Title
Mathematical Models and Methods for Behavior in Social Networks: Urban Crime, Self-Exciting Interactions, and Information Spread

Permalink
https://escholarship.org/uc/item/0q7570q5

Author
Zipkin, Joseph Robert

Publication Date
2014

Peer reviewed|Thesis/dissertation
University of California
Los Angeles

Mathematical Models and Methods for Behavior in Social Networks: Urban Crime, Self-Exciting Interactions, and Information Spread

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

by

Joseph Robert Zipkin

2014
Abstract of the Dissertation

Mathematical Models and Methods for Behavior in Social Networks: Urban Crime, Self-Exciting Interactions, and Information Spread

by

Joseph Robert Zipkin

Doctor of Philosophy in Mathematics

University of California, Los Angeles, 2014

Professor Andrea Bertozzi, Chair

We present three applications of mathematics to problems in the study of social networks. The problems are different enough that each requires its own model and set of methods. We investigate these models through theory and simulation. The first application is to urban crime and police response. We prove some stability results for an agent-based model and present an efficient algorithm for numerical simulation. The second application is to communication patterns in social networks. Adopting a point-process model to reflect observed temporal bursts in communication, we discuss parameter estimation for such models. We then consider the problem of filling in missing data in records of communication, formulating it as a variational problem and developing a numerical method to solve it. The third application is to the spread of information within social networks. We introduce a simple model, provide some theoretical results, and discuss results of simulations.
The dissertation of Joseph Robert Zipkin is approved.

P. Jeffrey Brantingham

Inwon Kim

Joseph Teran

Andrea Bertozzi, Committee Chair

University of California, Los Angeles

2014
For my grandfather, Hyman David Zipkin,
who was eclectic, good, and inspiring.

*L’Chaim.*
# Table of Contents

1 Introduction ................................................................. 1

2 Urban crime and police response ....................................... 4
   2.1 Introduction ......................................................... 4
      2.1.1 Crime patterns .................................................. 4
      2.1.2 The discrete Short model ..................................... 5
      2.1.3 The continuous Short model ................................. 7
      2.1.4 Police in the Short model ..................................... 11
      2.1.5 Other models .................................................... 12
   2.2 Optimal deployment of police ...................................... 13
      2.2.1 Model specification ............................................ 13
      2.2.2 Dual formulation: a free boundary problem ............... 15
      2.2.3 Linear stability ............................................... 21
   2.3 Numerical implementation ......................................... 33
      2.3.1 Semi-implicit spectral method ................................ 33
      2.3.2 Reduction of the optimization problem .................... 35
      2.3.3 General deterrence: a discrete false position method ... 37
      2.3.4 Exponential deterrence: a homomorphism-based method ... 38
   2.4 Numerical results ................................................... 38
      2.4.1 Solution behavior .............................................. 39
      2.4.2 Dynamics of $\kappa$ ......................................... 41
      2.4.3 Return to linear stability .................................... 42
2.4.4 Comparison with the discrete model .................. 43
2.5 The radial problem for small police forces ................. 45
  2.5.1 Restatement as a boundary-value problem ............... 45
  2.5.2 Numerical method ..................................... 47
  2.5.3 Results ............................................. 48
2.6 Conclusion ............................................. 49

3 Point-process models of human interaction: Parameter estimation and missing data recovery ...................... 51
  3.1 Introduction ........................................... 51
    3.1.1 Burstiness and Hawkes processes .................... 51
    3.1.2 The IkeNet data set ................................ 54
  3.2 EM estimation of Hawkes process parameters ............... 56
    3.2.1 Generating Hawkes process time series ............... 57
    3.2.2 The EM algorithm .................................... 58
    3.2.3 Example: exponential triggering .................... 60
    3.2.4 Example: power-law triggering ...................... 61
    3.2.5 Comparison of exponential and power-law .......... 62
  3.3 The missing data problem ................................ 64
    3.3.1 Objective functions ................................ 65
    3.3.2 Numerical implementation ............................ 69
  3.4 Results ................................................. 79
    3.4.1 IkeNet ............................................ 80
    3.4.2 Simulated time series ............................... 83
    3.4.3 Discussion ......................................... 86
3.5 Conclusion .............................................................. 87

4 Information spread in social networks .............................. 89
  4.1 Introduction .......................................................... 89
  4.2 Model specification ................................................. 90
  4.3 Examples .............................................................. 91
    4.3.1 1D nearest-neighbor information sharing ............... 91
    4.3.2 The 4-regular graph .......................................... 98
    4.3.3 Erdos-Renyi random graphs: average dynamics .......... 99
  4.4 The Kolmogorov forward equation .............................. 100
  4.5 Future work ....................................................... 103

References ................................................................. 104
List of Figures

2.1 A snapshot of the attractiveness $A$ from simulations of the discrete model on a toroidal grid. ........................................... 6

2.2 A numerical solution of the continuous model (2.8) (left) and (2.9) (right) with periodic boundary conditions. $\eta = 0.03$, $A_0 = 0.5$, $\overline{B} = 2.5$. 9

2.3 Plots of the criminal density $\rho$ from numerical solutions of (2.14-2.16). Here $d(k) = e^{-k}$. (a) A near-steady-state solution with $\eta = 0.03$, $A_0 = 0.5$, $\overline{B} = 2.5$, and no police, same as Figure 2.2. (b) The same system at time 500 after $K = 300$ police were introduced at time 0. (c) $K = 500$, $t = 500$. (d) $K = 700$, $t = 500$. ........................................... 40

2.4 Plots of the criminal density $\rho$ from numerical solutions of (2.14-2.16) Here $d(k) = e^{-k}$. (a) A near-steady-state solution with $\eta = 0.05$, $A_0 = 1.1$, $\overline{B} = 2.9$, and no police. (b) The same system at time 500 after $K = 0.66$ police were introduced at time 0. (c) $K = 0.67$, $t = 500$. 40

2.5 Data gathered from solutions of (2.14-2.16), $\eta = 0.03$, $A_0 = 0.5$, $\overline{B} = 2.5$. For different values of $K$, the maximum value of $\kappa$ at $t = 1000$ is plotted for two classes of solutions. The solid line comes from solutions seeded with stable hot spots. The dashed line comes from solutions seeded with the homogeneous equilibrium values (2.23). Inset pictures are snapshots of $A$. ........................................... 41

2.6 Given police resources $K$, $\epsilon^*(K)$ is the critical amplitude of the Fourier perturbation in an initial condition for (2.14-2.16). For values of $\epsilon$ below $\epsilon^*(K)$, the system converges to homogeneous equilibrium. For values of $\epsilon$ above $\epsilon^*(K)$, the system remains spatially heterogeneous. 43
2.7 Plots of $A$ from numerical solutions of the discrete system with police,
$A_0 = \frac{1}{30}, \eta = 0.03, \omega = \frac{1}{15}, \theta = 0.56, \Gamma = 0.02, d(k) = e^{-k}$. (a) Stationary hot spots without police. (b) The same system at $t = 400$ after $K = 300$ police were introduced at $t = 0$. (c) $K = 500, t = 400$. (d) $K = 700, t = 400$.

2.8 $\max \kappa(\cdot, t)$ vs. $K$ at $t = 500$ for the two-dimensional problem (solid) and the radial problem (dashed).

3.1 Two time series. The axis is time, and circles indicate events. Each time series has 68 events. (a) Timestamps of emails sent between IkeNet user 6 and IkeNet user 15. (b) A simulated Poisson process.

3.2 Three realizations of a Hawkes process with $\mu = 0.15$ and $g(t) = 0.5e^{-0.6t}$. The horizontal axis is time. Circles indicate events, and the solid curve is the intensity.

3.3 Histogram of the number of emails sent between each pair of officers. Only pairs who exchanged fewer than 100 emails are shown; see Table 3.1 for the others.

3.4 Histograms showing the results of EM estimation of model parameters for (a) exponential and (b) power law triggering functions. For each model 50,000 time series were generated. About 1% of the results for $\omega$ and $q$ are omitted because they are outliers that exceed the right limit of the graph.

3.5 Triggering functions. Exponential: $g(t) = 3e^{-6t}$. Power law: $g(t) = (1 + t)^{-3}$. 
3.6 Diagrams of $\ell^p$ constraints (bold) with level sets of a function $f$. The dot indicates the point maximizing $f$ subject to the constraint. It occurs at the intersection between the constraint and the maximal level set that intersects it. (a) $p = 2$, $\partial f/\partial x_1 > 0$, $\partial f/\partial x_2 > 0$. (b) $p = 1$, $\partial f/\partial x_1 > 0$, $\partial f/\partial x_2 > 0$. (c) $p = 2$, $\partial f/\partial x_1 < 0$, $\partial f/\partial x_2 > 0$. 

3.7 Histogram of $\|x_{MRL}\|_0$ for bidirectional identity loss, $|I| = 1$, for all 8,896 cases.

4.1 Plots of the expectation value of knowledge $E(k_n(t))$ for nearest-neighbor information sharing, with parameters $N = 101$, $\alpha = 0.5$, $\lambda = 0.5$. The solid curves are analytical solutions from Proposition 4.3, and the circles are Monte Carlo values taken from 10,000 simulations.

4.2 Plots of $k_n(t)$ together with critical points (circles) and inflection points (squares) obtained numerically from (4.4) and (4.5), respectively.

4.3 $x_c(t)$ and $x_{\pm}(t)$ as a function of time, $t$. The solid line is the critical point $x_c(t)$ as predicted by the linear approximation (4.7), and the circles are the true values given by (4.4). The bottom and top dashes lines are the inflection points $x_-(t)$ and $x_+(t)$, respectively, as predicted by the linear approximation (4.8). The crosses are the true values given by (4.5). (“True values” are calculated by Newton’s method.)

4.4 Plots of the expectation value of knowledge $E(k_n(t))$ for nearest-neighbor information sharing on the 4-regular graph, at times $t = 50, 150, 250$, with parameters $N = 16^2$, $\alpha = 0.5$, $\lambda = 0.1$. Shown is the average knowledge taken from 10,000 Monte Carlo simulations.

4.5 A plot of the average knowledge in the system, $\overline{k}(t)$, versus time. Here $N = 50$, $M = 300$, $\lambda = 0.1$, $\alpha = 0.5$. Results are from 10,000 Monte Carlo simulations. The random Erdos-Renyi graph was different in each simulation.
<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Pairs of officers who exchanged &gt; 100 emails</td>
<td>55</td>
</tr>
<tr>
<td>3.2</td>
<td>EM estimation results</td>
<td>62</td>
</tr>
<tr>
<td>3.3</td>
<td>Log-likelihood loss vs. maximum</td>
<td>64</td>
</tr>
<tr>
<td>3.4</td>
<td>Objective functions</td>
<td>70</td>
</tr>
<tr>
<td>3.5</td>
<td>IkeNet: Predictive power for missing sender by method ($</td>
<td>I</td>
</tr>
<tr>
<td>3.6</td>
<td>IkeNet: Predictive power for missing receiver by method ($</td>
<td>I</td>
</tr>
<tr>
<td>3.7</td>
<td>IkeNet: Predictive power for unidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.8</td>
<td>IkeNet: Predictive power for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.9</td>
<td>IkeNet: Average energy for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.10</td>
<td>IkeNet: Predictive power for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.11</td>
<td>IkeNet: Average energy for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.12</td>
<td>Toy networks: Predictive power for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.13</td>
<td>FauxNet: Predictive power for bidirectional identity loss ($</td>
<td>I</td>
</tr>
<tr>
<td>3.14</td>
<td>FauxNet: Predictive power for bidirectional identity loss ($</td>
<td>I</td>
</tr>
</tbody>
</table>
ACKNOWLEDGMENTS

In preparing this thesis I have benefited from the assistance and support of many generous people. First and foremost, this thesis would not exist without the thoughtful guidance of my advisor, Andrea Bertozzi. Her confidence in me and support for my research have been invaluable. I am also grateful to the other members of my examination committee: Inwon Kim and Joseph Teran, who taught me much of what I know about applied and computational mathematics; and Jeffrey Brantingham, who inspired me to apply that knowledge to problems in social science.

I have been lucky to collaborate with Kathryn Coronges, who provided the IkeNet data and helped me understand it; Frederic Schoenberg, whose brought his statistical knowledge and sharp intuition to the point process project; and Martin Short, whose ubiquity in my research life despite moving to Atlanta reflects what a great collaborator and mentor he has been. I have also benefited from helpful discussions of the research in this thesis with Christoph Brune, Eric Fox, Theodore Kolokolnikov, Alexey Stomakhin, and Wotao Yin.

I am grateful for the financial support of the National Defense Science and Engineering Graduate Fellowship. My research also benefited from financial support of NSF grants DMS-0907931 and DMS-0968309, AFOSR MURI grant FA 9550-10-1-0569, ONR grant N000141210838, ARO MURI grants W911NF-10-1-0472 and W911NF-11-1-0332, and ARI MIPR 10086985.

I owe a great debt as well to my other math teachers. They are too numerous to list here, except for those whose encouragement led me to attend graduate school and become a mathematician: Ingrid Daubechies, Yakov Sinai, Bud Stuart, and Ramin Takloo-Bighash.

Finally, I must thank my family. California has felt like home because of my dear family in the Bay Area: Nina Zipkin-Berry, Bob Zipkin, Leo “Taz’dingo” Ungar, Pam Lampkin, Ben Zipkin, Anna Zipkin, Frank Berry, and Katie Gose. Also I am grateful
that I have been able to count on the love and support of my parents Paul Zipkin and Karen Stone and my sister Leah Zipkin. Finally, I'd like to thank my wife Emily Kosten, who since PHY 105 at Princeton has been the best partner and truest friend I could hope for.
Vita

2010 M.A., Mathematics, UCLA

2007 A.B., Mathematics, Princeton University

Senior Thesis: A Wavelet-Based Estimator of a Parameter of Long-Range Dependence With Applications to Financial Time Series

Professional Appointments

2011–14 Graduate Fellow, UCLA Department of Mathematics

2009–11 Teaching Assistant, UCLA Department of Mathematics

2010 Mentor, UCLA Applied Math REU


Publications


**HONORS & AWARDS**

2011–14 National Defense Science and Engineering Graduate Fellowship

2009–11 National Science Foundation VIGRE Fellowship

2004 Shapiro Prize for Academic Excellence, Princeton University

2004 Quin Morton Writing Prize, Princeton University
CHAPTER 1

Introduction

Each of the three main chapters of this thesis presents a model of a certain aspect of behavior in social networks and methods for simulating that model numerically. The common threads are the applicability to social science, the novelty of the models, and the use of scientific computing. In building the models we are guided by the demands of the application rather than interest in a given subfield. This means we use PDEs, optimal control, likelihood maximization, or stochastics as the situation warrants. Then, in turn, the numerical methods are guided by the demands of the models. The hope is that investigating these kinds of models for can inform our understanding of the behaviors in social networks that inspire them.

The applications belong to the same family, but the models themselves are different enough that including too much detail now would leave this introductory chapter disjointed. Instead, this chapter summarizes the work of the chapters following it and leave the bulk of the exposition for the chapters themselves. This chapter could be thought of as a collection of long abstracts, with the details common to introductions of dissertations in mathematics spread among the other chapters.

Chapter 2 presents a model of urban crime and police response first published in a paper with Martin Short and Andrea Bertozzi in *Discrete and Continuous Dynamical Systems* [84]. It builds on a prior agent-based model of urban crime [68, 69, 70]. Hot spots of crime localized in space and time are well documented. The previous model is a coupled system of PDEs describing criminal density in an urban area and how attractive criminals find different regions within the area. (The connection to social
networks is implicit: attractiveness diffuses through the system as information about past burglaries spreads.) However, it assumes a static or otherwise suboptimal police response to the crime. We introduce a program of police response to hot spots of crime in which the police adapt dynamically to changing crime patterns. In particular, they choose their deployment to solve an optimal control problem at every time. This gives rise to a free boundary problem for the police deployment’s spatial support. We present an efficient algorithm for solving this problem numerically and show that police presence can prompt adjacent hot spots to merge into larger structures we call “warm worms”. We conclude with a look at the one-dimensional radial system.

Chapter 3 draws from work with Frederic Schoenberg, Kathryn Coronges, and Andrea Bertozzi submitted to the *European Journal of Applied Mathematics* [83]. Electronic communications, as well as other categories of interactions within social networks, exhibit bursts of activity localized in time. Given this behavior, we adopt a self-exciting Hawkes process model. First we investigate parameter estimation of such processes and find that the choice of triggering function is not as important as getting the correct parameters given a choice. Then we present a method for filling in missing data in records of communications in social networks. This method draws on prior work on investigating gang crimes [73] but is closer to a maximum-likelihood approach. Finally we demonstrate the method using a data set composed of email records from a social network based at the United States Military Academy. The methods perform differently on this data and data from simulations, but the performance degrades only slightly as information is removed. The ability to fill in large blocks of missing social network data has implications for security, surveillance, and privacy.

Chapter 4 introduces a model of information-sharing in a social network. The social network is modeled as an undirected graph. Graph neighbors share knowledge with one another following random interactions governed by i.i.d. Poisson processes. Because knowledge is not destroyed when it is shared, the limiting behavior is not
diffusive but instead recalls the transport equation. We explore the dynamics of the 1D nearest-neighbor graph, note some of the difficulties in analysis for other configurations, and derive a Kolmogorov forward equation. We conclude by discussing possible directions to take with the model in order to develop it more fully.
CHAPTER 2

Urban crime and police response

2.1 Introduction

This chapter, which draws heavily from [84], adapts the Short model of urban crime [68] to include dynamic response by police. In particular, they choose their deployment to solve an optimal control problem at every time, responding to “hot spots” of high crime by deploying extra resources there in an attempt to deter further crime. This gives rise to a free-boundary problem for the boundary of the support of the police. We present an efficient algorithm for solving this problem numerically and show that police presence can prompt surprising interactions among adjacent hotspots.

2.1.1 Crime patterns

Urban crime is ubiquitous and heterogeneous: while it can certainly affect a whole city, clusters of crimes are often localized in time and space, forming “hot spots” of increased criminal activity [8, 25]. Simple spatial heterogeneity in the environment is insufficient to explain the temporal variations in crime recurrence [70]. Rather, the emergence of hot spots is linked to repeat victimization: a successful offender likely to re-offend in the same location, or nearby (near-repeat victimization). Such patterns have been observed in illicit activities from residential burglary [38] to insurgent activity in Iraq [75]. For law enforcement, a seemingly obvious response to a hot spot of crime is to deploy additional resources to the hot spot to deter further crime. These so-called “cops on the dots” strategies have sometimes successfully dispersed the hot
spot, but other times have merely displaced the crime to surrounding areas [7].

2.1.2 The discrete Short model

Short et al. [68] introduce an agent-based model for dynamics of crime. This model will be the basis for ours, so we review it in detail. Agents are criminals taking a biased random walk on a rectangular lattice with periodic boundary conditions, representing an urban environment. Time steps are discrete, with length $dt$. In each time step, each agent either strikes (i.e., commits a crime) at its present node or moves to an adjacent node. The agents’ decisions are influenced by a scalar attractiveness field $A$, whose dynamics are coupled to the agents’ dynamics.

During time step $t$, an agent at node $x$ strikes at $x$ with probability

$$p_s(x, t) = 1 - e^{-A(x, t)dt}.$$  \hfill (2.1)

Agents who strike exit the system; otherwise they move to an adjacent node. If node $x'$ is adjacent to node $x$ (denoted hereafter by $x' \sim x$), an agent moves from $x$ to $x'$ with probability $p_m(x, t; x')$, defined by

$$p_m(x, t; x') = \frac{A(x', t)}{\sum_{x'' \sim x} A(x'', t)}.$$  

The attractiveness $A$ is split into a static component $A_0$ and a dynamic component $B$ so that $A = A_0 + B$. To model repeat victimization, $B$ at $x$ is increased by $\theta$ for each strike that occurred at $x$ during the previous time period. Then, to model near-repeat victimization, $B$ diffuses to adjacent nodes and decays. If $S(x, t)$ denotes the number of strikes at node $x$ at time $t$, the dynamics of $B$ are given by

$$B(x, t + dt) = \left( (1 - \eta)B(x, t) + \frac{\eta}{4} \sum_{x' \sim x} B(x', t) \right) (1 - \omega dt) + \theta S(x, t),$$ \hfill (2.2)
Let \( n(x, t) \) be the number of criminals at node \( x \) at time \( t \). In addition to the above movement and exit rules, criminals generate at each node according to a Poisson process with parameter \( \Gamma \). Thus the expected number of criminals at each node follows

\[
E(n(x, t + dt|n(\cdot, t)) = \sum_{x' \sim x} n(x', t)(1 - p_s(x, t))p_m(x', t; x) + \Gamma dt. \tag{2.3}
\]

Under different choices of parameters, the system can give rise to different hot spot patterns. One such pattern is shown in Figure 2.1. The system is in homogeneous equilibrium when all nodes have the same attractiveness \( \bar{A} = A_0 + \bar{B} \) and the same average number of criminals \( \bar{n} \). In this scenario there is no net diffusion, so attractiveness from crimes balances the temporal decay in (2.2):

\[
\theta \pi p_s = \omega dt \bar{B}
\]

Likewise, there is no net movement of criminals, so the number of criminals exiting after striking balances the number entering in (2.3):

\[
\bar{n} p_s = \Gamma dt
\]
The equilibrium values are thus
\[
\mathcal{B} = \frac{\theta\Gamma}{\omega}, \quad \pi = \frac{\Gamma dt}{1 - e^{\Lambda dt}}.
\]

2.1.3 The continuous Short model

We now derive a continuum version of this model by taking a hydrodynamic limit of (2.2-2.3). Let \( \ell \) denote the length between any two grid nodes, and replace the diffusion coefficient \( \eta \) and the strike response coefficient \( \theta \) with their spatial-rate counterparts \( \eta/\ell^2 \) and \( \theta/\ell^2 \) so that \( \eta \) and \( \theta \) remain meaningful as \( \ell \to 0 \). Rewrite (2.2) as
\[
A(x, t + dt) - A_0 = \left( A(x, t) - A_0 + \eta \frac{\Delta_d A(x, t)}{\ell^2} \right) (1 - \omega dt) + \theta \frac{S(x, t)}{\ell^2}, \quad (2.4)
\]
where \( \Delta_d \) is the discrete Laplacian on the toroidal grid.

