: Is}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status: Ss}'
        else :
            dstState = '{status: S}"
    else:
        dstState = '{status: S}"

elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status: D}"

if srcState == '{status: D}':
    if eventp < (self.mu*dt + self.d1*dt):
        eventq = numpy.random.random_sample()
        if eventq < self.sigma:
            dstState = '{status: Ss}'
        else :
            dstState = '{status: S}"

elif eventp - (self.mu*dt + self.d1*dt) < self.gamma*dt:
    dstState = '{status: D}"

elif srcState == '{status: T}':
    if eventp < (self.mu*dt + self.d2*dt):
eventq = numpy.random.random_sample()
if eventq < self.sigma:
    dstState = '{status:Ss}'
else:
    dstState = '{status:S}'

node[1][self.STATE_ATTR_NAME] = dstState

return node

def networkUpdateRule(self, network, dt):
    disLinks = []
    # Make a list of all edges that need to be rewired. For all nodes...
    for node in network.nodes_iter(data = True):
        # If node is de-sorting...
        srcState = node[1][self.STATE_ATTR_NAME]
        if (srcState == '{status:Ss}' or srcState == '{status:ls}'):  # Check the state of all its nearest neighbors...
            for nn in nx.neighbors(network, node[0]):
                nNsrcState = nn[1][self.STATE_ATTR_NAME]
                if (nNsrcState == '{status:D}' or nNsrcState == '{status:T}'):  # If discordant, add to list of discordant edges
                    disLinks.append((node[0], nn[0]))

    # Remove and attempt to rewire all discordant edges
    while disLinks:
        edge = disLinks.pop()
        network.remove_edge(*edge)
        # Look for an edge to perform rewiring
        for secondEdge in disLinks:
            if (edge[0] != secondEdge[0] and edge[1] != secondEdge[1]):  # If a proper link is found, then remove the second link...
                disLinks.remove(secondEdge)
                network.remove_edge(*secondEdge)
                # and rewire the network with two new edges...
                network.add_edge(edge[0], secondEdge[0])
                network.add_edge(edge[1], secondEdge[1])
                # Once rewiring is done, no need to check other potential edges
                ... break

return network

Module containing scripted processes used in network model simulations.
Bibliography