Let \( \rho = E(n)/\ell^2 \), the expectation value of the criminal density. Taking expectations in (2.4) we find that most of the additive terms are linear in the random variables; we will not change the notation for these. The exception is
\[
E(S(x, t)) \approx E(n(x, t))p_s(x, t) = \ell^2 \rho(x, t)(A(x, t)dt + O(dt^2)). \quad (2.5)
\]
The approximation notation \( \approx \) is necessary because, as Jones, Brantingham, and Chayes [39] point out, \( n \) and \( p_s \) are random variables that are likely not independent, but their correlations are not very important for understanding the system’s overall dynamics. One consequence of (2.5) is that \( \rho(x, t)A(x, t) \) is a good proxy for the amount of crime happening at \( (x, t) \), an observation we use throughout this chapter.

Substituting (2.5) into the expectation of (2.4), subtracting \( A(x, t) - A_0 \) from both sides, dividing by \( dt \), and disregarding \( O(dt) \) terms,
\[
\frac{A(x, t + dt) - A(x, t)}{dt} = \frac{\Delta_d A(x, t)}{\ell^2} + \theta \rho(x, t)A(x, t) - \omega (A(x, t) - A_0). \quad (2.6)
\]
Now rewrite (2.3) as

$$\ell^2 \rho(x, t + dt) = \ell^2 A(x, t) \sum_{x' \sim x} \frac{\rho(x', t)(1 - A(x', t)dt + O(dt^2))}{\sum_{x'' \sim x'} A(x'', t)} + \ell^2 \Gamma dt.$$ 

The criminal birth coefficient $\Gamma$ is rescaled by $\ell^2$ to preserve the system’s total birth rate. Dividing by $\ell^2$ and disregarding $O(dt^2)$ terms,

$$\rho(x, t + dt) = A(x, t) \sum_{x' \sim x} \frac{\rho(x', t)(1 - A(x', t)dt)}{\sum_{x'' \sim x'} A(x'', t)} + \Gamma dt.$$ 

This in turn can be written

$$\frac{\rho(x, t + dt) - \rho(x, t)}{dt} = A(x, t) \sum_{x' \sim x} \left( \frac{\rho(x', t)}{\sum_{x'' \sim x'} A(x', t)} - \frac{\rho(x, t)}{4A(x, t)} \right)$$

$$- A(x, t) \sum_{x' \sim x} \sum_{x'' \sim x'} \frac{\rho(x', t)A(x', t)}{A(x'', t)} + \Gamma. \quad (2.7)$$

We find the hydrodynamic limit by limiting (2.6) and (2.7) as $dt$ and $\ell \to 0$. The first term on the right-hand side of (2.7) appears like it will cause trouble, but some resolute calculation reveals that the sum is $O(\ell^2)$ if $\rho$ and $A$ have differentiable limits. If we hold $\ell^2/dt$ fixed at $dt$, $\ell \to 0$ and perform some rescaling, the limiting equations are

$$\frac{\partial A}{\partial t} = \eta \triangle A + \rho A - A + A_0, \quad (2.8)$$

$$\frac{\partial \rho}{\partial t} = \nabla \cdot (\nabla \rho - 2\rho \nabla \log A) - \rho A + \overline{B}. \quad (2.9)$$

(The rescaling replaces some parameters with $\overline{B}$, which becomes a parameter together with $\eta$ and $A_0$.)

This reaction-diffusion system resembles the well known Keller-Segel model of
chemotaxis, which in its most basic form is

\[
\begin{align*}
\frac{\partial v}{\partial t} &= \Delta v + u - v, \\
\frac{\partial u}{\partial t} &= \nabla \cdot (c_1 \nabla u - c_2 \nabla v).
\end{align*}
\]

In chemotaxis, cells, whose density is represented by \( u \), move in response to a chemical signal, whose concentration is represented by \( v \). The cells generate the signal, which diffuses and decays outside the cells [41, 42, 33]. Our \( A \) and \( \rho \) correspond to \( v \) and \( u \) respectively, except attractiveness grows in proportion to the number of crimes rather than the number of criminals, and criminals enter and exit rather than being conserved.

Solutions of the PDE system (2.8-2.9) exhibit stationary hot spots given appropriate parameters and initial conditions. Figure 2.2 shows a typical configuration of attractiveness \( A \) (left) and criminal density \( \rho \) (right). The solution was computed numerically using techniques described in [68] from a randomly perturbed initial condition. In a steady state solution the configuration of hot spots would be more uniform; however, the convergence from the situation in Figure 2.2 to steady state happens over a much longer time scale than the emergence of the hot spots.
Short et al. [68] perform a linear stability analysis and predict what combinations of the parameters \((\eta, A_0, \overline{B})\) will produce the hot spot-generating instability. Setting the derivatives in (2.8-2.9) to 0 and solving for \(A\) and \(\rho\) gives the homogeneous steady-state solution

\[
A = \overline{B} + A_0, \quad \rho = \frac{\overline{B}}{\overline{B} + A_0}.
\] (2.10)

The perturbations have the form \(a(x, t) = r(x, t) = e^{\sigma t + in \cdot x}\) (Laplace in time with rate \(\sigma\), Fourier in space with mode \(n \in \mathbb{Z}^2\)) and so are eigenfunctions of the differential operators \(\partial/\partial t\) and \(\triangle\). The equations are radial, so we can let \(n\) be a scalar integer: \(a(x, t) = r(x, t) = e^{\sigma t + in(x_1 + x_2)}\). The linearized system is

\[
\begin{pmatrix}
-\eta n^2 - 1 + \overline{\rho} & A \\
\frac{2n^2}{A} n^2 - \overline{\rho} & -n^2 - \overline{A}
\end{pmatrix}
\begin{pmatrix}
a \\
r
\end{pmatrix}
= \sigma
\begin{pmatrix}
a \\
r
\end{pmatrix}
\] (2.11)

The system is linearly unstable about the homogeneous steady state if for any \(n\) there exists a positive \(\sigma\) solving (2.11), or equivalently if the matrix in (2.11) has a positive eigenvalue. Short et al. find that linear instability has the necessary condition \(\overline{B} > \frac{1}{2} A_0\). Once that is satisfied, a sufficient condition is

\[
\eta < \frac{3\overline{\rho} + 1 - \sqrt{12\overline{\rho}}}{\overline{B} + A_0}.
\]

Short et al. [69] extend this work with weakly nonlinear stability analysis. They find that for several hot spot geometries the instability is characterized by bifurcations, some transcritical and some pitchfork, which can be subcritical or supercritical depending on the parameters.

prove several bifurcation results. Chaturapruek et al. [16] recast both the discrete and continuous short models using Lévy flights to drive the criminals’ motion, resulting in a nonlocal PDE system.

2.1.4 Police in the Short model

The original Short model does not consider police explicitly. In particular, it does not contemplate how criminals would respond to police presence or how police would respond to the emergence of hot spots. Here we review several attempts to answer this question.

In [69] and [70], Short et al. add a deterrence term $d$ to (2.8-2.9) to model the effect of police deployment. Deterrence decreases the total crime committed and discourages new criminals from entering the system. The continuum equations become

\[
\frac{\partial A}{\partial t} = \eta \Delta A + d \rho A - A + A_0 \tag{2.12}
\]

\[
\frac{\partial \rho}{\partial t} = \nabla \cdot (\nabla \rho - 2\rho \nabla \log A) - d \rho A + dB. \tag{2.13}
\]

To model a cops-on-the-dots deployment, $d$ is a function of space whose effect is centered at the center of a hot spot. Specifically, if the police start at time $t_0$, then

\[
d(x) = \frac{1}{2} \left(1 - \tanh(\mu(A(x, t_0) - A_c))\right),
\]

where $\mu$ and $A_c$ are positive parameters. Note that $d$ is not a function of time; it is a fixed deterrence profile based on the attractiveness at time $t_0$. In [69] and [70] it is shown that the deterrence can dissipate or merely displace hot spots of crime, depending on whether the pitchfork bifurcation is subcritical or supercritical.

Pitcher [61] modifies the original Short model to reflect slightly different assumptions about criminal behavior. Under certain parameters her model is linearly unstable and exhibits numerical solutions with hot spots resembling those in Figure 2.2. She
also considers the effect of police deploying to minimize crime and facing a resource constraint. Over time the police displace the criminals and the attractiveness to the hot spots’ edges, creating ring-like structures.

Jones, Brantingham and Chayes [39] return to the discrete model. They assign their police three deployment strategies: random patrols, in which police take a true random walk through the domain; cops on the dots, in which police, like criminals, take a biased random walk toward areas of greater attractiveness; and peripheral interdiction, in which police are biased to seek out the edges of hot spots. In agreement with research using field data, they find that random patrols are largely ineffective. Cops on the dots is effective early into its implementation, but peripheral interdiction eventually does just as well. For larger hot spots, peripheral interdiction is eventually better at reducing crime than cops on the dots.

2.1.5 Other models

Berestycki and Nadal [2] postulate a different PDE model grounded in economic rather than anthropological literature. Unlike the Short model it is not agent-based, but like the Short model it reflects repeat victimization and exhibits hot spots. Police interact with the system by exerting limited control over the local cost of illegal activity. Given enough resources, an appropriate police strategy can moderate the strength of the hot spots over time. Berestycki, Rodríguez and Ryzhik [3] find traveling wave solutions to the system.

Birks, Townsley, and Stewart [4] present a novel agent-based model of criminal behavior. As in the discrete Short model, criminal agents move through a rectangular grid and decide whether to strike based in part on different levels of attractiveness encountered throughout the domain. However, in their model targets are distributed randomly throughout the domain, rather than uniformly at one per node. Furthermore, agents are constrained to move along nodes of a transport network, also deter-
mined randomly. They are more likely to strike at targets near their current position if those targets are more attractive and if they are more aware of the targets. The study finds that hot spots are more likely to emerge if each agent has only a few destination transit nodes, if attractiveness is heterogeneous, and if awareness is heterogeneous. They do not discuss police directly.

2.2 Optimal deployment of police

We propose a new method for police deployment that takes into account police departments’ ability to strategize and coordinate their motions. Where prior work has postulated either that police deploy to a fixed location or that they may move but without any coordination, we propose that the police are constantly responding to new crimes that are committed. Their rule of movement will be to always deploy to minimize the total crime occurring instantaneously. This approach reflects police’s expertise, hierarchy, and coordination, in contrast with criminals who may not collaborate or even share a common goal.

2.2.1 Model specification

Return to equations (2.12-2.13), but assume now that police need not choose a static deployment based on the state of the system at one point in time. Instead allow them to allocate resources dynamically based on how the system responds to them. Under this framework we need a different understanding of the deterrence factor $d$. Rather than being a fixed function of space it must change in accordance with how the police choose to deploy. Let $\kappa(x, t)$ be the amount of police resources deployed to point $x$ at time $t$. Then the deterrence factor prevailing at $(x, t)$ is $d(\kappa(x, t))$, with $d$ now a function from $[0, \infty)$ to $(0, 1]$.

Intuition and convenience allow us to describe several properties we expect of $d$. 

13
Definition 2.1. A function \( d : [0, \infty) \to (0, 1] \) is a deterrence function if and only if each of the following is satisfied:

1. \( d(0) = 1 \): If no police are present at a location, the criminals behave there as they would in the original model.
2. \( d \) is decreasing: More police should result in less crime.
3. \( d \) is convex: Returns to additional police should diminish.
4. \( d \) is \( C^2 \): There are no critical levels of police that prevent \( d \) from being smooth.
5. \( d \) is positive: The police cannot prevent all crime no matter how much they deploy; however,
6. \( \lim_{k \to \infty} d(k) = 0 \): They can achieve a target deterrence level if they deploy enough resources.

Assumptions 3 and 4 together imply by the Inverse Function Theorem that \( d' \) is invertible and \( (d')^{-1} \) is \( C^1 \); we will use this later. The choice \( d(k) = e^{-k} \) will be convenient, particularly because of a computational advantage described in section 2.3.4.

The police deploy to solve a minimization problem. For now we will assume that their goal at time \( t \) is to minimize the total crime in the system at time \( t \), that is, \( \int d(\kappa) \rho A \, dx \), which we sometimes denote \( F(\kappa) \). They face a positivity constraint \( \kappa \geq 0 \) and some appropriate constraint on their total resources. For now we will assume that at each time step the police must deploy a fixed amount of resources \( K \); that is, for all \( t \), \( \int \kappa \, dx = K \).
The problem with dynamic police is therefore

\[ \frac{\partial A}{\partial t} = \eta \Delta A + d(\kappa)\rho A - A + A_0, \]  
\[ \frac{\partial \rho}{\partial t} = \nabla \cdot (\nabla \rho - 2\rho \nabla \log A) - d(\kappa)\rho A + d(\kappa)\bar{B}, \] 
\[ \kappa = \arg \min \left\{ \int_\Omega d(k)\rho A \, dx : k \in L^1(\Omega), k \geq 0, \int_\Omega k \, dx = K \right\}. \]  
(2.14) (2.15) (2.16)

The optimization problem (2.16) is the novel part of the model.

The feasible set \( \{k \in L^1(\Omega) : k \geq 0, \int_\Omega k \, dx = K \} \) is a convex subset of \( L^1 \), and (by the convexity of \( d \)) the objective functional \( \kappa \mapsto \int \int d(k)\rho A \, dx \) is a convex functional. Thus the optimization problem is convex. In the dual formulation it becomes a free boundary problem, with the boundary determined by \( \rho, A, \) and \( \lambda \), the dual variable associated to the \( L^1 \) constraint.

2.2.2 Dual formulation: a free boundary problem

The dual formulation has an immediate interpretation in the perspective of the police commander deciding where to deploy resources. Each incremental resource should go to the place where it can have the most incremental impact, until all the resources are used up. Limitations on the commander’s resources force him to ignore areas where the amount of crime is below some critical threshold. This threshold is determined by the dual variable \( \lambda \).

Throughout this section, fix \( t \) and consider the problem of recovering the optimal \( \kappa \) from \( \rho A \). We will assume that the introduction of police into the Short model has not disrupted the continuity guaranteed by [63] too much. In particular, we assume that \( \rho A \) is continuous.

**Theorem 2.2.** Let \( \rho A \) be non-negative everywhere, positive on a set of positive mea-
sure, and continuous. Then there exists $\lambda > 0$ such that

$$
\kappa = (d')^{-1}(-\lambda/\rho A)\chi_{U(\lambda, \rho A)},
$$

(2.17)

where $U(\lambda, f) = \{x \in \Omega : f(x) > -\lambda/d'(0)\}$. This $\lambda$ is the Lagrange multiplier associated to the $L^1$ constraint (2.16) and as such is the unique solution to the nonlinear equation

$$
G(\lambda) := \int_{U(\lambda, \rho A)} (d')^{-1}(-\lambda/\rho A)dx = K.
$$

(2.18)

The proof of Theorem 2.2 is presented in Lemmas 2.3-2.6. First we dispatch the easy case when $\kappa > 0$ throughout $\Omega$.

**Lemma 2.3.** Theorem 2.2 holds if $\kappa > 0$ everywhere.

*Proof.* We need not impose the $\kappa \geq 0$ constraint explicitly. Only the $L^1$ constraint remains, and the Lagrange multiplier method in calculus of variations suffices. The stationarity criterion is $d'(\kappa)\rho A + \lambda = 0$, where $\lambda$ is the Lagrange multiplier associated to the $L^1$ constraint. As noted previously $d'$ is invertible, so solving for $\kappa$ gives (2.17). Substituting this into the $L^1$ constraint gives (2.18), noting that $U(\lambda, \rho A) = \Omega$ in this case.

$G$ is strictly decreasing because $d'$ (and hence $(d')^{-1}$) is strictly increasing. Thus the $\lambda$ solving (2.18) is unique. $\square$

Now assume that $\kappa = 0$ on a subset of $\Omega$ of positive measure. There exists a Lagrange multiplier $\lambda > 0$ such that $d'(\kappa)\rho A + \lambda = 0$ wherever $\kappa > 0$. The $\kappa$ defined in (2.17) is a weakly increasing function of $\rho A$. In particular, there is some threshold value $C$ such that $\kappa(x) > 0$ if $\rho(x)A(x) > C$ but $\kappa(x) = 0$ if $\rho(x)A(x) < C$. Lemmas 2.4 and 2.5 show that $C = -\lambda/d'(0)$. The central idea is that the choice of $\lambda$ in (2.18) allows $\kappa$ to be continuous as $\rho A$ passes the $-\lambda/d'(0)$ threshold. Were it not continuous, it would be advantageous to redistribute some $\kappa$ from the edges of its
support to nearby areas where it is 0. Lemma 2.6 proves that this is the unique λ solving (2.18) by showing that G is strictly decreasing.

**Lemma 2.4.** $C \geq -\lambda/d'(0)$.

*Proof.* Suppose otherwise; then $C < -\lambda/d'(0)$. Let $\rho(x)A(x) = \frac{1}{2}(C - \lambda/d'(0))$; such an $x$ exists because $\rho A$ is continuous. Then $\kappa(x) > 0$, and

$$d'(\kappa(x)) = -\frac{2\lambda}{C - \lambda/d'(0)}.$$  

Both sides are negative, so

$$d'(\kappa(x)) < -\frac{2\lambda}{-\lambda/d'(0) - \lambda/d'(0)} = d'(0).$$

But by assumption $d$ is convex, so $d'(\kappa(x)) > d'(0)$. Thus $C \geq -\lambda/d'(0)$.  

**Lemma 2.5.** $C \leq -\lambda/d'(0)$.

*Proof.* Suppose otherwise; then $C > -\lambda/d'(0)$. Let $\epsilon \in (0, \frac{1}{2}C)$. Because $\rho A$ is continuous, there exist $\delta > 0$ and open subsets $U^+, U^-$ of $\Omega$ such that $|U^+| = |U^-| = \delta$, $0 < \rho A - C < \epsilon$ in $U^+$, and $0 < C - \rho A < \epsilon$ in $U^-$. By definition of $C$, $\kappa$ is positive in $U^+$ but zero in $U^-$. Setting $\kappa^* = (d')^{-1}(-\lambda/C)$, $\kappa > \kappa^* > 0$ in $U^+$. This last set of inequalities is independent of $\epsilon$, so we are free to further stipulate that $\epsilon < \frac{1}{2}\kappa^*$.

Now let $\hat{\kappa} = \kappa - \epsilon\chi_{U^+} + \epsilon\chi_{U^-}$, where $\chi_V$ is the indicator function on the set $V$. This $\hat{\kappa}$ meets the positivity and $L^1$ constraints, and

$$F(\hat{\kappa}) - F(\kappa) = \int_{U^+} (d(\kappa - \epsilon) - d(\kappa))\rho A dx + \int_{U^-} (d(\kappa + \epsilon) - d(\kappa))\rho A dx$$  

$$= \int_{U^+} (d(\kappa - \epsilon) - d(\kappa))\rho A dx + \int_{U^-} (d(\epsilon) - 1)\rho A dx$$  

$$< \int_{U^+} (d(\kappa - \epsilon) - d(\kappa))(C + \epsilon) dx + \int_{U^-} (d(\epsilon) - 1)(C - \epsilon) dx$$  

$$= (C + \epsilon) \int_{U^+} (d(\kappa - \epsilon) - d(\kappa)) dx + \delta(C - \epsilon)(d(\epsilon) - 1). \quad (2.19)$$
By Taylor’s theorem, there exists \( \epsilon^* (0) \in (0, \epsilon) \) such that
\[
d(\epsilon) - 1 = d'(0)\epsilon + \frac{1}{2} d''(\epsilon^*(0))\epsilon^2.
\]

Let \( M \) be the supremum of all values of \( d''(k) \) such that \( k \) is within \( \frac{1}{2}\kappa^* \) of any value taken by \( \kappa \). \( M \) is finite because \( \Omega \) is finite, \( \kappa \) is finite, and \( d'' \) is continuous. \( M \) is positive because \( d'' \) is positive. We can therefore write
\[
d(\epsilon) - 1 \leq d'(0)\epsilon + \frac{1}{2} M\epsilon^2. \tag{2.20}
\]

By similar reasoning, for each \( \kappa \) there exists \( \epsilon^*(\kappa) \in (0, \epsilon) \) such that
\[
d(\kappa - \epsilon) - d(\kappa) = -d'(\kappa)\epsilon + \frac{1}{2} d''(\kappa - \epsilon^* (\kappa))\epsilon^2 \leq -d'(\kappa)\epsilon + \frac{1}{2} M\epsilon^2. \tag{2.21}
\]

Substituting (2.20) and (2.21) into (2.19),
\[
F(\hat{\kappa}) - F(\kappa) < (C + \epsilon) \int_{U^+} (-d'(\kappa)\epsilon + \frac{1}{2} M\epsilon^2) dx + \delta(C - \epsilon)(d'(0)\epsilon + \frac{1}{2} M\epsilon^2)
\]
\[
= -(C + \epsilon)\epsilon \int_{U^+} d'(\kappa) dx + \delta(C - \epsilon)d'(0)\epsilon + \delta CM\epsilon^2
\]
\[
< -(C + \epsilon)\epsilon \int_{U^+} d'(\kappa^*) dx + \delta(C - \epsilon)d'(0)\epsilon + \delta CM\epsilon^2
\]
\[
= -\delta(C + \epsilon)d'(\kappa^*)\epsilon + \delta(C - \epsilon)d'(0)\epsilon + \delta CM\epsilon^2
\]
\[
= \delta C(d'(0) - d'(\kappa^*))\epsilon + \delta(CM - d'(\kappa^*) - d'(0))\epsilon^2. \tag{2.22}
\]

Choose
\[
\epsilon < \left| \frac{C(d'(0) - d'(\kappa^*))}{CM - d'(\kappa^*) - d'(0)} \right|
\]
so that (2.22) has the same sign as its linear term. Because \( d' \) is increasing, this term is negative. Thus \( F(\hat{\kappa}) - F(\kappa) < 0 \), but then \( \kappa \) is not optimal. Thus \( C \leq -\lambda/d'(0) \). □

Therefore, as predicted, \( C = -\lambda/d'(0) \), implying (2.17). This in turn implies
(2.18), so it remains only to show that $\lambda$ solves (2.18) uniquely. Lemma 2.6 suffices for this.

**Lemma 2.6.** $G$ is strictly decreasing.

**Proof.** If $h > 0$, then

$$
\frac{G(\lambda + h) - G(\lambda)}{h} = \int_{U(\lambda, \rho A)} (d')^{-1} \left( \frac{-(\lambda + h)/\rho A}{h} \right) dx
- \frac{1}{h} \int_{U(\lambda, \rho A) \setminus U(\lambda + h, \rho A)} (d')^{-1} \left( -\frac{\lambda}{\rho A} \right) dx
=: I_1(h) - I_2(h).
$$

Because everything is continuous,

$$
\lim_{h \to 0^+} I_1(h) = \int_{U(\lambda, \rho A)} \frac{\partial}{\partial \lambda} (d')^{-1} \left( -\frac{\lambda}{\rho A} \right) dx = -\int_{U(\lambda, \rho A)} \frac{1}{\rho A d''((d')^{-1}(-\lambda/\rho A))} dx.
$$

Because $h > 0$,

$$
I_2(h) < \frac{1}{h} \int_{U(\lambda, \rho A) \setminus U(\lambda + h, \rho A)} (d')^{-1} \left( \frac{-\lambda}{-(\lambda + h)/d'(0)} \right) dx
= \frac{|U(\lambda, \rho A) \setminus U(\lambda + h, \rho A)|}{h} (d')^{-1} \left( \frac{\lambda}{\lambda + h} d'(0) \right).
$$

The Taylor approximation

$$
(d')^{-1} \left( \frac{\lambda}{\lambda + h} d'(0) \right) = -\frac{d'(0)}{\lambda d''(0)} h + o(h)
$$

then yields

$$
I_2(h) < -\frac{d'(0)}{\lambda d''(0)} |U(\lambda, \rho A) \setminus U(\lambda + h, \rho A)| + o(1).
$$
Because $U(\lambda, \rho A) \setminus U(\lambda + h_2, \rho A) \subset U(\lambda, \rho A) \setminus U(\lambda + h_1, \rho A)$ if $h_2 < h_1$,

$$
\lim_{h \to 0^+} I_2(h) \leq -\frac{d'(0)}{\lambda d''(0)} \left| \bigcap_{h > 0} \{ -\lambda/d'(0) < \rho A \leq -(\lambda + h)/d'(0) \} \right|
= -\frac{d'(0)}{\lambda d''(0)} \left| \{ \rho A > -\lambda/d'(0) \} \cap \bigcap_{h > 0} \{ \rho A \leq -(\lambda + h)/d'(0) \} \right|
= -\frac{d'(0)}{\lambda d''(0)} \left| \{ \rho A > -\lambda/d'(0) \} \cap \{ \rho A \leq -\lambda/d'(0) \} \right|
= 0.
$$

But $I_2(h) > 0$ by definition, so $\lim_{h \to 0^+} I_2(h) = 0$. Thus

$$
\lim_{h \to 0^+} \frac{G(\lambda + h) - G(\lambda)}{h} = -\int_{\{ \rho A > -\lambda/d'(0) \}} \frac{1}{\rho A d''((d')^{-1}(-\lambda/\rho A))} dx < 0.
$$

If $h < 0$, then

$$
\frac{G(\lambda + h) - G(\lambda)}{h} = \frac{1}{h} \int_{U(\lambda+h,\rho A) \setminus U(\lambda,\rho A)} (d')^{-1}(-(\lambda + h)/\rho A) dx
+ \int_{U(\lambda,\rho A)} (d')^{-1}(-(\lambda + h)/\rho A) - (d')^{-1}(-\lambda/\rho A) \frac{1}{h} dx
=: I_3(h) + I_4(h).
$$

As before,

$$
\lim_{h \to 0^-} I_4(h) = \int_{U(\lambda,\rho A)} \frac{\partial}{\partial \lambda} (d')^{-1}(-\lambda/\rho A) dx = -\int_{U(\lambda,\rho A)} \frac{1}{\rho A d''((d')^{-1}(-\lambda/\rho A))} dx.
$$

We cannot guarantee that $\lim_{h \to 0^-} I_3(h) = 0$. This is because

$$
\lim_{h \to 0^-} |U(\lambda + h, \rho A) \setminus U(\lambda, \rho A)| = |\{ \rho A = -\lambda/d'(0) \}|,
$$

which is not necessarily 0. (If it is, then the left- and right-handed derivatives have
the same negative value, and we are through.) However, we do have the bounds
\[
\frac{d'(0)}{\lambda d''(0)} \left| -\left( \frac{\lambda + h}{d'(0)} \right) < \rho A \leq -\frac{\lambda}{d'(0)} \right| + o(1) \leq I_3(h) \leq 0
\]
as \( h \to 0^- \). Thus the left-handed derivative is no greater than the right-handed derivative, which is itself negative, and \( G \) is strictly decreasing. \( \square \)

We may therefore replace (2.16) in the statement of the problem with (2.18), with \( \kappa \) understood to be given explicitly by (2.17). This formulation will be helpful for determining \( \kappa \) numerically.

Geometrically, \( \lambda \) specifies the threshold crime level below which the police will not bother to deploy. This level is precisely \( \lambda \) scaled by the positive constant \( -1/d''(0) \). As the system evolves, \( \lambda \) too will change. Implicit differentiation in \( t \) on the equation \( G(\lambda) = K \) gives
\[
\frac{\lambda_t}{\lambda} = \frac{\int_{U(\lambda, \rho A)} \frac{1}{\rho A d''(\kappa)} \left( \frac{A_t}{A} + \frac{\rho_t}{\rho} \right) dx}{\int_{U(\lambda, \rho A)} \frac{1}{\rho A d''(\kappa)} dx}.
\]
Thus the growth rate of \( \lambda \) is determined by a weighted average of the growth rates of \( A \) and \( \rho \) inside \( U(\lambda, \rho A) \), the region in which the police are deployed. In particular, if \( \rho \) and \( A \) reach steady state inside \( U(\lambda, \rho A) \), then \( \lambda_t = 0 \), and the whole system is in steady state.

Corollary 2.7 reveals why exponential deterrence will be so convenient later on.

**Corollary 2.7.** If \( d(k) = e^{-k} \), then throughout \( U(\lambda, \rho A) \), \( d(\kappa)\rho A = \lambda \).

**Proof.** A straightforward calculation following from \( (d')^{-1}(y) = -\log(-y) \). \( \square \)

### 2.2.3 Linear stability

The homogeneous steady state solution \((A, \rho, \kappa) = (\tilde{A}, \tilde{\rho}, \tilde{\kappa})\) to (2.14-2.16) is given by
\[
\tilde{A} = d(\tilde{\kappa})\overline{B} + A_0, \quad \tilde{\rho} = \frac{\overline{B}}{\tilde{A}}, \quad \tilde{\kappa} = \frac{K}{|\Omega|}.
\]
Compare this with (2.10), the homogeneous steady state solution of the original Short model. We can also write \( \tilde{A} = d(\tilde{\kappa})\tilde{A} + (1 - d(\tilde{\kappa}))A_0 \). The equilibrium value of \( A \) is thus a convex combination of the equilibrium value without police (\( \tilde{A} \)) and its minimum possible value (\( A_0 \)). More police decreases \( \tilde{A} \) but increases \( \tilde{\rho} \) proportionally, so that \( \tilde{\rho}\tilde{A} = \tilde{\rho}\tilde{A} \). The average amount of crime in \( \Omega \) is \( d(\tilde{\kappa})\tilde{A} \), less by a factor of \( d(\tilde{\kappa}) \) than in the system without police.

To understand how the addition of police impacts the stability of the system, we perform a linear stability analysis centered at this equilibrium. Because \( \kappa \) is completely determined at each time by \( A \) and \( \rho \), we perturb only \( A \) and \( \rho \) and then determine the consequences on \( \kappa \). Let \( \epsilon > 0 \) be a small parameter specifying the scale of the perturbation: \( A = \tilde{A} + \epsilon a, \rho = \tilde{\rho} + \epsilon r \). The scale parameter \( \epsilon \) relative to the perturbations \( a \) and \( r \) is such that \( A \) and \( \rho \) are positive everywhere. Let \( \kappa \) denote the solution to (2.16) given this \( A \) and \( \rho \).

First we prove that, if \( \epsilon \) is small enough, then \( \kappa > 0 \) throughout \( \Omega \).

**Proposition 2.8.** Let \( K > 0 \), and let \( \rho A \) be positive and bounded. Then there exists \( \epsilon_0 > 0 \) such that, if \( 0 < \epsilon < \epsilon_0 \), then \( \kappa \) as defined in (2.16) is positive throughout \( \Omega \).

**Proof.** It suffices to show that, for all \( y \in \Omega \),

\[
\int_\Omega (d')^{-1}
\left(
\frac{d'(0)}{\rho(x)A(x)}
\right)
\left(
\frac{\rho(y)A(y)}{\rho(x)A(x)}
\right)
\leq K.
\]

If this is so, then the police deploy to every point in \( \Omega \). Let \( M = \max(\max |a|, \max |r|) \).

Because \((d')^{-1}\) is increasing and \( d'(0) < 0 \),

\[
\int_\Omega (d')^{-1}
\left(
\frac{d'(0)}{\rho(x)A(x)}
\right)
\leq \int_\Omega (d')^{-1}
\left(
\frac{\min \rho A}{\max \rho A}
\right)
\leq \int_\Omega (d')^{-1}
\left(
\frac{d'(0)\min \rho A}{\max \rho A}
\right)
\leq \int_\Omega (d')^{-1}
\left(
\frac{\tilde{\rho}\tilde{A} + (\tilde{\rho} + \tilde{A})M \epsilon + M^2 \epsilon^2}{\tilde{\rho}\tilde{A} - (\tilde{\rho} + \tilde{A})M \epsilon - M^2 \epsilon^2}
\right)
\leq \frac{\Omega |(d')^{-1}|}{\left(\frac{\tilde{\rho}\tilde{A} + (\tilde{\rho} + \tilde{A})M \epsilon + M^2 \epsilon^2}{\tilde{\rho}\tilde{A} - (\tilde{\rho} + \tilde{A})M \epsilon - M^2 \epsilon^2}\right)}.
\]

(2.24)
When $\epsilon = 0$, the quantity in (2.24) is 0. It is also a continuous function of $\epsilon$, so it is less than $K$ up to some positive value, say $\epsilon_0$. For $\epsilon \in (0, \epsilon_0)$, $\kappa > 0$ throughout $\Omega$. \hfill \Box

Assuming henceforth that $\epsilon$ is small enough to satisfy Proposition 2.8,

$$\kappa = (d')^{-1} \left( - \frac{\lambda(\epsilon)}{(\tilde{\rho} + \epsilon r)(\tilde{A} + \epsilon a)} \right).$$

Note the dependence of $\lambda$ on $\epsilon$. We seek a Taylor expansion of $\kappa$ in $\epsilon$, centered at $\epsilon = 0$. By assumption, when $\epsilon = 0$, $\kappa = \tilde{\kappa}$. For the linear coefficient,

$$\frac{\partial \kappa}{\partial \epsilon} \bigg|_{\epsilon=0} = \frac{1}{d''((d')^{-1}(-\lambda(0)/\tilde{\rho}\tilde{A}))} \cdot \tilde{\rho}\tilde{A} \lambda'(0) - (\tilde{\rho}a + \tilde{A}r)\lambda(0)$$

$$= \frac{1}{d''(\tilde{\kappa})} \left( \frac{(\tilde{\rho}a + \tilde{A}r)\lambda(0)}{\tilde{\rho}^2\tilde{A}^2} - \lambda'(0) \frac{\lambda(0)}{\tilde{\rho}\tilde{A}} \right).$$

(2.25)

Recalling that $\lambda(0)$ is the Lagrange multiplier associated to the homogeneous steady state, $\lambda(0) = -d'(\tilde{\kappa})\tilde{\rho}\tilde{A}$. We get $\lambda'(0)$ through implicit differentiation on the $L^1$ constraint

$$\int_{\Omega} (d')^{-1} \left( \frac{\lambda(\epsilon)}{(\tilde{\rho} + \epsilon r)(\tilde{A} + \epsilon a)} \right) \, dx = K.$$

Differentiating with respect to $\epsilon$ and setting $\epsilon = 0$,

$$\int_{\Omega} \frac{1}{d''(\tilde{\kappa})} \left( \frac{(\tilde{\rho}a + \tilde{A}r)\lambda(0)}{\tilde{\rho}^2\tilde{A}^2} - \frac{\lambda'(0)}{\tilde{\rho}\tilde{A}} \right) \, dx = 0$$

$$\lambda'(0) = \frac{\lambda(0)}{\left|\Omega\right|\tilde{\rho}\tilde{A}} \left( \tilde{\rho} \int_{\Omega} a \, dx + \tilde{A} \int_{\Omega} r \, dx \right).$$

Assuming the perturbations have zero spatial mean, $\lambda'(0) = 0$. Substituting into (2.25),

$$\frac{\partial \kappa}{\partial \epsilon} \bigg|_{\epsilon=0} = \frac{1}{d''(\tilde{\kappa})} \left( \frac{(\tilde{\rho}a + \tilde{A}r)}{\tilde{\rho}^2\tilde{A}^2} \right) = - \frac{d'(\tilde{\kappa})}{d''(\tilde{\kappa})} \left( \frac{r}{\tilde{\rho}} + \frac{a}{\tilde{A}} \right).$$

Thus

$$\kappa = \tilde{\kappa} - \frac{d'(\tilde{\kappa})}{d''(\tilde{\kappa})} \left( \frac{r}{\tilde{\rho}} + \frac{a}{\tilde{A}} \right) \epsilon + o(\epsilon).$$

23
This leads to an expansion of \(d(\kappa)\) about \(\epsilon = 0\):

\[
d(\kappa) = d(\tilde{\kappa}) - \frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})} \left( \frac{r}{\rho} + \frac{a}{A} \right) \epsilon + o(\epsilon).
\]

Substitute the perturbation into (2.14):

\[
\frac{\partial a}{\partial t} = \epsilon \eta \Delta a + \left( d(\tilde{\kappa}) - \frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})} \left( \frac{r}{\rho} + \frac{a}{A} \right) \epsilon \right) (\tilde{A} + \epsilon a)(\tilde{\rho} + \epsilon r) - (\tilde{A} + \epsilon a - A_0) + o(\epsilon)
\]

\[
= d(\tilde{\kappa}) \tilde{\rho} - (\tilde{A} - A_0)
\]

\[
+ \left( \eta \Delta a - a + d(\tilde{\kappa}) \tilde{A} r + d(\tilde{\kappa}) \tilde{\rho} a - \frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})} \tilde{\rho} \tilde{A} \left( \frac{r}{\rho} + \frac{a}{A} \right) \right) \epsilon + o(\epsilon).
\]

The order-1 terms cancel by (2.23). Disregarding the superlinear terms, we get

\[
\frac{\partial a}{\partial t} = \eta \Delta a - a + \left( d(\tilde{\kappa}) - \frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})} \right) \tilde{\rho} a + \left( d(\tilde{\kappa}) - \frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})} \right) \tilde{A} r
\]

Follow the same course for (2.15):

\[
\frac{\partial r}{\partial t} = \Delta r - d(\tilde{\kappa}) \tilde{A} r - \frac{2\tilde{\rho}}{A} \Delta a - d(\tilde{\kappa}) \tilde{\rho} a.
\]

We choose the form \(a(x,t) = r(x,t) = e^{\sigma t + in \cdot x}\) for the perturbations so they will be eigenfunctions of the differential operators \(\partial/\partial t\) and \(\Delta\). The equations are radial, so we can let \(n\) be scalar: \(a(x,t) = r(x,t) = e^{\sigma t + in(x_1 + x_2)}\) This yields the eigenvalue system

\[
\begin{pmatrix}
-\eta n^2 - 1 + D(\tilde{\kappa}) \tilde{\rho} & D(\tilde{\kappa}) \tilde{A} \\
\frac{2\tilde{\rho}}{A} n^2 - d(\tilde{\kappa}) \tilde{\rho} & -n^2 - d(\tilde{\kappa}) \tilde{A}
\end{pmatrix}
\begin{pmatrix}
a \\
r
\end{pmatrix}
= \sigma
\begin{pmatrix}
a \\
r
\end{pmatrix}
\]

(2.26)

where \(D(\tilde{\kappa}) = d(\tilde{\kappa}) - d'(\tilde{\kappa})^2/d''(\tilde{\kappa})\). Note that \(D(\tilde{\kappa}) < d(\tilde{\kappa})\). This system is the same as (2.11) but with some terms in the stability matrix scaled by \(d(\tilde{\kappa})\) or \(D(\tilde{\kappa})\).

This system can be linearly unstable, for example if \(\eta = \frac{1}{10}, A_0 = \frac{1}{100}, \tilde{B} = \frac{1}{2}, d(\kappa) = (1 + \kappa)^{-1}, K = 1, n = 1\). For the remainder of this section we present sufficient conditions for linear stability (or equivalently necessary conditions for linear
instability) because these give clues to how effective police can be in stopping crime. We begin with a special case.

**Proposition 2.9.** If \( d \) is exponential, then the system is unconditionally linearly stable.

**Proof.** In this case \( D \) is identically 0, so the matrix in (2.26) becomes

\[
\begin{pmatrix}
-\eta n^2 - 1 & 0 \\
2\tilde{\rho}n^2 - d(\tilde{\kappa})\tilde{\rho} - n^2 - d(\tilde{\kappa})\tilde{A}
\end{pmatrix}.
\]

The unconditionally negative eigenvalues can be read off the diagonal. \( \square \)

This is a marked contrast with the \( K = 0 \) case, the original Short system, in which the proper choice of parameters can yield linear instability [68]; this system is linearly stable for any positive value of \( K \). The key to the discontinuity lies in Proposition 2.8: given \( K > 0 \), \( \kappa \) is everywhere positive once \( \epsilon \) is small enough. In contrast, when \( K = 0 \), \( \kappa \) cannot be positive anywhere.

Of course, \( D(\tilde{\kappa}) \) need not be identically 0 in general.\(^1\) Theorem 2.11 states that even these scenarios can be stabilized given enough police. It and several other results rely on the following lemma.

**Lemma 2.10.** For all \( \eta, A_0, \bar{B}, K > 0 \) and every deterrence function \( d \), if the system is linearly unstable, then

\[
1 + d(\tilde{\kappa})\eta\tilde{A} - 3D(\tilde{\kappa})\tilde{\rho} < -\eta. \tag{2.27}
\]

**Proof.** The greater of the eigenvalues of the matrix in (2.26) is

\[
\sigma_n = \frac{-q_\eta + \sqrt{q_\eta^2 - 4\left(\eta n^4 + (1 + d(\tilde{\kappa})\eta\tilde{A} - 3D(\tilde{\kappa})\tilde{\rho})n^2 + d(\tilde{\kappa})\tilde{A}\right)}}{2} \tag{2.28}
\]

\(^1\)Indeed, the only deterrence function \( d \) solving the ordinary differential equation \( D(k) = 0 \) is \( d(k) = e^{-ck} \), with \( c \) a constant of integration.
where \( q_n = 1 + d(\bar{\kappa}) \bar{A} + (1 + \eta)n^2 - D(\bar{\kappa})\bar{\rho} \). The system is linearly unstable if and only if \( \text{Re} \sigma_n > 0 \) for some integer \( n \).

First we show that \( q_n > 0 \), for which \( q_0 > 0 \) suffices. First,

\[
d(\bar{\kappa})(d(\bar{\kappa})B + A_0)^2 + B \frac{d'(\bar{\kappa})^2}{d''(\bar{\kappa})} > 0
\]

\[
d(\bar{\kappa})B + A_0 + d(\bar{\kappa})(d(\bar{\kappa})B + A_0)^2 - \left( d(\bar{\kappa}) - \frac{d'(\bar{\kappa})^2}{d''(\bar{\kappa})} \right)B > A_0
\]

\[
1 + d(\bar{\kappa})(d(\bar{\kappa})B + A_0) - \left( \frac{d'(\bar{\kappa})^2}{d''(\bar{\kappa})} \right) \frac{B}{d(\bar{\kappa})B + A_0} > \frac{A_0}{d(\bar{\kappa})B + A_0}.
\]

Recalling the definitions of \( \bar{A} \) and \( \bar{\rho} \) from (2.23),

\[
1 + d(\bar{\kappa})\bar{A} - D(\bar{\kappa})\bar{\rho} > \frac{A_0}{\bar{A}},
\]

so \( q_0 > 0 \).

Thus \( \text{Re} \sigma_n > 0 \) only if the square root in (2.28) is real and greater than \( q_n \). Equivalently, there exists an integer \( n \) such that

\[
\eta n^4 + (1 + d(\bar{\kappa})\eta \bar{A} - 3D(\bar{\kappa})\bar{\rho})n^2 + d(\bar{\kappa})\bar{A} < 0.
\]

(2.29)

Equivalently, the polynomial \( f_1 \) defined by

\[
f_1(x) = \eta x^2 + (1 + d(\bar{\kappa})\eta \bar{A} - 3D(\bar{\kappa})\bar{\rho})x + d(\bar{\kappa})\bar{A}
\]

has two real roots, and a positive square integer lies between them. This is true only if the greater root of \( f_1 \) is greater than 1:

\[
-\frac{1 + d(\bar{\kappa})\eta \bar{A} - 3D(\bar{\kappa})\bar{\rho} + \sqrt{(1 + d(\bar{\kappa})\eta \bar{A} - 3D(\bar{\kappa})\bar{\rho})^2 - 4\eta d(\bar{\kappa})\bar{A}}}{2\eta} > 1.
\]
 equivalently,
\[
\sqrt{(1 + d(\tilde{\kappa}) \eta \tilde{\Lambda} - 3D(\tilde{\kappa}) \tilde{\rho})^2 - 4d(\tilde{\kappa}) \eta \tilde{\Lambda} > 2\eta + 1 + d(\tilde{\kappa}) \eta \tilde{\Lambda} - 3D(\tilde{\kappa}) \tilde{\rho}.}
\] 
(2.30)

If the right-hand side of (2.30) is positive, then we can square both sides, leading ultimately to the inequality

\[
1 + d(\tilde{\kappa}) \eta \tilde{\Lambda} - 3D(\tilde{\kappa}) \tilde{\rho} < -\eta - d(\tilde{\kappa}) \tilde{\Lambda}.
\]

If the right-hand side of (2.30) is negative, then

\[
1 + d(\tilde{\kappa}) \eta \tilde{\Lambda} - 3D(\tilde{\kappa}) \tilde{\rho} < -2\eta.
\]

In either case, (2.27) holds. □

Theorem 2.11. For all \( \eta, A_0, B > 0 \) and every deterrence function \( d \), there exists \( K^* > 0 \) such that the system is linearly stable if \( K > K^* \).

Proof. Suppose the system is linearly unstable. Then (2.27) holds, which implies

\[
1 + d(\tilde{\kappa}) \eta \tilde{\Lambda} - 3d(\tilde{\kappa}) \tilde{\rho} < -\eta.
\]

Per (2.23), substitute \( \frac{B}{\tilde{\Lambda}} \) for \( \tilde{\rho} \) and \( d(\tilde{\kappa}) \frac{B}{\tilde{\Lambda}} + A_0 \) for \( \tilde{\Lambda} \). Collect terms in \( d(\tilde{\kappa}) \):

\[
\eta \frac{B^2}{\tilde{\Lambda}} d(\tilde{\kappa})^3 + 2\eta A_0 \frac{B}{\tilde{\Lambda}} d(\tilde{\kappa})^2 + (\eta A_0^2 + (\eta - 2)B) d(\tilde{\kappa}) + (\eta + 1)A_0 \leq 0.
\]

Define the polynomial \( f_2 \) by

\[
f_2(y) = \eta \frac{B^2}{\tilde{\Lambda}} y^3 + 2\eta A_0 \frac{B}{\tilde{\Lambda}} y^2 + (\eta A_0^2 + (\eta - 2)B) y + (\eta + 1)A_0.
\]

Because \( f_2(0) = (\eta + 1)A_0 > 0 \), by the intermediate value theorem there exists \( y_2 \in (0, 1] \) such that \( f_1(y) \geq 0 \) for all \( y \in [0, y_2] \). Let \( K^* = d^{-1}(y_2)|\Omega| \). By Definition
2.1, if \( K > K^* \), then \( d(\tilde{\kappa}) \leq y_2 \). Thus \( f_2(d(\tilde{\kappa})) \geq 0 \), so by Lemma 2.10 the system is linearly stable.

Thus \( K^* \) is bounded from above in terms of the zeros of a cubic polynomial. Specifically, \( y_2 \) is the lesser of 1 and the least positive root of \( f_2 \), if it has a positive root. If it does not, then the system is linearly stable for all \( K > 0 \).

The \( K^* \) found in the above proof is by no means sharp. Indeed, Proposition 2.9 tells us that \( K^* = 0 \) when \( d \) is exponential. Even in general cases we may be able to sharpen it further. For (2.29) to hold, the polynomial \( f_1 \) must have two real roots, so its discriminant must be positive. Equivalently,

\[
(1 + d(\tilde{\kappa})\eta\tilde{A} - 3D(\tilde{\kappa})\tilde{\rho})^2 \geq 4d(\tilde{\kappa})\eta\tilde{A}.
\] (2.31)

If \( 1 + d(\tilde{\kappa})\eta\tilde{A} - 3D(\tilde{\kappa})\tilde{\rho} > 0 \), then the roots of \( p \) are not positive, so (2.31) reduces to

\[
1 + d(\tilde{\kappa})\eta\tilde{A} - 3D(\tilde{\kappa})\tilde{\rho} \leq -2\sqrt{d(\tilde{\kappa})\eta\tilde{A}}.
\] (2.32)

If we use (2.32) in the place of (2.27) in the proof of Theorem 2.11, we get the function \( f_3 \) defined by

\[
f_3(y) = \eta\tilde{B}^2y^3 + 2\eta A_0\tilde{B}y^2 + (\eta A_0^2 - 2\tilde{B})y + 2\sqrt{\eta(\tilde{B}y + A_0)^3}y^{1/2} + A_0.
\]

in the place of \( f_2 \). Because \( f_3 \) is continuous and \( f_3(0) > 0 \), a similar intermediate value theorem argument works here as well. If \( y_3 \) is the lesser of 1 and the least positive root of \( f_3 \), then \( K^* = \min(d^{-1}(y_2), d^{-1}(y_3))|\Omega| \). Corollary 2.19 sharpens \( K^* \) further.

Moreover, Theorem 2.11 is not the only sufficient condition for linear stability. Propositions 2.12 and 2.14 and Corollaries 2.13 and 2.15 provide conditions relying only on the parameters \( \eta, A_0, \) and \( \tilde{B} \).

**Proposition 2.12.** If \( \eta \geq 2\tilde{B}/(\tilde{B} + A_0^2) \), then the system is linearly stable.
**Proof.** The polynomial $f_2$ has all positive coefficients, except possibly for the linear coefficient. If $\eta \geq 2\bar{B}/(\bar{B} + A_0^2)$, then the linear coefficient is non-negative, so $f_2$ is monotone. Thus it can have no positive roots, since $f_2(0) > 0$. □

**Corollary 2.13.** If $\eta \geq 2$, then the system is linearly stable.

**Proposition 2.14.** If $\eta \geq (2\bar{B} - A_0)/(\bar{B} + A_0)$, then the system is linearly stable.

**Proof.** Write (2.27) as

$$(\eta + 1)A_0 + (\eta - 2)d(\tilde{\kappa})\bar{B} + d(\tilde{\kappa})\eta(d(\tilde{\kappa})\bar{B} + A_0)^2 < -3\bar{B}\frac{d'(\tilde{\kappa})^2}{d''(\tilde{\kappa})^2}. $$

If the system is linearly unstable, then $\eta - 2 < 0$ by Corollary 2.13. Because the right-hand side is negative and $0 < d(\tilde{\kappa}) < 1$,

$$(\eta + 1)A_0 + (\eta - 2)\bar{B} < 0. \quad (2.33)$$

Because (2.27) is necessary for linear instability by Lemma 2.10, so is (2.33). □

**Corollary 2.15.** If $2\bar{B} \leq A_0$, then the system is linearly stable.

Since stating Theorem 2.11 we have ignored the information about the shape of $d$ encoded in the $d'(\tilde{\kappa})^2/d''(\tilde{\kappa})$ term of $D(\tilde{\kappa})$. We now use it and Lemma 2.10 to treat another special case.

**Proposition 2.16.** If $d(\kappa) = (1 + \kappa)^{-p}$ with $p \geq 2$, then the system is unconditionally linearly stable.

**Proof.** In this case,

$$D(\kappa) = \frac{1}{p+1} (1 + \kappa)^{-p} = \frac{1}{p+1} d(\kappa).$$

Substituting this into (2.27) gives

$$1 + d(\tilde{\kappa})\eta\hat{A} - \frac{3}{p+1} d(\tilde{\kappa})\hat{p} < -\eta.$$
Collecting terms in $d(\kappa)$,

$$\eta B^2 d(\kappa)^3 + 2\eta A_0 B d(\kappa)^2 + (\eta A_0^2 + (\eta + 1 - \frac{3}{p+1}) B) d(\kappa) + (\eta + 1) A_0 < 0.$$ 

This can hold only if the polynomial $f_4$ defined by

$$f_4(y) = \eta B^2 y^3 + 2\eta A_0 B y^2 + (\eta A_0^2 + (\eta + 1 - \frac{3}{p+1}) B) y + (\eta + 1) A_0$$

has a root between 0 and 1. Since $f_4(0) > 0$ and all coefficients are positive, except perhaps for the linear coefficient, the linear coefficient must be negative for such a root to exist. Thus

$$p < \frac{3}{\eta + 1 + \eta A_0^2 / B} - 1$$

is a necessary condition for linear stability. But this is impossible if $p \geq 2$. 

Proposition 2.17 uses the $(d')^2/d''$ information to show that, in general, $d$ must decay slower than exponentially in unstable regions.

**Proposition 2.17.** Given $\eta, A_0, \overline{B} > 0$ and a deterrence function $d$, let $[K_1, K_2]$ be an interval such that the system is linearly unstable if $K_1 \leq K \leq K_2$. If $K_1 < K \leq K_2$,

then

$$d(\kappa) > d(\kappa_1) \exp \left( \frac{d'(\kappa_1)}{d(\kappa_1)} (\kappa - \kappa_1) \right). \quad (2.34)$$

**Proof.** Recall first that $\kappa = K/|\Omega|$, so we may speak interchangeably of $K$ and $\kappa$. It follows from Lemma 2.10 that $D$ is positive on $[\kappa_1, \kappa_2]$, or equivalently

$$\frac{d''(\kappa)}{d'(\kappa)} < \frac{d'(\kappa)}{d(\kappa)}.$$
Integrating both sides from \( \tilde{\kappa}_1 \) to \( \tilde{\kappa} \),

\[
\log \frac{d'(\tilde{\kappa})}{d'(\tilde{\kappa}_1)} < \log \frac{d(\tilde{\kappa})}{d(\tilde{\kappa}_1)}.
\]

Integrating again,

\[
\log \frac{d(\tilde{\kappa})}{d(\tilde{\kappa}_1)} > \frac{d'(\tilde{\kappa}_1)}{d(\tilde{\kappa}_1)} (\tilde{\kappa} - \tilde{\kappa}_1),
\]

which is (2.34).

Proposition 2.9 is a corollary to Proposition 2.17.

We now present a final result limiting the instability of the system. Like Proposition 2.17, it establishes a lower bound for \( d(\tilde{\kappa}) \) in unstable regions based on local properties of \( d \). It relies on a similar technique.

**Proposition 2.18.** Given \( \eta, A_0, \overline{B} > 0 \) and a deterrence function \( d \), let \([K_1, K_2]\) be an interval such that the system is linearly unstable if \( K_1 \leq K \leq K_2 \). Then there exist positive constants \( \alpha_1, \alpha_2, \alpha_3, \) and \( p \) such that, if \( K_1 < K \leq K_2 \), then

\[
d(\tilde{\kappa}) > \alpha_1 + (\alpha_2 + \alpha_3 (\tilde{\kappa} - \tilde{\kappa}_1))^{-p}.
\]

**Proof.** By Lemma 2.10, if the system is linearly unstable, then

\[
\frac{1}{3\overline{B}} \frac{d''(\tilde{\kappa})}{d'(\tilde{\kappa})} > -\frac{d'(\tilde{\kappa})}{(1 + \eta + d(\tilde{\kappa})\eta(d(\tilde{\kappa})\overline{B} + A_0))(d(\tilde{\kappa})\overline{B} + A_0) - 3\overline{B}d(\tilde{\kappa})}.
\]

We would like to use the same procedure as in Proposition 2.17, integrating both sides to get a first-order differential inequality in \( d \). However, the left-hand side is not integrable analytically, so we replace (2.27) with

\[
1 + d(\tilde{\kappa})\eta \frac{A_0^2}{A} - 3D(\tilde{\kappa})\tilde{\rho} < -\eta,
\]

\[ (2.36) \]
which follows from (2.27) because $A_0 < \bar{A}$. Write (2.36) as

$$3\bar{B} \frac{d''(\bar{\kappa})}{d'(\bar{\kappa})^2} < -(\eta A_0^2 + (\eta - 2)\bar{B})d(\bar{\kappa}) - (\eta + 1)A_0. \quad (2.37)$$

The left-hand side is positive, so the right-hand side is also positive. Thus we may take reciprocals of both sides:

$$\frac{1}{3\bar{B}} \frac{d''(\bar{\kappa})}{d'(\bar{\kappa})^2} > \frac{1}{-(\eta A_0^2 + (\eta - 2)\bar{B})d(\bar{\kappa}) - (\eta + 1)A_0}. \quad (2.38)$$

Let $b = -(\eta A_0^2 + (\eta - 2)\bar{B})$, $c = (\eta + 1)A_0$. By Proposition 2.12,

$$b > -\frac{2\bar{B}}{\bar{B} + A_0^2} A_0^2 - \left(\frac{2\bar{B}}{\bar{B} + A_0^2} - 2\right) \bar{B} = 0.$$

By (2.37), $bd(\bar{\kappa}) - c > 0$. Write (2.38) as

$$\frac{1}{3\bar{B}} \frac{d''(\bar{\kappa})}{d'(\bar{\kappa})^2} < \frac{d'(\bar{\kappa})}{bd(\bar{\kappa}) - c} \quad (2.39)$$

Integrate (2.39) from $\bar{\kappa}_1$ to $\bar{\kappa}$:

$$\frac{1}{3\bar{B}} \log \frac{d'(\bar{\kappa})}{d'(\bar{\kappa}_1)} < \frac{1}{b} \log \frac{bd(\bar{\kappa}) - c}{bd(\bar{\kappa}_1) - c}$$

$$d'(\bar{\kappa}) > \left(\frac{bd(\bar{\kappa}) - c}{bd(\bar{\kappa}_1) - c}\right)^{3\bar{B}/b}$$

$$d'(\bar{\kappa}_1) < \left(\frac{bd(\bar{\kappa}) - c}{bd(\bar{\kappa}_1) - c}\right)^{3\bar{B}/b}.$$  

(2.40)

Now observe

$$-\eta A_0^2 - \eta \bar{B} < \bar{B}$$

$$-\eta A_0^2 - (\eta - 2)\bar{B} < 3\bar{B}$$

$$b < 3\bar{B}.$$
Thus $3\bar{B}/b > 1$, so integrating (2.40) gives

$$\frac{1}{b - 3\bar{B}} \left( (bd(\tilde{\kappa}) - c)^{1 - 3\bar{B}/b} - (bd(\tilde{\kappa}_1) - c)^{1 - 3\bar{B}/b} \right) > \frac{d'(\tilde{\kappa}_1)}{(bd(\tilde{\kappa}_1) - c)^{3\bar{B}/b}} (\tilde{\kappa} - \tilde{\kappa}_1)$$

Equivalently,

$$d(\tilde{\kappa}) > \frac{c}{b} + \frac{1}{b} \left( (c - bd(\tilde{\kappa}_1))^{1 - 3\bar{B}/b} + \frac{(b - 3\bar{B})d'(\tilde{\kappa}_1)}{(c - bd(\tilde{\kappa}_1))^{3\bar{B}/b}} (\tilde{\kappa} - \tilde{\kappa}_1) \right)^{b/(b - 3\bar{B})}.$$

Bringing $1/b$ inside the parentheses yields (2.35).

**Corollary 2.19.** If $K \geq d^{-1}(c/b)|\Omega|$, then the system is linearly stable.

### 2.3 Numerical implementation

First we describe our procedure for solving (2.14-2.15) numerically. Divide the two-dimensional torus $\Omega$ into an equally spaced square grid with $N$ nodes on each side, distance $h$ apart. Fixed time steps suffice. Let $(A^n, \rho^n, \kappa^n)$ denote the state of the system at time step $n$. First we outline the finite-difference scheme used to find $A^{n+1}$ and $\rho^{n+1}$ from $(A^n, \rho^n, \kappa^n)$, and then we present the optimization method to compute $\kappa^{n+1}$ from $(A^{n+1}, \rho^{n+1})$.

#### 2.3.1 Semi-implicit spectral method

To calculate $A^{n+1}$ and $\rho^{n+1}$ from $(A^n, \rho^n, \kappa^n)$, we use a finite difference scheme closely resembling the one described in [68]. However, we track an attractiveness-normalized criminal density $z = \rho/A$ at each step instead of the criminal density $\rho$. Equations
\( \frac{\partial A}{\partial t} = \eta \Delta A + d(\kappa)A^2 z - (A - A_0), \) \hfill (2.41) 
\( \frac{\partial z}{\partial t} = \Delta z - \left( \frac{(1 + \eta)\Delta A - A + A_0}{A} - d(\kappa)A \right) z - d(\kappa)Az^2 + d(\kappa)\frac{B}{A}. \) \hfill (2.42)

As Sun [74] observes, this transformation banishes the advective term in (2.15), eliminating concerns about upwinding.

Following [68] we solve equations (2.41-2.42) numerically by a semi-implicit method, using spectral methods for the implicit part. Specifically, we solve

\[
\left( \frac{1}{dt} + 1 - \eta \Delta \right) A^n + 1 = Q(A^n, z^n, \kappa^n), \] \hfill (2.43)
\[
\left( \frac{1}{dt} - \Delta \right) z^{n + 1} = R(A^n, z^n, \kappa^n), \] \hfill (2.44)

where

\[
Q(A, z, \kappa) = \frac{1}{dt} A + d(\kappa)A^2 z - (A - A_0),
\]
\[
R(A, z, \kappa) = \frac{1}{dt} z + \left( \frac{(1 + \eta)\Delta A - A + A_0}{A} - d(\kappa)A \right) z - d(\kappa)Az^2 + d(\kappa)\frac{B}{A}.
\]

Letting the circumflex denote the Fourier transform, (2.43-2.44) is equivalent to

\[
\hat{A}_{i,j}^{n+1} = \frac{\hat{Q}_{i,j}}{1/dt + 1 + 2\eta(2 - \cos 2\pi i/N - \cos 2\pi j/N)/h^2},
\]
\[
\hat{z}_{i,j}^{n+1} = \frac{\hat{R}_{i,j}}{1/dt + 2(2 - \cos 2\pi i/N - \cos 2\pi j/N)/h^2},
\]

We let \( N \) be a power of 2 so that \( \hat{Q} \) and \( \hat{R} \) can be computed efficiently from \( Q \) and \( R \) using the fast Fourier transform, as can \( A \) and \( z \) from \( \hat{A} \) and \( \hat{z} \).
2.3.2 Reduction of the optimization problem

We turn to the optimization problem, which in the discretized context is to find \( \kappa^{n+1} \in \mathbb{R}^{N \times N} \) solving

\[
\kappa^{n+1} = \arg \min \left\{ \sum_{i,j=1}^{N} d(k_{i,j}) \rho^{n+1}_{i,j} A^{n+1}_{i,j} : k_{i,j} \geq 0, \sum_{i,j=1}^{N} k_{i,j} = K \right\}.
\]

Efficiency will be important, as the optimization problem must be solved anew in every time step. The strategy will be to reduce the problem to solving a nonlinear equation in a single (integer) variable and then using a combination of iterative methods to solve this equation. This recalls the dual formulation in section 2.2.2 above. When \( d(k) = e^{-k} \), the form of the nonlinear equation will suggest a simpler algorithm.

Rearrange the \( N^2 \) values of \( \rho^{n+1}_{i,j} A^{n+1}_{i,j} \) into a one-dimensional vector \( f \in \mathbb{R}^{N^2} \), so that its values are in descending order. If \( \alpha \) is the coordinate map mediating the two \( (f_{\alpha(i,j)} = \rho^{n+1}_{i,j} A^{n+1}_{i,j}) \), then it is natural to define another vector \( \xi \in \mathbb{R}^{N^2} \) by \( \xi_{\alpha(i,j)} = \kappa^{n+1}_{i,j} \); that is, \( \xi \) is the rearrangement of \( \kappa^{n+1} \) matching the rearrangement of \( f \). We can recast the optimization problem in terms of \( \xi \): we want to find \( \xi \in \mathbb{R}^{N^2} \) minimizing \( \sum_{\alpha=1}^{N^2} d(\xi_{\alpha}) f_{\alpha} \) subject to \( \xi_{\alpha} \geq 0, \sum_{\alpha=1}^{N^2} \xi_{\alpha} = K \).

In the optimal configuration there will be a \( \beta \) such that \( \xi_{\alpha} > 0 \) if \( \alpha < \beta \) and \( \xi_{\alpha} = 0 \) if \( \alpha > \beta \). Any police deployed to the right of such a \( \beta \) would be more effective deployed to the left of \( \beta \). If \( \beta \) is known, we can restrict consideration to the dimensions in which \( \xi \) is positive. A straightforward application of Lagrange multipliers then gives that \( d'(\xi_{\alpha}) f_{\alpha} \) must be the same for all \( \alpha \) between 1 and \( \beta \). We can use this to read off the values of \( \xi \):

\[
\xi_{\alpha} = \begin{cases} (d')^{-1}(d'(0)f_{\beta}/f_{\alpha}) & \text{if } \alpha \leq \beta, \\ 0 & \text{otherwise.} \end{cases} \quad (2.45)
\]

Determining \( \xi \) is thus a matter only of solving the nonlinear resource constraint.
equation for $\beta$:

$$G(\beta) := \sum_{\alpha=1}^{\beta} (d')^{-1}(d'(0)\frac{f_{\beta}}{f_{\alpha}}) = K.$$  \hspace{1cm} (2.46)

We can then recover $\kappa^{n+1}$ if we know the coordinate map $\alpha(i, j)$. The above equation has at most one solution, because $G$ is increasing:

$$G(\beta + 1) - G(\beta) = (d')^{-1}(d'(0)\frac{f_{\beta+1}}{f_{\beta+1}})$$

$$+ \sum_{\alpha=1}^{\beta} (d')^{-1}(d'(0)\frac{f_{\beta+1}}{f_{\alpha}}) - (d')^{-1}(d'(0)\frac{f_{\beta}}{f_{\alpha}})$$

The extra $\beta + 1$ term is 0. By construction $f$ is decreasing, and $(d')^{-1}$ is increasing because $d'$ is. Thus $G(\beta + 1) - G(\beta) > 0$.

It is of course unlikely that we will find a $\beta$ solving (2.46) exactly. Rather, there will be some $\beta$ for which $G(\beta) < K$ but $G(\beta + 1) > K$. If we then set $\xi$ as in (2.45), there will be some police left over, which for convenience we will distribute evenly among those $\xi_{\alpha}$ for which $\alpha \leq \beta$. We do not expect this small liberty we’ve taken to affect the results meaningfully.

As we noted above, to use this reduction we need to determine $\alpha(i, j)$. The first time we do this we can expect it to take $O(N^2 \log N)$ floating point operations using the quicksort algorithm on $f$. However, if the time step is small, $\alpha$ should not change very much between time steps. Thus if we use the previous time step’s $\alpha$ as an initial guess we can expect quicksort to finish in less than $O(N^2 \log N)$.

It may be that $K$ is so large that (2.46) has no solution for $\beta \leq N^2$. In this scenario the police will deploy to every grid point. It is still the case that $d'(\xi_{\alpha})f_{\alpha}$ is the same for all $\alpha$; in fact, they equal $\lambda$, the Lagrange multiplier of the $\ell^1$ constraint. The problem in this case is to solve

$$H(\lambda) := \sum_{\alpha=1}^{N^2} (d')^{-1}(-\lambda/f_{\alpha}) = K$$
for $\lambda$; then $\xi_\alpha = (d')^{-1}(-\lambda/f_\alpha)$. This is a problem in a continuous variable, and standard iterative methods suffice. They are unfortunately expensive because we have to sum all the way to $N^2$ at each iteration. For the following we restrict consideration to the harder scenario in which $\beta < N^2$.

### 2.3.3 General deterrence: a discrete false position method

We now present the iterative method to solve (2.46). Our main workhorse will be the false position method, but we will use the secant method to initialize a window of an appropriate size and bisection if the false position method stalls, as it can near the solution.

We begin with an initial guess $\beta^0$. Suppose $G(\beta^0) < K$; then we know $\beta \in [a^0, b^0]$, where $a^0 = \beta$ and $b = N^2$. If $\beta^0$ is a good initial guess, then information around $\beta^0$ will be more helpful than information around $N^2$. Therefore, we hold off on beginning false position at first. We compute $\beta^1$ using the secant method, with $G(\beta^0 + 1) - G(\beta^0)$ as the slope. If $G(\beta^1) > K$, we let $a^1 = a^0$, $b^1 = \beta^1$, and switch to the false position method. Otherwise, we let $a^1 = \beta^1$, $b^1 = b^0$, and repeat until $G(\beta^n) > K$. The same method works mutatis mutandis for when $G(\beta^0) > K$; in this case we repeat until $G(\beta^n) < K$, when we set $a^n = \beta^n$.

Once we have $b^n < N^2$, we use a discrete false position method to narrow the window $[a^n, b^n]$ progressively. The false position method applied to a function of a discrete variable can stall as the window narrows to the solution; that is, it can select $[a^{n+1}, b^{n+1}] = [a^n, b^n]$. In practice we found this happened a significant minority of the time. In these cases we switch to bisection to guarantee convergence.

We have no reason to expect to guess $\beta$ close to correctly a priori, so we can expect this algorithm to be expensive the first time it is run in any given simulation. However, we do not expect $\beta$ to change much between successive time steps, so using the prior time step’s solution as an initial guess should speed up successive runs of
2.3.4 Exponential deterrence: a homomorphism-based method

When \( d(k) = e^{-k} \), \( G \) has the form

\[
G(\beta) = \sum_{\alpha=1}^{\beta} \log\left(\frac{f_\alpha}{f_\beta}\right).
\]

Computing \( G(\beta) \) directly requires \( \beta \) logarithm evaluations, so we would prefer to evaluate it as few times as possible. In fact we can cut down the work significantly by observing the following consequence of the fact that the logarithm is a homomorphism:

\[
G(\beta) = G(\beta - 1) + \beta \log\left(\frac{f_{\beta-1}}{f_{\beta}}\right).
\] (2.47)

This means that starting by calculating \( G(1) \) and incrementing until \( G(\beta) \geq K \) takes about \( \beta \) logarithm evaluations and \( 3\beta \) additional floating point operations, a very reasonable cost. We therefore simply compute \( G(\alpha) \) recursively until we find \( G(\alpha) > K \), and set \( \beta = \alpha - 1 \).

2.4 Numerical results

We present several results from runs of the finite difference scheme. First we present several solution patterns and discuss them qualitatively. Then we consider patterns of police distribution for different values of \( K \), the total amount of police. Finally we return to Proposition 2.8 and consider the relationship between \( K \) and \( \epsilon \), the amplitude of a Fourier perturbation off the homogeneous steady state.
2.4.1 Solution behavior

We present the results of several runs of the numerics described in the previous sections. In each case we used grid size $N = 256$, grid spacing $dx = \frac{1}{4}$, and periodic boundary conditions. The length of the time step varied between 0.02 and 0.5 depending on observed stability criteria.

We observed three regimes of behavior, illustrated in Figure 2.3. In general, hot spots respond to the introduction of police by expanding in radius but decreasing in height. When there are few police, the hot spots widen slightly and then remain at equilibrium (Figure 2.3b). When there are many police, the hot spots all overlap, and the system approaches a spatially homogeneous equilibrium (Figure 2.3d). These states correspond to the two kinds of equilibrium states observed in the model without police, namely hot spots and no hot spots. However, a moderate number of police can produce an intermediate state in which hot spots merge only with a few neighbors, creating a system of persistent “warm worms” with a diffuse interface into worm-like cold regions of little crime (Figure 2.3c). In this case the police and criminals both remain within the warm regions. We have observed this behavior with several choices of deterrence function $d$.

The third regime is absent under some parameter choices. Figure 2.4 shows two plots of a system with two close values of $K$ ($K = 0.66$ for (b) and $K = 0.67$ for (c)). One produces hot spot flattening and the other hot spot dissipation, with no apparent intermediate regime of warm worms. Figure 2.4 suggests a geometric explanation: the hot spots are distributed so uniformly (compare with Figure 2.3) that a hot spot cannot collide with one neighbor without colliding with them all.

Qualitatively, the “warm worms” solution resembles well known phase-separated solutions of the Cahn-Hilliard equation [9]. In that case, the phase separation arises from the double-well Ginzburg-Landau potential. In our equations, however, the only equilibrium, stable or otherwise, is given by (2.23). Moreover, the values of $A$, $\rho$, and
Figure 2.3: Plots of the criminal density $\rho$ from numerical solutions of (2.14-2.16). Here $d(k) = e^{-k}$. (a) A near-steady-state solution with $\eta = 0.03$, $A_0 = 0.5$, $\overline{B} = 2.5$, and no police, same as Figure 2.2. (b) The same system at time 500 after $K = 300$ police were introduced at time 0. (c) $K = 500$, $t = 500$. (d) $K = 700$, $t = 500$.

Figure 2.4: Plots of the criminal density $\rho$ from numerical solutions of (2.14-2.16) Here $d(k) = e^{-k}$. (a) A near-steady-state solution with $\eta = 0.05$, $A_0 = 1.1$, $\overline{B} = 2.9$, and no police. (b) The same system at time 500 after $K = 0.66$ police were introduced at time 0. (c) $K = 0.67$, $t = 500$. 

40
Figure 2.5: Data gathered from solutions of (2.14-2.16), $\eta = 0.03$, $A_0 = 0.5$, $\overline{B} = 2.5$. For different values of $K$, the maximum value of $\kappa$ at $t = 1000$ is plotted for two classes of solutions. The solid line comes from solutions seeded with stable hot spots. The dashed line comes from solutions seeded with the homogeneous equilibrium values (2.23). Inset pictures are snapshots of $A$.

$\kappa$ at the worms’ peaks are greater than the equilibrium values, and the values at the troughs are less. The dynamics here are clearly quite different from the Cahn-Hilliard dynamics.

The “warm worms” regime does not appear stable; indeed, once the topology settles we observe a slow drift in the system, in the worms’ prevailing direction. It is possible that the worms regime is a metastable state that in the long run will approach either the first or third regime (that is, flattened hot spots or homogeneous equilibrium). This convergence has not occurred on the time scales we have considered.

### 2.4.2 Dynamics of $\kappa$

The three regimes are also reflected in Figure 2.5. Here our initial condition resembles Figure 2.2, a state of stable hot spots found by running the original Short model.
\( t = 0 \) we introduce police and run until \( t = 1000 \), varying the value of \( K \) between runs. We then plotted the maximum value of \( \kappa \) in each simulation against \( K \), yielding the solid line. We repeated this process starting from a homogeneous initial condition, as well; this is the dashed line.

The hot spot flattening regime ends at about \( K = 420 \), when \( \max \kappa \) abruptly turns downward. The warm worms regime ends at around \( K = 580 \), when we get total hot spot suppression and the two branches merge. The bumpy pattern in the intermediate regime is consistent with the notion that warm worms is not a stable regime. Sudden changes in \( \max \kappa \) occur at transition points in the topology of the worms, as shown in the inset images.

We have not plotted \( \min \kappa \) as part of Figure 2.5. For the dashed line minimum and maximum coincide, while for the solid line the minimum is 0 until the branches merge, at which point it joins them. In all our simulations, if \( \min \kappa \) ever exceeds 0 the system ultimately converges to homogeneous equilibrium.

### 2.4.3 Return to linear stability

We now investigate Proposition 2.8 numerically. For a given value of \( K \), consider the initial condition \( A(x, 0) = \tilde{A} + \epsilon f(x), \rho(x, 0) = \tilde{\rho} + \epsilon f(x) \), where \( \epsilon \) is a small positive parameter, \( f \) is a Fourier perturbation, and \( \tilde{A} \) and \( \tilde{\rho} \) are the homogeneous equilibrium values of \( A \) and \( \rho \) for the given parameters, including \( K \), given in (2.23). Proposition 2.8 tells us that, for a small enough value of \( \epsilon \), \( \kappa \) will be nonzero everywhere. The only steady states we have observed in which the police deploy everywhere is the homogeneous steady state; that is, in all observed cases where \( \kappa \) is eventually supported throughout \( \Omega \), \((A, \rho)\) converges to \((\tilde{A}, \tilde{\rho})\).

We can ask then, given \( K \), what critical value of \( \epsilon \) separates the regimes where the system converges to \((\tilde{A}, \tilde{\rho})\) or remains spatially heterogeneous. Figure 2.6 shows this critical value \( \epsilon^*(K) \) for several values of \( K \). We found this value numerically, by
doing a binary search through different candidate values and then running the scheme described above. The relationship between $K$ and $\epsilon^*(K)$ appears linear.

### 2.4.4 Comparison with the discrete model

We now implement police within the discrete, stochastic crime model introduced in [68] and described briefly in section 2.1.2. As in the continuous model, police should decrease the rate at which new criminals enter the system and discourage existing criminals from committing crimes. The former effect is accomplished by replacing the constant criminal birth parameter $\Gamma$ in (2.3) with the spatially varying $d(\kappa(x,t))\Gamma$. The latter effect is accomplished modifying the strike probability specified in (2.1):

$$p_s(x,t) = 1 - e^{-d(\kappa(x,t))A(x,t)dt}.$$
Figure 2.7: Plots of $A$ from numerical solutions of the discrete system with police, $A_0 = \frac{1}{30}$, $\eta = 0.03$, $\omega = \frac{1}{17}$, $\theta = 0.56$, $\Gamma = 0.02$, $d(k) = e^{-k}$. (a) Stationary hot spots without police. (b) The same system at $t = 400$ after $K = 300$ police were introduced at $t = 0$. (c) $K = 500$, $t = 400$. (d) $K = 700$, $t = 400$.

All that remains is the rule for choosing $\kappa$. In the discrete model there is no continuous criminal density $\rho$, only the integer-valued number of criminals $n$. Using $nA$ in the place of $\rho A$ in the objective function for $\kappa$ produces a discontinuous $\kappa$ whose support is highly irregular. It also implies that the police know where every criminal is at any given time, rather than the overall distribution of criminals represented by $\rho$. Instead we use $A^2$, which like $\rho A$ is steeper than $A$ but follows its pattern of high and low values. At time $t$, $\kappa$ is chosen by

$$
\kappa = \arg \min \left\{ \sum_{i,j=1}^{N} d(k_{i,j})A_{i,j}(t)^2 : k \in \mathbb{R}^{N \times N}, k_{i,j} \geq 0, \sum_{i,j=1}^{N} k_{i,j} = K \right\}.
$$

The algorithms described in section 2.3 can be used here, with $f = A^2$ instead of $f = \rho A$.

Figure 2.7 shows snapshots of the attractiveness $A$ from numerical simulations of the discrete model with police. We begin with (a), a state with stationary hot spots.
Then in (b), (c), and (d) we add 300, 500, and 700 police, respectively, and let time run, as in Figure 2.3. We see several characteristics shared with the continuous system as shown in Figure 2.3. In particular, (c) shows a “warm worms” regime arising in the discrete model, as well. The worms evolve differently than in the continuous system because of the discrete system’s stochastic nature.

2.5 The radial problem for small police forces

The first behavior regime (flattened hot spots) is represented in Figure 2.5 to the left of \( K \approx 400 \). For small \( K \) the police are concentrated entirely within the hot spot centers, and the increase in \( \max \kappa \) is faster than \( K^{1/2} \). To better understand this small-\( K \) behavior we focus on the case of a single hot spot in steady state when total police resources \( K \) is small. For simplicity we take \( d(k) = e^{-k} \) for this section. As in [44], replace \( \rho \) with the transformed variable \( V = \rho/A^2 \) to simplify the ODEs. Now \( d(\kappa)V A^3 \) represents the total crime at a location.

2.5.1 Restatement as a boundary-value problem

The governing equations are radially symmetric, so any such solution should be radially symmetric. Changing the spatial coordinates to radial and assuming a steady state, we have the ODE system

\[
0 = (rA_r)_r + d(\kappa)V A^3 - A + A_0
\]
\[
0 = (rV_r A^2)_r - d(\kappa)V A^3 + d(\kappa)\overline{B}
\]
\[
\kappa = (d')^{-1}(-\lambda/\rho A)\chi_{U(\lambda,\rho A)}
\]

where \( \lambda > 0 \) satisfies

\[
\int_{U(\lambda,\rho A)} (d')^{-1}(-\lambda/\rho A)r \, dr = K.
\]

\( K \) can be rescaled by \( 2\pi \) without loss of generality.
The notion is that we have a hot spot centered at the origin, so total crime should be decreasing in \( r \) until it reaches a trough, say at \( r = \beta \), whose value would in the full two-dimensional system be determined by the proximity of other hot spots. Because \( VA^3 \) is decreasing and \( K \) is small, we can predict that \( U(\lambda, \rho A) \) will be an interval with one endpoint at the origin and the other endpoint at some \( b \in (0, \beta) \). Furthermore, we can take advantage of Corollary 2.7 to simplify \( d(\kappa)VA^3 = V(b)A(b)^3 \) within \((0, b)\). We therefore have a boundary-value problem:

\[
0 = \frac{2}{r}(rA_r)_r + V(b)A(b)^3 - A + A_0 \quad \text{if } 0 < r < b, \tag{2.48}
\]

\[
0 = \frac{1}{r}(rV_rA^2) - V(b)A(b)^3 + \frac{B}{VA^3} \quad \text{if } 0 < r < b, \tag{2.49}
\]

\[
0 = \frac{2}{r}(rA_r)_r + VA^3 - A + A_0 \quad \text{if } b < r < \beta, \tag{2.50}
\]

\[
0 = \frac{1}{r}(rV_rA^2)_r - VA^3 + B \quad \text{if } b < r < \beta, \tag{2.51}
\]

\[
K = \int_0^b \log \left( \frac{VA^3}{V(b)A(b)^3} \right) r \, dr \tag{2.52}
\]

The boundary conditions are \( A'(0) = V'(0) = A'(-\beta) = V'(-\beta) = 0 \) and \( A \) and \( V \) continuous across \( b \). We have also specified that the hot spot peaks at 0 and troughs at \( \beta \).

Next we show that regular solutions to this boundary-value problem are regular at the boundary. First note that a simple analog to Theorem 2.2 holds for this radial problem, by parallel reasoning.

**Proposition 2.20.** Any solution \((V, A)\) to the boundary-value problem that is \( C^2 \) in \((0, b)\) and \( C^2 \) in \((b, \beta)\) is \( C^2 \) in \((0, \beta)\).

**Proof.** It suffices to show that \( V \) and \( A \) are \( C^2 \) across \( b \). It is given that \( V \) and \( A \) are continuous across \( b \). Furthermore, because \( d(\kappa) \) is a continuous function of \( r \) by Proposition 2.2, \( d(\kappa)VA^3 - A + A_0 \) is a continuous function of \( r \). Thus \( \frac{2}{r}(rA_r)_r \) is continuous through \( b \), and hence so is \( (rA_r)_r \). This means that the antiderivative \( rA_r \) is \( C^1 \) around \( b \), and hence \( A_r \) is \( C^1 \) at \( b \). Thus \( A \) is \( C^2 \) around \( b \).
Likewise, $-d(\kappa)VA^3 + d(\kappa)\overline{B}$ is continuous at $b$, so $(rV_rA^2)_r$ is. Thus $V_rA^2$ is $C^1$. We just showed $A$ is $C^2$, so $V_r$ is $C^1$. Thus $V$ is $C^2$.

In particular, Proposition 2.20 allows us to use a similar numerical scheme to that used on the full 2D problem without treating the boundary specially. We outline the numerical method in the next section.

We conclude by noting briefly that (2.48) has a general solution in terms of a Bessel function of the first kind, namely

$$A(r) = A_0 + V(b)A(b)^3 + (A(0) - A_0 - V(b)A(b)^3)J_0(\frac{ir}{\sqrt{\eta}}).$$

Taylor coefficients for a series solution for $V$ in $(0, b)$ can be derived from (2.48) and (2.49).

### 2.5.2 Numerical method

We compute solutions to (2.48-2.52) by running a fixed-point iteration, essentially simulating the time-dependent equations until they reach steady state. At each step we must solve a finite difference problem and an optimization problem. Our initial condition is a low trough away from the origin and a peak at the origin, so that with the right choice of parameters the steady state will be a hot spot centered at the origin.

The finite difference solver is based on the same transformation (2.41-2.42) used in the 2D problem. The only extra difficulty is the degeneracy of the radial Laplacian at the origin. Here we employ standard techniques for dealing with degenerate diffusion, following [82]. If our domain is partitioned into a grid by $0 = r_0 < r_1 < \cdots < r_N = \beta$, then the discrete Laplacian operator $\triangle_r$ is defined by

$$\triangle_r u_i = \frac{2}{r_{i+1}^2 - r_i^2} \left( \frac{2r_{i+1}}{r_{i+2} - r_i} (u_{i+1} - u_i) - \frac{2r_i}{r_{i+1} - r_{i-1}} (u_i - u_{i-1}) \right).$$
for \(1 \leq i \leq N - 2\). With Neumann boundary conditions the edge terms are

\[
(\Delta_r u)_0 = \frac{8}{r_1 r_2} (u_1 - u_0),
\]

\[
(\Delta_r u)_{N-1} = -\frac{2}{r_N^2 - r_{N-1}^2} \frac{2r_{N-1}}{r_N - r_{N-2}} 2(u_{N-1} - u_{N-2}).
\]

Offsetting the extra difficulty of implementing degenerate diffusion, the above operator \(\Delta_r\) is tridiagonal, so the Thomas algorithm solves the implicit part of each finite difference step in \(O(N)\) time. Also, it can be defined on grids of variable length.

The optimization solver is essentially the same as in the 2D exponential case, relying on the identity (2.47). The radial solver is actually simpler because total crime is decreasing on \((0, \beta)\), and therefore we needn’t sort the grid points every time step.

Because we plan to investigate the problem for small values of \(K\), we choose the grid to be fine near the origin and coarser away from the origin. This will give us a more precise \(\kappa\) contour.

### 2.5.3 Results

Figure 2.8 shows that the radial problem reproduces the small-\(K\) behavior of the full 2-D problem. The solid line is the same as in Figure 2.5, while the dashed line shows results from the radial simulation. The shapes are nearly identical for small \(K\). They settle to lines of different slope for large \(K\) only because their domains are different. The radial problem’s transition between steady states occurs at about \(K = 550\), when the two-dimensional problem is still in the “warm worms” regime. This may suggest that the warm worms are indeed a metastable state connecting two stable equilibria (flattened hot spots and homogeneity).
Figure 2.8: \( \max \kappa(\cdot, t) \) vs. \( K \) at \( t = 500 \) for the two-dimensional problem (solid) and the radial problem (dashed).

### 2.6 Conclusion

We have introduce into the Short model police behavior that seeks to minimize crime. The equations can be restated as a free-boundary problem, with the boundary being a level set of the crime level \( \rho A \) determined by \( \lambda \), the dual variable associated to the \( L^1 \) constraint on police resources. Given enough police, the linear instability in the original Short model is stabilized. We present a numerical scheme and discuss patterns in the results.

Several of our results suggest that even a temporary infusion of police into a crime-ridden area can reduce crime. In section 4 we provided sudden infusions of police to an area with existing hot spots of crime. For the initial condition considered in Figure 2.5, when \( K \) was greater than about 650 the hot spots were wiped away completely by \( t = 500 \). However, the case of Figure 2.3(c) is instructive: if there are not enough police, the effect can be to reduce crime overall but greatly increase the
area over which criminals act. This recalls the discrete version of the original Short model, which in some parameter regimes had migratory hot spots. This may be an undesirable result for policymakers. Even if the “warm worms” regime is a metastable transition state toward homogeneous equilibrium, the transition would occur over a timescale that policymakers may find unacceptable.

Indeed, the status of the “warm worms” regime remains an open problem. An investigation of the bifurcation theory of the two stable states may provide some clues. Another open problem is the well posedness of the free-boundary problem (2.14-2.18). We saw that police generally stabilize the hot spots, but regularity may be lost across the free boundary. If a two-dimensional analogue to Proposition 2.20 holds, then there is at least some regularity. Sharper results on the linear stability of the two-dimensional system would also be welcome.

Several extensions of this work also present themselves. We assume here that police react instantaneously to changes in the total crime $\rho A$. In practice they receive delayed, incomplete, noisy indicators of the amount of crime. Even with improved information, police officials generally make decisions at regularly scheduled meetings rather than continuously. One possible way to model this would be to force $\kappa$ to be piecewise constant in time, only changing at certain prescribed values of $t$, and to force police to optimize against an outdated picture of $\rho A$.

Though police presence in (2.14-2.18) modifies criminal perception and behavior to some extent, the core of their dynamics still arises from the random walk model described in section 2.1.2. Routine activity theory suggests that criminals are not as strategic as police, but it also suggests that they are at least somewhat strategic as they evaluate their opportunities. A more complete treatment would consider the interplay between criminal and police strategies using the tools of game theory.
CHAPTER 3

Point-process models of human interaction:
Parameter estimation and missing data recovery

3.1 Introduction

3.1.1 Burstiness and Hawkes processes

The ways humans interact has long been a subject of interest. The rise of electronic communication, and particularly social media, has made large data sets of human interactions available. Growing interest in privacy and cybercommunications has led to questions about what can be learned from this data and how it is used.

A natural first question is how to model patterns of social interactions. A point process seems a natural choice, but the simplest point process, the Poisson process, is ill suited to modeling several classes of human activity, including communication. The problem, broadly speaking, is that human activity patterns tend to be “bursty”, that is, more tightly clustered in time than a Poisson process. See, for example, Figure 3.1. Two time series are plotted. Figure 1(a) is taken from the IkeNet data set, which will be discussed in detail later. It shows the times that two particular users sent each other emails. Figure 1(b) is a realization of a Poisson process. The two time series have the same number of events, but the IkeNet time series is more strongly clustered. This suggests a Poisson process is a suboptimal choice for modeling human interactions. Bursty dynamics have been observed in Web browsing [76], emails [1], communications within electronic social networking systems [66], mobile phone calls
Figure 3.1: Two time series. The axis is time, and circles indicate events. Each time series has 68 events. (a) Timestamps of emails sent between IkeNet user 6 and IkeNet user 15. (b) A simulated Poisson process.

[53], FTP requests [60], and even face-to-face interactions [37].

In 1971 Hawkes [30, 31] introduced a class of self-exciting point processes that have come to bear his name. A Hawkes process is a nonhomogeneous Poisson process \( n(t) \) whose intensity is governed by

\[
\lambda(t) = \mu + \sum_{t_i < t} g(t - t_i; \theta). \tag{3.1}
\]

Each \( t_i \) is an event time, \( \mu \) is a deterministic background intensity, and \( g \) is a triggering function specifying how much a recent event increases the intensity, hence the notion of the Hawkes process as self-exciting. Here we note explicitly the dependence of \( g \) on a vector \( \theta \) of parameters because we will estimate these parameters statistically, but we may omit it later for notational convenience. (Nonparametric approaches to estimating \( g \) have also been developed [48, 51].) Likewise we may write \( \lambda(t|\{t_i\}_{i=1}^{n(t)}) \) when we want to emphasize the dependence of \( \lambda \) on the history. The background intensity \( \mu \) can be time-dependent, but we take it as a constant for simplicity. This choice has precedent in seismology [51].

Figure 3.2 shows Hawkes process realizations with \( \mu = 0.15 \) and \( g(t) = 0.5e^{-0.6t} \). The intensity and event times are plotted against time. The Hawkes process events are more tightly clustered in time than the Poisson process of Figure 3.1(b), perhaps
Figure 3.2: Three realizations of a Hawkes process with $\mu = 0.15$ and $g(t) = 0.5e^{-0.6t}$. The horizontal axis is time. Circles indicate events, and the solid curve is the intensity.

more closely resembling Figure 3.1(a).

The Hawkes process appears in the seismology literature as a model for the timing of earthquakes and their aftershocks [56]. As interest in and availability of large data sets of human activities have grown, Hawkes processes have been used to model electronic communications [17], gang crimes [27, 32, 73], and even terrorist and insurgent activity [49, 54].

The constraints on $\mu$ and $g$ are modest. First, we assume that $\mu > 0$. Second, so that the process is self-exciting rather than self-dampening, we assume $g$ is non-negative. Finally, we assume that $\int_0^\infty g(t; \theta)dt < 1$ to ensure that the process is stationary. The importance of this assumption becomes clear when we recognize that $\int_0^\infty g(t; \theta)dt$ is the expected number of immediate descendants of each event. Were it greater than 1, then each event could be expected to give rise to infinitely many others. This would make the process explosive and impossible to simulate repeatedly. It also runs against intuition for our application to emails within a social network.
(all email threads end eventually) or indeed any of the other applications mentioned above.

Our approach recalls that of Stomakhin, Short & Bertozzi’s work on networks of criminal gang rivalries [73]. A gang that has been victimized by a rival will often retaliate, setting off a burst of tit-for-tat crimes. Stomakhin, Short & Bertozzi associate to each pair of rival gangs an independent Hawkes process whose events represent crimes committed by one gang against the other. Then, noting that law enforcement often knows which gang was victimized but not which gang was the perpetrator, they cast the task of solving the crime as a missing data problem, in which a history of gang crimes is known but some of the identities of the gangs involved in particular incidents are hidden. Like Stomakhin, Short & Bertozzi, we will assign independent Hawkes processes to the connections within a social network and solve a missing data problem. However, our variational approach will be different.

Lee et al. [47] also use message data to solve an inverse problem. However, they seek the actors’ positions in physical space rather than their identities. Also their approach is fundamentally Bayesian, while ours is based in maximum likelihood.

### 3.1.2 The IkeNet data set

Between 2010 and 2011, email exchange data was collected from 22 volunteers, all mid-career United States Army officers enrolled in the Eisenhower Leadership Development Program, a one-year graduate program administered jointly by Columbia University and the United States Military Academy. During their enrollment, members of this “Ike” network were given cell phones with which they could access their military email accounts. Of the 22 participants, 19 (90%) were male, and 17 (77%) were Caucasian. At the start of the project they ranged in age from 26 to 33 years.

The data set consists of time stamps and anonymized sender and receiver codes from 8,896 emails sent among the participating officers over a 361-day period. This
Table 3.1: Pairs of officers who exchanged > 100 emails

<table>
<thead>
<tr>
<th>Pair</th>
<th>Number of emails</th>
<th>Pair</th>
<th>Number of emails</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9,18)</td>
<td>1,042</td>
<td>(18,22)</td>
<td>222</td>
</tr>
<tr>
<td>(11,22)</td>
<td>511</td>
<td>(4,13)</td>
<td>134</td>
</tr>
<tr>
<td>(13,17)</td>
<td>302</td>
<td>(9,13)</td>
<td>131</td>
</tr>
<tr>
<td>(11,13)</td>
<td>293</td>
<td>(13,18)</td>
<td>130</td>
</tr>
<tr>
<td>(8,18)</td>
<td>281</td>
<td>(13,22)</td>
<td>120</td>
</tr>
<tr>
<td>(13,15)</td>
<td>223</td>
<td>(3,17)</td>
<td>116</td>
</tr>
</tbody>
</table>

Figure 3.3: Histogram of the number of emails sent between each pair of officers. Only pairs who exchanged fewer than 100 emails are shown; see Table 3.1 for the others.

is a social network with 253 connections. (We include self-connections because the volunteers emailed themselves.) Emails were sent along 250 of these connections.

The emails are by no means distributed evenly among these 250 connections. Table 3.1 lists the 12 pairs of officers who exchanged more than 100 emails. The top pair (9,18) exchanged 1,042 emails, or 11.7% of all the emails in the corpus. Together these top 12 exchanged 3,505 emails, or 39.4% of the corpus. Figure 3.3 is a histogram of the number of emails exchanged among the remaining pairs, all of them less than 100. Many of the pairs of officers exchanged only a few emails, while a few pairs exchanged a substantial proportion of all emails in the corpus, and a few users (13, 18, 22) appear three times or more in this list of highly active pairs. These
observations are consistent with a core–periphery structure, which is a characteristic of many social networks [20].

Fox et al. [28] perform several statistical studies of this data set, including fitting Hawkes processes to the email patterns via maximum likelihood estimation. They find that a Hawkes process model fits the IkeNet data better than a homogeneous Poisson model, as measured by the Akaike information criterion (AIC). They also incorporate the results of a leadership survey administered to the volunteers, revealing more details of the social network.

Our approach differs from Fox et al.’s in two basic ways. First, while they assign an independent Hawkes process to each officer (i.e., each node in the network), we assign one to each relationship between officers (i.e., each edge in the network). This is appropriate to the missing data problem, in which differences in the officers’ relationships matter a great deal. Second, while Fox et al. allow the background rate $\mu$ to change periodically to capture daily and weekly rhythms in email traffic, we take $\mu$ as a constant. We expect this simplification’s impact to be modest, because Fox et al. found only a modest improvement in AIC by moving to a time-varying $\mu$, and because we do not expect it to have much import for our missing data problem.

### 3.2 EM estimation of Hawkes process parameters

First we must discuss fitting the parameters of a Hawkes process to data. We take a maximum-likelihood approach, using an expectation-maximization numerical method to combat the problem’s ill conditioning [51, 78]. Finally, we give several examples for different choices of the triggering function $g$. It is most common in the literature to assume an exponential form for $g$ [17, 28, 32, 52, 73], though other forms are also in use, including power law [19, 57] and the exponential multiplied by a polynomial [58]. Our comparison of exponential and power-law forms suggests that it does not matter which is used, validating the frequent use of the exponential form.
The general problem is, given an interval $[0, T]$ and a time series \( \{t_i\}_{i=1}^{n(T)} \) falling in that interval, to produce statistical estimates \( \hat{\mu} \) and \( \hat{g} \) for the \( \mu \) and \( g \) of the Hawkes process assumed to generate the data. Nonparametric methods of estimating \( g \) exist \cite{48}, but our approach will be to assume a form for \( g \) (in statistical parlance, to adopt a model for \( g \)) and instead estimate \( \theta \), the vector of parameters, together with \( \mu \) using maximum likelihood, yielding parameter estimates \( (\hat{\mu}, \hat{\theta}) \).

The likelihood that a nonhomogeneous Poisson process generated a history \( \{t_i\}_{i=1}^{n(T)} \) is
\[
L = \exp \left( -\int_0^T \lambda(t|\{t_i\}_{i=1}^{n(T)}) dt \right) \prod_{i=1}^{n(T)} \lambda(t_i|\{t_j\}_{j=1}^{i-1}).
\] (3.2)

See \cite{64} for a detailed discussion. It is standard to instead maximize the log-likelihood, which for a Hawkes process as in (3.1) has the form
\[
\log L(\mu, \theta) = \sum_{i=1}^{n(T)} \left( \log \left( \mu + \sum_{j=1}^{i-1} g(t_i - t_j; \theta) \right) - \int_{0}^{T-t_i} g(t; \theta) dt \right) - \mu T.
\] (3.3)

Ozaki \cite{59} treats maximum likelihood estimation of the parameters when \( g \) is exponential.

### 3.2.1 Generating Hawkes process time series

Throughout this section, and again in section 4 when considering simulated networks, we use Lewis’s thinning method \cite{50, 56} to generate artificial Hawkes process time series. Briefly, given a history \( \{t_i\}_{i=1}^n \) at time \( t \), we simulate an independent exponential random variable \( s \) with rate parameter \( \lambda(t|\{t_i\}_{i=1}^n) \). Were this process homogeneous, we would take \( t_{n+1} = t + s \), set \( t = t + s \), and continue. However, because the intensity decays following an event, we only do this with probability \( \lambda(t+s|\{t_i\}_{i=1}^n) / \lambda(t|\{t_i\}_{i=1}^n) \). If we do not, we set \( t = t + s \) and generate a new \( s \). The procedure continues until \( t > T \).
3.2.2 The EM algorithm

To estimate the Hawkes process parameters we adapt the expectation-maximization (EM) algorithm of Veen & Schoenberg [78]. The algorithm maximizes the likelihood (3.2), but indirectly, so as to avoid the conditioning problems of maximizing (3.3) by standard iterative methods.

The algorithm relies on the Hawkes process’s branching structure. The linearity of the intensity process (3.1) allows us to calculate the probability that a given event was triggered by any previous event; otherwise it is a background event. The probability that an event occurring at time $t_i$ is a background event is $\mu/\lambda(t_i)$, and the probability that it was caused by an event that occurred at time $t_j < t_i$ is $g(t_i - t_j)/\lambda(t_i)$.

The EM algorithm alternates between an expectation step and a maximization step. At the $k$th iteration we have an estimate $(\mu^{(k)}, \theta^{(k)})$ of the parameters. The expectation step of the $(k + 1)$th iteration uses those parameters to calculate $p^{(k+1)}_{i,i}$ and $p^{(k+1)}_{i,j}$, respectively the probabilities that event $i$ was a background event or was caused by event $j$:

$$p^{(k+1)}_{i,i} = \frac{\mu^{(k)}}{\mu^{(k)} + \sum_{j=1}^{i-1} g(t_i - t_j; \theta^{(k)})},$$

$$p^{(k+1)}_{i,j} = \frac{g(t_i - t_j; \theta^{(k)})}{\mu^{(k)} + \sum_{j=1}^{i-1} g(t_i - t_j; \theta^{(k)})}.$$  

The maximization step maximizes the complete data likelihood of the branching structure. The likelihood of a given structure can be decomposed into independent pieces:

- The number of background events. This is a Poisson random variable (call it $b$) with expectation $\mu T$. Its likelihood is

$$L_1(\mu) = e^{-\mu T} \frac{(\mu T)^b}{b!}.$$
The number of immediate descendants of each event, both background and triggered, given $b$. Let $d_i$ be the number of descendants of event $i$. It is also Poisson, and its expectation is $\int_0^{T-t_i} g(t; \theta) dt$. Lewis & Mohler [48] found that approximating this by $G(\theta) = \int_0^\infty g(t; \theta) dt$ had only a modest impact on the reliability of results, so we adopt this approximation for simplicity. Because each $d_i$ is independent of the others, their joint likelihood is

$$L_2(\theta) = \prod_{i=1}^n e^{-G(\theta)} \frac{G(\theta)^{d_i}}{d_i!}.$$

The timing of the descendant events given $b$ and all the $d_i$. Let $j(i)$ be the event of which $i$ is the immediate descendant, with $j(i) = i$ if $i$ is a background event. The likelihood of event $i$ occurring at time $t_i$ is $g(t_i - t_{j(i)}; \theta)/G(\theta)$ (we again approximate a finite integral of $g$ by $G(\theta)$), so the joint likelihood of all events’ timing is

$$L_3(\theta) = \prod_{i:j(i)<i} \frac{g(t_i - t_{j(i)}; \theta)}{G(\theta)}.$$

The background events are distributed uniformly in $[0,T]$, so their timing does not enter into the likelihood.

The likelihood of the overall branching structure is the product of $L_1(\theta)$, $L_2(\theta)$, and $L_3(\theta)$. The log-likelihood is

$$\ell_c(\mu, \theta) = -\mu T + b \log \mu + b \log T - \log(b!) + \sum_{i=1}^n (-G(\theta) + d_i \log G(\theta) - \log(d_i!))$$

$$+ \sum_{i:j(i)<i} (\log g(t_i - t_{j(i)}; \theta) - \log G(\theta)).$$

We are maximizing with respect to the parameters $(\mu, \theta)$, so we disregard additive terms that are constants in them. Then we take the expectation with respect to the
probabilities calculated in the expectation step:

\[ E^{(k+1)}(\mu, \theta) = -\mu T + (\log \mu) \sum_{i=1}^{n} p_{i,i}^{(k+1)} - nG(\theta) + \sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k+1)} \log g(t_i - t_j; \theta). \]

It is this function that we maximize with respect to \((\mu, \theta)\).

Regardless of the model for \(g\), the maximizing value of \(\mu\) is

\[ \hat{\mu}^{(k+1)} = \frac{\sum_{i=1}^{n} p_{i,i}^{(k+1)}}{T}. \]

The maximizing \(\theta\) satisfies

\[ \nabla G(\hat{\theta}^{(k+1)}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k+1)} \nabla_{\theta} g(t_i - t_j; \hat{\theta}^{(k+1)}) \frac{g(t_i - t_j; \hat{\theta}^{(k+1)})}{g(t_i - t_j; \hat{\theta}^{(k+1)})}. \]  

(3.4)

Fortunately, for both the models we choose for \(g\), (3.4) reduces to tractable algebraic expressions for each component of \(\hat{\theta}^{(k+1)}\).

### 3.2.3 Example: exponential triggering

First, we choose \(g(t; \alpha, \omega) = \alpha \omega e^{-\omega t}\). The \(L^1\) condition on \(g\) is equivalent to \(\omega > 0\) and \(0 \leq \alpha < 1\). The \(\theta\) condition (3.4) reduces to

\[ \hat{\alpha}^{(k+1)} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)}}{n}, \quad \hat{\omega}^{(k+1)} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)} (t_i - t_j)}{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)}}. \]

We generated 50,000 realizations of a Hawkes process with this triggering function, with \(T = 361\), \(\mu = 0.05\), \(\alpha = 0.5\), and \(\omega = 6\). (These values were chosen to correspond with the IkeNet data.) We then estimated the parameters using the EM algorithm. The results are presented in Table 3.2 and Figure 3.4(a). The estimates for the parameters are distributed about their ground-truth values, with a slight rightward skew for \(\mu\) and more pronounced leftward and rightward skews for \(\alpha\) and \(\omega\),
Figure 3.4: Histograms showing the results of EM estimation of model parameters for (a) exponential and (b) power law triggering functions. For each model 50,000 time series were generated. About 1% of the results for $\omega$ and $q$ are omitted because they are outliers that exceed the right limit of the graph.

respectively. Of the 50,000 estimates for $\omega$, 504 or about 1% were greater than 18; these are omitted from the histogram.

3.2.4 Example: power-law triggering

Many human behavior patterns exhibit power-law scaling in inter-event times [1]. Therefore, we now choose $g(t; \alpha, q) = \alpha(q - 1)(1 + t)^{-q}$. This has the same number of parameters as the previous section’s exponential model. The $L^1$ condition on $g$ is
Table 3.2: EM estimation results

<table>
<thead>
<tr>
<th>Model</th>
<th>Parameter</th>
<th>Ground truth</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exponential</td>
<td>$\mu$</td>
<td>0.05</td>
<td>0.05002</td>
</tr>
<tr>
<td></td>
<td>$\alpha$</td>
<td>0.5</td>
<td>0.4733</td>
</tr>
<tr>
<td></td>
<td>$\omega$</td>
<td>6</td>
<td>6.753</td>
</tr>
<tr>
<td>Power law</td>
<td>$\mu$</td>
<td>0.05</td>
<td>0.05095</td>
</tr>
<tr>
<td></td>
<td>$\alpha$</td>
<td>0.5</td>
<td>0.4641</td>
</tr>
<tr>
<td></td>
<td>$q$</td>
<td>3</td>
<td>3.590</td>
</tr>
</tbody>
</table>

equivalent to $q > 1$ and $0 \leq \alpha < 1$. The $\theta$ condition (3.4) reduces to

$$
\hat{\alpha}^{(k+1)} = \frac{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)}}{n}, \quad q^{(k+1)} = 1 + \frac{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)}}{\sum_{i=1}^{n} \sum_{j=1}^{i-1} p_{i,j}^{(k)} \log(1 + t_i - t_j)}.
$$

Again, we generated 50,000 realizations with $T$, $\mu$, and $\alpha$ as above, and $q = 3$. The results are presented in Table 3.2 and Figure 3.4(b). As with the exponential triggering function, estimates for $\mu$ and $\alpha$ are overall close to their ground truths with, respectively, a slight rightward skew and a more pronounced leftward skew. The estimates of $q$ clearly peak around 3 but skew rightward. Of the 50,000 estimates for $q$, 446 or about 0.9% were greater than 11; these are omitted from the histogram.

### 3.2.5 Comparison of exponential and power-law

In practice we may not know the best form of the triggering function to use when modeling a point process. Nonparametric methods are one solution [48]; however, these can be cumbersome, and without enough data they invite overfitting. Instead, we ask whether time series generated by the two triggering functions discussed in sections 3.2.3 and 3.2.4 can be told apart. The triggering functions are plotted together in Figure 3.5. They have the same integral, but the power-law triggering function has a longer tail. One might reasonably expect these two triggering functions to produce different behaviors.

Most of the time we consider the likelihood only in the context of maximizing it
Figure 3.5: Triggering functions. Exponential: $g(t) = 3e^{-6t}$. Power law: $g(t) = (1 + t)^{-3}$.

with respect to the parameters or the model, given a history. But the likelihood has comparative value, as well. Comparing the likelihoods of models or sets of parameters to the maximum likelihood value reveals how much likelihood we lose by adopting suboptimal assumptions.

To wit, we calculate different likelihood values given the 50,000 Hawkes process realizations we generated for each triggering function in sections 3.2.3 and 3.2.4. For each exponential history $H = \{t_i\}_{i=1}^n$, we compute the log-likelihood (3.3) of the EM parameters $(\hat{\mu}_{\text{exp}}(H), \hat{\theta}_{\text{exp}}(H))$ and the exponential ground-truth parameters $(0.05, 0.5, 6)$. We also calculate $(\hat{\mu}_{\text{pow}}(H), \hat{\theta}_{\text{pow}}(H))$, the parameters maximizing the likelihood under a power law model, and compute their likelihood. For comparison we also compute the likelihood for the power-law ground-truth parameters $(0.05, 0.5, 3)$. We then repeat the process mutatis mutandis for each power-law history. In this way we hope to quantify the loss incurred by using the “wrong” model for the triggering function, as compared to the loss incurred by using the “right” model with the “wrong” parameters. Because both models have the same number of parameters, the penalty term of the Akaike information criterion is unnecessary.

Table 3.3 summarizes the results. The numbers are the average loss in log-likelihood
from the maximum by adopting a certain model and parameters across all realizations. The first column is the loss from using the “correct” model and the EM parameters. As expected this is 0 for both models. The second column is the loss from adopting the “incorrect” model but using the likelihood-maximizing parameters given that model. The third column is the loss from using the “correct” model’s ground-truth parameters rather than the likelihood-maximizing parameters. The fourth column is the loss using the “incorrect” model’s ground-truth parameters. We have no reason to expect this last category to perform well; we include it for a sense of scaling.

In both cases, the loss from using the EM parameters assuming the wrong model is significantly less than the loss from using the right model with the ground-truth parameters. To emphasize, these are the parameters that actually generated the histories, and they still are not as good as a certain set of parameters attached to the wrong model (though not every set, as the fourth column makes clear). The clear moral is that selecting the “correct” model is not as important as finding the likelihood-maximizing parameters once a model has been selected. This justifies the common assumption of the convenient exponential form for the triggering function.

### 3.3 The missing data problem

In this section we state the missing data problem and discuss its numerical solution. We take a variational approach, maximizing a discriminant function subject to certain constraints. For the numerics we adapt the curvilinear method of Wen & Yin [81].
3.3.1 Objective functions

Suppose that we have records of $N$ emails sent among a social network of $V$ members, as in the IkeNet data set. But suppose that for some subset of the emails, we do not know who sent or received them. More generally, we want to identify which of the $M$ edges each email in the subset was drawn from. Because $M$ scales with $V^2$, a direct approach enumerating all possibilities and checking them is not scaleable. Instead, we relax the problem as in [73].

Number the $M$ connections from 1 to $M$. (The order does not matter.) The history of events is $H = \{t_i\}_{i=1}^N$. This history is partitioned into $C$, the events for which we know which connection the event happened on, and $I$, the incomplete-information event. The complete set has the obvious partition $C = \bigcup_{m=1}^M C_m$ into the histories associated to each connection.

We present four methods for classifying the incomplete events. The first two are simple, model-free methods based on basic statistics of $H$. The other two are variational methods maximizing a sort of score function. In each case we have what amounts to a family of discriminant functions, one for each of the $M$ connections. The value of the discriminant function for $t_i \in I$ on connection $m$ is $x_{i,m}$. We speak of $x_i$ as the vector of weights associated to $t_i \in I$. Not every $x_i$ need belong to the same space, or even have the same dimension, as the others. We need define $x_{i,m}$ only for those edges $m$ to which $t_i$ could belong. For example, if we know that one of the parties to an email was officer 1, we need not consider the weight on the connection between officers 2 and 3.

The first classification method is a method of modes, which sets $x_{i,m} = |C_m|$. The only dependence on $i$ comes from the fact that we do not set $x_{i,m}$ if message $i$ could not have been sent on connection $m$. The second method is a nearest-neighbor weighting, which weights depending on the proximity in time (forward or backward)
of the nearest known event: \( x_{i,m} = \max\{|t_i - t_j|^{-1} : t_j \in C_m\}. \) These two methods are in a sense dual to one another: the method of modes is a simple, model-free, global method, and the nearest-neighbor method is a simple, model-free, local method. They can serve as benchmarks for the other methods, which assume a Hawkes process model and in so doing incorporate both global and local information.

The third method for \( x_{i,m} \) is a relaxed maximum likelihood method. The likelihood of a given history and parameter set is

\[
L = \left( \prod_{t_i \in I} \lambda_m(t_i) \right) \prod_{m=1}^M \left( \prod_{t_i \in C_m} \lambda_m(t_i) \right) e^{- \int_0^T \lambda_m(t) dt}.
\]

A true MLE approach would find the \( \{m_i : t_i \in I\} \) maximizing the likelihood. However, there are \( M^{|I|} \) possible values, so this approach quickly becomes infeasible as \( M \) and \( |I| \) grow. We instead consider a relaxed problem, in which we maximize the related quantity

\[
L = \prod_{m=1}^M \left( \prod_{t_i \in C_m} \lambda_m(t_i; x) \right) \left( \prod_{t_i \in I} \lambda_m(t_i; x)^{x_{i,m}} \right) e^{- \int_0^T \lambda_m(t; x) dt}
\]

where

\[
\lambda_m(t; x) = \mu_m + \sum_{t_i \in C_m, t_i < t} g(t - t_i; \theta_m) + \sum_{t_i \in I, t_i < t} x_{i,m} g(t - t_i; \theta_m).
\]

If we restrict the vector \( x_i \) to be a Kronecker delta, we recover the original maximum likelihood. The relaxation is in the constraint on each \( x_i : \|x_i\|_2 = 1 \) and \( x_{i,m} \geq 0 \) for all \( m \). In practice we will maximize not \( L \) directly but a quantity that is off by an additive constant from its logarithm, namely

\[
F_{\text{MRL}}(x) = \sum_{m=1}^M \left( \sum_{t_i \in C_m} \log \lambda_m(t_i; x) + \sum_{t_i \in I} x_{i,m} \log \lambda_m(t_i; x) - \sum_{t_i \in I} x_{i,m} G_m(T - t_i) \right),
\]

where \( G_m(t) = \int_0^t g(s; \theta_m) ds \). (MRL here stands for maximum relaxed likelihood.)

\footnote{The maximand can be replaced with \((\delta + |t_i - t_j|)^{-1}\) if some \( t_i \) coincides with some \( t_j \).}
The fourth method is the Stomakhin–Short–Bertozzi (SSB) method outlined in [73]. This essentially maximizes $F_{\text{SSB}}$ defined by

$$F_{\text{SSB}}(x) = \sum_{m=1}^{M} \sum_{t_i \in I} x_{i,m} \lambda_m(t_i; x)$$

subject to similar constraints on each $x_i$. The solution of this problem is a vector with all positive components. That statement is a corollary of the following. Intuitively, it makes sense to redistribute a little weight from a positive component to a zero component, because the benefit scales linearly with the size of the redistribution, while the cost scales quadratically.

**Proposition 3.1.** Let $n \geq 2$, and let $D$ be the portion of the unit sphere in the non-negative orthant of $\mathbb{R}^n$, i.e. $D = \{x \in \mathbb{R}^n : \|x\|_2 = 1, x_i \geq 0 \forall i\}$. Let $f : \mathbb{R}^n \to \mathbb{R}$ be differentiable with all positive partial derivatives on the non-negative orthant. Then there exists $x^* \in D$ maximizing $f$ on $D$, and $\|x^*\|_0 = n$, i.e. every component of $x^*$ is nonzero.

**Proof.** $x^*$ exists because $f$ is continuous and $D$ is compact. Suppose by way of contradiction that $\|x^*\|_0 < n$. Without loss of generality, $x^*_1 = 0$. By assumption $\|x^*\|_2 = 1$, so without loss of generality $x^*_2 > 0$. Define $\xi : [0, x^*_2] \to \mathbb{R}^n$ by

$$\xi_i(t) = \begin{cases} t & \text{if } i = 1, \\ \sqrt{(x^*_2)^2 - t^2} & \text{if } i = 2, \\ x^*_i & \text{if } 3 \leq i \leq n. \end{cases}$$

Then $\xi(t) \in D$ for every $t$. Because $f$ is differentiable there exist $t_0 > 0$ and $h : (0, t_0) \to \mathbb{R}$ such that $h(t) = o(t)$ as $t \to 0$, and if $0 < t < t_0$ then

$$f(\xi(t)) = f(x^*) + t \nabla f(x^*)^T \xi'(0) + h(t).$$
Easy computations show that $\xi'_1(0) = 1$, $\xi'_2(0) = 0$, and $\xi'_i(0) = 0$ if $3 \leq i \leq n$, so

$$f(\xi(t)) = f(x^*) + t \frac{\partial f}{\partial x_1}(x^*) + h(t).$$

By assumption $\frac{\partial f}{\partial x_1}(x^*) > 0$, so there exists $t_1 \in (0, t_0]$ such that if $0 < t < t_1$ then $|h(t)|/t < \frac{1}{2} \frac{\partial f}{\partial x_1}(x^*)$, in which case

$$f(\xi(t)) > f(x^*) + t \frac{\partial f}{\partial x_1}(x^*) - \frac{t}{2} \frac{\partial f}{\partial x_1}(x^*) > f(x^*),$$

contradicting the assumption that $x^*$ maximizes $f$ on $D$. Thus in fact $\|x^*\|_0 = n$. □

This result recalls a familiar observation about the geometry of $\ell^p$ optimization, presented in two dimensions in Figure 3.6. When all partial derivatives are positive, the geometry is as in Figure 3.6(a). If at some point a level set lies tangent to the constraint, or equivalently the gradient is normal to the constraint, then this point is an optimizer. (This is the basis for the theory of Lagrange multipliers.) The partial derivatives are positive, so the level sets have negative slope. In the non-negative quadrant the $\ell^2$ constraint takes every negative number as a slope, so a point of tangency is guaranteed to exist. This is often contrasted with the $\ell^1$ case, where the
constraint takes only one slope and tangency may not occur, as in Figure 3.6(b). (This is why $\ell^1$ optimizers are often sparse, for example as in [10, 22, 23, 65].) However, one can just as easily contrast Figure 3.6(a) with Figure 3.6(c), where the negative sign of one of the partial derivatives produces positively sloped level sets. Because we are not permitted outside the non-negative orthant, we must settle for the solution on the boundary. Figure 3.6(a) corresponds to $F_{SSB}$, and Figure 3.6(c) corresponds to $F_{MRL}$.

Nonetheless, the assumptions that all partial derivatives of $f$ on the non-negative orthant be positive was stronger than necessary. It would have sufficed if, for all $y \in D$ with a zero component $y_i = 0$, $\frac{\partial f}{\partial x_i}(y) > 0$. However,

$$\frac{\partial F_{SSB}}{\partial x_{i,m}} = \mu_m + \sum_{t_j \in C_m} g_m(|t_i - t_j|) + \sum_{t_j \in I; t_j \neq t_i} x_{j,m}g_m(|t_i - t_j|),$$

which satisfies the stronger assumption stated in the proposition except in the trivial degenerative case when some $\mu_m = 0$.

### 3.3.2 Numerical implementation

Computing $x$ for the method of modes and nearest-neighbor method is straightforward. Constrained maximization of $F_{SSB}$ and $F_{MRL}$ requires more care. Both optimizations have the form

$$\max F(x) \text{ s.t. } \|x_i\|_2 = 1 \forall i \text{ and } x_{i,m} \geq 0 \forall i,m.$$  

The forms of $F$ are summarized in Table 3.4. This is a variational approach to the classification problem. Variational methods have had success in various applications, including image processing [14, 15, 65].

Though $F_{SSB}$ was created to approximate the behavior of $F_{MRL}$, the two functions have different properties. For example, $F_{SSB}$ is a quadratic function with all positive coefficients, so within the feasible set all its partial derivatives are positive. This means
Table 3.4: Objective functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSB:</td>
<td>$\sum_{m=1}^{M} \sum_{t_i \in I} x_{i,m} \lambda_m(t_i; x)$</td>
</tr>
<tr>
<td>MRL:</td>
<td>$\sum_{m} \left( \sum_{t_i \in C_m} \log \lambda_m(t_i; x) + \sum_{t_i \in I} x_{i,m} \log \lambda_m(t_i; x) - \sum_{t_i \in I} x_{i,m} G_m(T - t_i) \right)$</td>
</tr>
</tbody>
</table>

that every component of the maximizing $x$ is positive. (See the appendix for a proof. Briefly, it makes sense to redistribute a little weight from a positive component to a zero component, because the benefit scales linearly with the size of the redistribution, while the cost scales quadratically.) Not so for $F_{MRL}$:

$$\frac{\partial F_{MRL}}{\partial x_{i,m}} = \log \lambda_m(t_i; x) + \sum_{t_j \in C_m; t_j > t_i} g_m(t_j - t_i) \frac{\lambda_m(t_j; x)}{\lambda_m(t_i; x)} + \sum_{t_j \in I; t_j > t_i} x_{i,m} g_m(t_j - t_i) \frac{1}{\lambda_m(t_j; x)} - G_m(T - t_i).$$

The two sums are positive, but the logarithm need not be, and $-G_m(T - t_i)$ can easily be the dominant term.

We used a modified version of the curvilinear search described in [81]. In this section we introduce that algorithm, discuss our modifications, and finally present the whole algorithm for reference.

3.3.2.1 Wen & Yin’s curvilinear search

Gradient ascent is the most basic and intuitive iterative method for smooth maximization, but it does not preserve norms. Wen & Yin [81] present a curvilinear adaptation that preserves orthogonal constraints of the form $X^T X = I$, of which our constraint $\|x_i\|_2 = 1$ is a special case. Let $F_{x_i}(x)$ denote the gradient of $F$ with respect to $x_i$, evaluated at $x$. Given $x$ and a step size $\tau > 0$, the method computes the update $y_i(\tau, x)$ according to a Crank–Nicolson-type scheme:

$$y_i(\tau, x) = x_i + \frac{\tau}{2} A(x, i)(x_i + y_i(\tau, x)),$$

(3.5)
where

\[ A(x, i) = F_{x_i}(x)x_i^T - x_iF_{x_i}(x)^T. \]

The form of \( A \) is inspired by work on \( p \)-harmonic flows with spherical constraints [29, 79]. Classical Crank–Nicolson would use \( \frac{1}{2}(F_{x_i}(x) + F_{x_i}(y(\tau, x))) \) as the step direction, where \( y(\tau, x) \) is \( x \) but with \( y_i(\tau, x) \) replacing \( x_i \). However, this does not guarantee the spherical constraint. Proposition 3.4 ensures that Wen–Yin does. First, Proposition 3.2 expresses \( y_i(\tau, x) \) explicitly.

**Proposition 3.2.** Given \( x \) and \( F \), there exists \( \tau_0 > 0 \) such that, if \( 0 < \tau < \tau_0 \), then

\[ y_i(\tau, x) = (1 - \beta_2)x_i + \beta_1F_{x_i}(x), \]  

(3.6)

where

\[ \beta_1 = \frac{\tau}{1 + (\frac{\tau}{2})^2\delta_i(x)}, \]

\[ \beta_2 = (F_{x_i}(x)^T x_i + \frac{\tau}{2}\delta_i(x))\beta_1, \]

\[ \delta_i(x) = \|F_{x_i}(x)\|_2^2 - (F_{x_i}(x)^T x_i)^2. \]

The quantities \( \beta_1 \) and \( \beta_2 \) are well defined, by the following lemma.

**Lemma 3.3.** \( \delta_i(x) \geq 0 \), with equality only if \( x_i \) is a local maximizer of \( x_i \mapsto F(x) \) on the constraint surface.

**Proof.** Because \( \|x_i\|_2 = 1 \), the inequality is the Cauchy–Schwarz inequality, and equality is equivalent to \( x_i \) and \( F_{x_i}(x) \) being parallel. \( \square \)

**Proof of Proposition 3.2.** Of course, one can verify (3.6) by substituting the right-hand side into (3.5). The calculations are straightforward and tedious. Instead we follow the argument of [81, Lemma 4] to demonstrate the origin of (3.6).
There exists $\tau_0 > 0$ such that $I - \frac{\tau}{2}A(x, i)$ is nonsingular if $0 \leq \tau < \tau_0$. Let $U = (F_{x_i}(x), x_i)$ and $V = (x_i, -F_{x_i}(x))$. Then $UV^T = A(x, i)$, and

$$y_i(\tau, x) = (I - \frac{\tau}{2}UV^T)^{-1}(I + \frac{\tau}{2}UV^T)x_i.$$ 

The Sherman–Morrison–Woodbury identity is

$$(I - \frac{\tau}{2}UV^T)^{-1} = I + \frac{\tau}{2}U(I + \frac{\tau}{2}V^TU)^{-1}V^T.$$ 

We verify that this is valid by calculating

$$\det(I + \frac{\tau}{2}V^TU) = \det\begin{pmatrix} 1 + \frac{\tau}{2}F_{x_i}(x)^Tx_i & \frac{\tau}{2}\|x_i\|_2^2 \\ -\frac{\tau}{2}\|F_{x_i}(x)\|_2^2 & 1 - \frac{\tau}{2}F_{x_i}(x)^Tx_i \end{pmatrix}$$

$$= (1 + \frac{\tau}{2}F_{x_i}(x)^Tx_i)(1 - \frac{\tau}{2}F_{x_i}(x)^Tx_i) + (\frac{\tau}{2})^2\|F_{x_i}(x)\|_2^2\|x_i\|_2^2$$

$$= 1 + (\frac{\tau}{2})^2\delta_i(x),$$

which is positive by Lemma 3.3. This means

$$(I + \frac{\tau}{2}V^TU)^{-1} = \frac{1}{1 + (\frac{\tau}{2})^2\delta_i(x)}\begin{pmatrix} 1 - \frac{\tau}{2}F_{x_i}(x)^Tx_i & -\frac{\tau}{2} \\ \frac{\tau}{2}\|F_{x_i}(x)\|_2^2 & 1 + \frac{\tau}{2}F_{x_i}(x)^Tx_i \end{pmatrix}.$$ 

Thus

$$(I - \frac{\tau}{2}UV^T)^{-1} = I + \frac{\beta_1}{2}(F_{x_i}(x), x_i)\begin{pmatrix} 1 - \frac{\tau}{2}F_{x_i}(x)^Tx_i & -\frac{\tau}{2} \\ \frac{\tau}{2}\|F_{x_i}(x)\|_2^2 & 1 + \frac{\tau}{2}F_{x_i}(x)^Tx_i \end{pmatrix}\begin{pmatrix} x_i^T \\ -F_{x_i}(x)^T \end{pmatrix}.$$ 

The remainder of the proof is matrix-vector multiplication. □

Note that $\beta_1$ and $\beta_2$ are well defined because $\delta_i(x) \geq 0$ by the Cauchy–Schwarz inequality.

**Proposition 3.4.** If $\|x_i\|_2 = 1$, then $\|y_i(\tau, x)\|_2 = 1$. 

72
Proof. By Proposition 3.2,

\[ \|y_i(\tau, x)\|_2^2 = (1 - \beta_2)^2\|x_i\|_2^2 + 2\beta_1(1 - \beta_2)F_{x_i}(x)^T x_i + \beta_1^2 \|F_{x_i}(x)\|_2^2 \]

Consider the constituent parts:

\[
(1 - \beta_2)^2\|x_i\|_2^2 = (1 - (F_{x_i}(x)^T x_i + \frac{\tau}{2}\delta_i(x))\beta_1)^2
\]

\[
= 1 + \beta_1^2(F_{x_i}(x)^T x_i)^2 + (\frac{\tau}{2})^2\beta_1^2\delta_i(x)^2 - 2\beta_1 F_{x_i}(x)^T x_i - \tau\beta_1 \delta_i(x) + \tau \beta_1^2 \delta_i(x) F_{x_i}(x)^T x_i,
\]

\[
2\beta_1(1 - \beta_2)F_{x_i}(x)^T x_i = 2\beta_1(1 - (F_{x_i}(x)^T x_i + \frac{\tau}{2}\delta_i(x))\beta_1)F_{x_i}(x)^T x_i
\]

\[
= 2\beta_1 F_{x_i}(x)^T x_i - 2\beta_1^2(F_{x_i}(x)^T x_i)^2 - \tau \beta_1^2 \delta_i(x) F_{x_i}(x)^T x_i
\]

Summing these,

\[
\|y_i(\tau, x)\|_2^2 = 1 - \beta_2^2(F_{x_i}(x)^T x_i)^2 + (\frac{\tau}{2})^2\beta_1^2\delta_i(x)^2 - \tau\beta_1 \delta_i(x) + \beta_1 \|F_{x_i}(x)\|_2^2
\]

\[
= 1 - \tau \beta_1 \delta_i(x) + \beta_1^2(\|F_{x_i}(x)\|_2^2 - (F_{x_i}(x)^T x_i)^2) + (\frac{\tau}{2})^2\delta_i(x)^2
\]

\[
= 1 - \tau \beta_1 \delta_i(x) + \beta_1 \delta_i(x) + (\frac{\tau}{2})^2\delta_i(x)^2
\]

\[
= 1 - \tau \beta_1 \delta_i(x) + \tau \beta_1 \delta_i(x)
\]

\[
= 1.
\]

Finally we verify that Wen–Yin gives an ascent direction.

**Proposition 3.5.** If \(x_i\) is suboptimal, then there exists \(\tau_0 > 0\) such that, if \(0 < \tau < \tau_0\), then \(F(y(\tau, x)) > F(x)\).

**Proof.** Because \(F\) is differentiable, there exists \(\tau_2 > 0\) and \(h : (0, \tau_2) \to \mathbb{R}\) such that
\[
\lim_{\tau \to 0} h(\tau)/\tau = 0 \quad \text{and if } 0 < \tau < \tau_2 \text{ then }
\]

\[
F(y(\tau, x)) - F(x) = \tau F_{x_i}(x)^T \frac{\partial y_i}{\partial \tau}(0, x) + h(\tau).
\]

Implicit differentiation on (3.5) gives

\[
\frac{\partial y_i}{\partial \tau}(\tau, x) = \frac{1}{2} A(x, i)(x_i + y_i(\tau, x)) + \frac{\tau}{2} A(x, i) \frac{\partial y_i}{\partial \tau}(\tau, x),
\]

so

\[
\frac{\partial y_i}{\partial \tau}(0, x) = \frac{1}{2} A(x, i)(x_i + x_i) + 0
= \|x_i\|^2 F_{x_i}(x) - (F_{x_i}(x)^T x_i) x_i
= F_{x_i}(x) - (F_{x_i}(x)^T x_i) x_i.
\]

Thus

\[
F(y(\tau, x)) - F(x) = \tau (F_{x_i}(x)^T F_{x_i}(x) - (F_{x_i}(x)^T x_i)^2) + h(\tau) = \tau \delta_i(x) + h(\tau).
\]

Because \(x_i\) is suboptimal, \(\delta_i(x) > 0\) by Lemma 3.3. There exists \(\tau_0 \in (0, \tau_2)\) such that, if \(0 < \tau < \tau_2\), then \(|h(\tau)|/\tau < \frac{1}{2}\delta_i(x)\). Thus

\[
F(y(\tau, x)) - F(x) > \tau \delta_i(x) - \frac{1}{2} \tau \delta_i(x) > 0.
\]

\[\square\]

3.3.2.2 Inequality constraints

The algorithm in [81] simply sets \(x_i^{(k+1)} = y_i(\tau, x_i^{(k)})\), with some adaptive time stepping for \(\tau\). While this preserves \(\|x_i\|_2\), it does not preserve the signs of the components of \(x_i\). Our inequality constraint \(x_{i,m} \geq 0\) forces us to concern ourselves with the signs.
If each component of \( x^{(k)} \) (the \( k \)th iterate) is positive but some component of \( y_i(\tau, x^{(k)}_i) \) is negative, then there exists a largest \( \sigma \in (0, \tau) \) so that \( y_i(\sigma, x^{(k)}) \) has all non-negative components. This \( \sigma \) is actually straightforward to compute, because each equation of the form \( y_{i,m}(\sigma, x^{(k)}_i) = 0 \) is a quadratic equation in \( \sigma \). However, we found that this technique was slow in practice because it only allows one dimension of \( x_i \) to reach 0 at a time. When \( F = F_{\text{SSB}} \), many components of the maximizer \( x^*_i \) are close to 0, so we would like to allow many of them to reach 0 at once so they can then turn around and find their correct (small, positive) value. When \( F = F_{\text{MRL}} \), many dimensions will ultimately belong to the active set, and we would like to identify several of them at a time if possible. Therefore, we adopt the less elegant but faster method of setting \( z = \max(0, y_i(\tau, x^{(k)}_i)) \), with the max done componentwise, and then redistributing the mass to preserve the \( \ell^2 \) norm, i.e. \( \tilde{x}_i^{(k+1)} = z/\|z\|_2 \).

If we adopt \( x^{(k+1)}_i = \tilde{x}_i^{(k+1)} \), then it may have components that are zero and that will become negative after another iteration of the curvilinear search. If we continue with these components, the algorithm may hang because the projection back to the sphere may become parallel to the curvilinear search direction. We can prevent this if we acknowledge that any dimensions \( m \) for which \( y_{i,m}(\tau, \tilde{x}_i^{(k+1)}) < 0 \) belong to the active set of inequality constraints. Noting from (3.6) that \( y_{i,m}(\tau, x) \) and \( F_{x_i}(x) \) have the same sign when \( x_{i,m} = 0 \), we set \( x^{(k+1)}_i = P(x, \tilde{x}_i^{(k+1)})\tilde{x}_i^{(k+1)} \), where \( P(x, \tilde{x}_i^{(k+1)}) \) is the projection onto the subspace of those dimensions \( m \) for which \( \tilde{x}_i^{(k+1)} > 0 \) or \( F_{x_i} > 0 \), with the derivative evaluated at \( x \) except with \( x_i \) replaced with \( \tilde{x}_i^{(k+1)} \). (As we iterate, we also remove dimensions from \( F \) and \( \nabla F \) so that dot products with \( x_i \) still make sense and so that we are not calculating derivatives unnecessarily.)

When \( F = F_{\text{SSB}} \) the solution can have many small positive components. It is possible that at \( x^{(1)}_i \) many components \( x^{(1)}_{i,m} \) are small and positive but have \( y_{i,m}(\tau, x^{(1)}_i) < 0 \), and many others are zero but have \( y_{i,m}(\tau, x^{(1)}_i) > 0 \). These sets of components trade places in \( x^{(2)}_i \), and the next iteration will send it back to very close to \( x^{(1)}_i \). If enough components keep “trading places” like this it can cause the algorithm to hang without
reaching the stopping criterion. We found that when $|I|$ was large this happened a small but nontrivial percentage of the time. We also found that we could eliminate the problem by checking the signs of the components of $x_i$ versus $y_i(\tau,x)$. If most were different, we tried $y_i(\tau/2,x)$, and then $y_i(\tau/4,x)$, and so on until a majority of the signs were preserved.

Once the iteration completes, we need to check that the dimensions we have projected away still correspond to active constraints. If they do not, we project $x^{(k)}$ into a larger space including the inactivated dimensions and resume iterating.

### 3.3.2.3 Stopping criterion

Wen & Yin [81] give a stopping criterion of $\|\nabla F\|_2 < \epsilon$. Our stopping criterion must be different, because we do not expect $\|\nabla F\|_2$ to decrease to 0 as we iterate. (Indeed, as noted above, the components of $\nabla F_{SSB}$ are always positive.) Instead we look for $\nabla F$ to be normal to the constraint surface. Since the constraint surface is a sphere, this means we want $\nabla F \cdot x$ to be large relative to the size of $\nabla F$. Specifically, our stopping criterion is

$$\min_{i \in I} \frac{|F_{x_i}(x_i^{(k)}) \cdot x_i^{(k)}|}{\|F_{x_i}(x_i^{(k)})\|_2} > 1 - \epsilon.$$

The absolute value in the numerator is necessary only if every $F_{x_i}(x_i^{(k)})$ is negative. This can happen when $F = F_{MRL}$ but not when $F = F_{SSB}$.

### 3.3.2.4 Algorithm

while $\max_{i \in I} |F_{x_i}(x_i) \cdot x_i|/\|F_{x_i}(x_i)\|_2 > \epsilon$ do
  for $i = 1 : |I|$ do
    $v = F_{x_i}(x)$
    $\delta = \|v\|_2^2 - (v^T x_i)^2$
    $\beta_1 = \tau/(1 + (\frac{\delta}{2})^2\delta)$
    $\beta_2 = (v^T x_i + \frac{\delta}{2})\beta_1$

76
\[ y = (1 - \beta_2)x_i + \beta_1 F_{x_i}(x) \]

\[ \tau = \tau \]

**while** most components of \( y \) have different signs than \( x_i \) **do**

\[ \tau = \tau/2 \]

\[ \beta_1 = \tau/(1 + (\frac{\tau}{2})^2 \delta) \]

\[ \beta_2 = (v^T x_i + \frac{\tau}{2} \delta) \beta_1 \]

\[ y = (1 - \beta_2)x_i + \beta_1 F_{x_i}(x) \]

end **while**

\[ z = \max(0, y) \] componentwise

\[ \tilde{x} = x \]

\[ \tilde{x}_i = z/\|z\|_2 \]

\[ v = F_{x_i}(\tilde{x}) \]

Let \( P \) project the space of \( x_i \) to the subspace where \( \tilde{x}_{i,m} > 0 \) or \( v_m > 0 \)

\[ x_i = P\tilde{x}_i \]

\[ F_{x_i} = PF_{x_i} \]

end for

end while

**for** \( i = 1 : |I| \) **do**

Let \( Q \) project the space of \( x_i \) into its original, full space

\[ w_i = Qx_i \]

\[ F_{x_i} = QF_{x_i} \]

end for

startover = false

**for** \( i = 1 : |I| \) **do**

\[ v = F_{x_i}(w) \]

**for all** \( m \) in the space of \( w_i \) **do**

**if** \( m \) is not in the space of \( x_i \) and \( v_i > 0 \) **then**

Project \( x_i \) into its own space augmented with dimension \( m \).
startover = true

end if

end for

end for

if startover then

for $i = 1 : |I|$ do

Project $F_{x_i}$ into the space of $x_i$

end for

Return to the start.

end if

3.3.2.5 Practical computing considerations

The most computationally expensive part of our C++ implementation of the algorithm is the computation of the derivative $F_{x_i}$. Care must be taken to minimize this expense. For reference, its components for our two choices of $F$ are

$$
\frac{\partial F_{SSB}}{\partial x_{i,m}} = \mu_m + \sum_{t_j \in C_m} g_m(|t_i - t_j|) + \sum_{t_j \in I; t_j \neq t_i} x_{j,m}g_m(|t_i - t_j|),
$$

and

$$
\frac{\partial F_{MRL}}{\partial x_{i,m}} = \log \lambda_m(t_i; x) + \sum_{t_j \in C_m; t_j > t_i} \frac{g_m(t_j - t_i)}{\lambda_m(t_j; x)} + \sum_{t_j \in I; t_j > t_i} \frac{x_{i,m}g_m(t_j - t_i)}{\lambda_m(t_j; x)} - G_m(T - t_i).
$$

Values of $g_m$ should never be computed “on the fly”; each should be precomputed and stored. Most of these values will be so small that treating them as zero will have a de minimis impact on the results, but avoiding computing them (and computing with them) saves tremendous time. Set a small threshold $\eta > 0$, and compute $g_m(t_i - t_j)$ only if it will exceed $\eta \mu_m/|C_m|$, i.e. if $|t_i - t_j| < g_m^{-1}(\eta \mu_m/|C_m|)$. This adds a layer of dependency tracking, but the savings in floating point operations are well worth it.
When $F = F_{SSB}$, the update formula

$$\frac{\partial F_{SSB}}{\partial x_{i,m}}(x^{(1)}) = \frac{\partial F_{SSB}}{\partial x_{i,m}}(x^{(0)}) + \sum_{t_j \in I; t_j \neq t_i} g_m(|t_i - t_j|)(x_{j,m}^{(1)} - x_{j,m}^{(0)})$$

can save time when recomputing $F_{x_i}$. When $F = F_{MRL}$, a corresponding update formula applies for $\lambda_m(t_j; x)$. The $\lambda$ values should be tracked, while the logarithm should be computed only when it is needed.

### 3.4 Results

Here we present results for different configurations of missing data. First we present results from the IkeNet data set. Then we test the methods on simulated time series on artificial social networks, including some toy networks and some meant to resemble IkeNet. We conclude the section by discussing the results in detail.

In each of our tests we begin with a complete data set, whether it is real (IkeNet) or simulated. Then we knock out some of the information to see whether we can recover it from the rest of the corpus. The information might be a particular email’s sender or receiver, an email’s sender and receiver, or the senders and receivers of several emails. When deleting one record at a time we repeat this for each record in the corpus. When deleting more than one record, exhausting the space of combinations is infeasible, so we take a Monte Carlo approach.

We consider a data recovery method successful when the correct component $x_{i,m}$ has a high weight relative to other components. In particular, we want $x_{i,m}$ to be the greatest component, or perhaps the second or third greatest. This metric was considered previously in [73] based on input from the LAPD. (The context there was solving gang crimes, where narrowing down the list of suspect gangs to three can help detectives.) We also present the results for top 5 and top 10 to showcase a property of the MRL optimizer.
Table 3.5: IkeNet: Predictive power for missing sender by method (|I| = 1)

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 2</th>
<th>Top 3</th>
<th>Top 5</th>
<th>Top 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modes</td>
<td>27.8%</td>
<td>41.1%</td>
<td>50.0%</td>
<td>62.9%</td>
<td>82.0%</td>
</tr>
<tr>
<td>NN</td>
<td>62.9%</td>
<td>75.1%</td>
<td>79.8%</td>
<td>85.3%</td>
<td>92.6%</td>
</tr>
<tr>
<td>SSB</td>
<td>63.1%</td>
<td>74.7%</td>
<td>80.0%</td>
<td>85.8%</td>
<td>93.3%</td>
</tr>
<tr>
<td>MRL</td>
<td>61.1%</td>
<td>70.0%</td>
<td>72.4%</td>
<td>73.3%</td>
<td>73.6%</td>
</tr>
</tbody>
</table>

We estimate the Hawkes process parameters using the techniques described in section 2. The SSB and MRL iterations are seeded with the solution from the nearest-neighbor method.

3.4.1 IkeNet

3.4.1.1 Unidirectional identity loss, one at a time

First we took each email in the corpus and saw whether we could determine who sent it knowing its receiver and the rest of the corpus. Repeating this for each email in the corpus meant 8,896 separate runs with |I| = 1 each time. The average performance is shown in Table 3.5.

Table 3.5 shows that SSB, nearest-neighbor (NN), and MRL guess the correct sender about 60% of the time. There is a clear ranking among them, with SSB outperforming nearest-neighbor and nearest-neighbor outperforming MRL. MRL’s relative performance decreases left to right. The method of modes performs poorer than the other methods.

Table 3.6 shows the results when we repeat the process but try to guess the receiver knowing the sender. The numbers are slightly different, but the same patterns prevail.

3.4.1.2 Unidirectional identity loss, missing proportions

We now consider what happens when larger blocks of data are missing, which will be the case in applications. We selected a percentage of the emails at random and
removed the sender or receiver information (chosen randomly for each email). We then attempted to recover the missing data. We repeated this process for 10,000 Monte Carlo runs at each missing percentage.

Table 3.7 shows the results. As expected, the performance decreases as the missing proportion increases from 5% to 20%, but only by a few percentage points. This demonstrates the methods’ robustness to larger missing blocks of data. Interestingly, MRL overtakes SSB as the missing proportion increases, but only for top 1. The method of modes experiences no degradation. This is not a surprise; it returns the same top pairs shown in Table 3.1 until enough data is missing in the right places.
Table 3.8: IkeNet: Predictive power for bidirectional identity loss ($|I| = 1$)

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 2</th>
<th>Top 3</th>
<th>Top 5</th>
<th>Top 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modes</td>
<td>11.7%</td>
<td>17.5%</td>
<td>20.9%</td>
<td>27.3%</td>
<td>36.7%</td>
</tr>
<tr>
<td>NN</td>
<td>37.9%</td>
<td>51.3%</td>
<td>58.5%</td>
<td>65.6%</td>
<td>73.2%</td>
</tr>
<tr>
<td>SSB</td>
<td>39.6%</td>
<td>51.1%</td>
<td>57.6%</td>
<td>65.3%</td>
<td>73.0%</td>
</tr>
<tr>
<td>MRL</td>
<td>36.4%</td>
<td>47.8%</td>
<td>55.0%</td>
<td>61.4%</td>
<td>66.1%</td>
</tr>
</tbody>
</table>

Table 3.9: IkeNet: Average energy for bidirectional identity loss ($|I| = 1$)

<table>
<thead>
<tr>
<th>Method</th>
<th>$F_{SSB}$</th>
<th>$F_{MRL}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modes</td>
<td>45.82</td>
<td>85.62</td>
</tr>
<tr>
<td>NN</td>
<td>122.39</td>
<td>99.37</td>
</tr>
<tr>
<td>SSB</td>
<td>141.39</td>
<td>99.47</td>
</tr>
<tr>
<td>MRL</td>
<td>118.09</td>
<td>101.01</td>
</tr>
</tbody>
</table>

that the order statistics change.

3.4.1.3 Bidirectional identity loss, one at a time

We repeated the one-at-a-time procedure with deleting both sender and receiver from each email, resulting in bidirectional identity loss. Table 3.8 presents the results. The methods do not perform as well as when only the sender or receiver is missing because instead of choosing among the 22 edges connected to each node they must choose among the 253 edges in the complete graph.\(^2\) Nonetheless the local methods guessed the correct edge about 40% of the time and got in the top 3 about 55-60% of the time. MRL still underperforms, but by less than with unidirectional loss. The method of modes continues to underperform all other methods.

Table 3.9 presents average numerical values of $F_{SSB}$ and $F_{MRL}$ evaluated at the bidirectional identity loss solutions in Table 3.8.\(^3\) Horizontal comparison of the values is meaningless, but vertical comparison is not. The results verify that the SSB and MRL solutions maximize $F_{SSB}$ and $F_{MRL}$, respectively.

\(^2\)Actually there are only 250 edges; as noted above, three pairs of agents exchanged no emails.

\(^3\)The values shown are actually of $F(x) - F_{\text{min}}$ to highlight the differences in scale.
Table 3.10: IkeNet: Predictive power for bidirectional identity loss (\(|I| > 1\))

| \(|I|/N\) | Method | Top 1 % | Top 2 % | Top 3 % | Top 5 % | Top 10 % |
|----------|--------|---------|---------|---------|---------|---------|
| 5%       | Modes  | 11.7%   | 17.5%   | 20.8%   | 27.3%   | 36.7%   |
|         | NN     | 37.6%   | 50.8%   | 57.9%   | 64.9%   | 72.4%   |
|         | SSB    | 38.6%   | 50.4%   | 56.9%   | 64.3%   | 72.2%   |
|         | MRL    | 36.0%   | 47.4%   | 54.4%   | 60.9%   | 65.2%   |
| 10%      | Modes  | 11.7%   | 17.5%   | 20.8%   | 27.3%   | 36.7%   |
|         | NN     | 37.3%   | 50.3%   | 57.2%   | 64.1%   | 71.5%   |
|         | SSB    | 37.5%   | 49.3%   | 55.8%   | 63.2%   | 71.3%   |
|         | MRL    | 35.6%   | 47.0%   | 53.8%   | 60.2%   | 64.4%   |

Table 3.11: IkeNet: Average energy for bidirectional identity loss (\(|I|/N = 5\%\))

| Method | \(F_{SSB}/|I|\) | \(F_{MRL}/|I|\) |
|--------|-----------------|-----------------|
| Modes  | 49.12           | 84.08           |
| NN     | 120.67          | 97.87           |
| SSB    | 147.45          | 97.88           |
| MRL    | 115.53          | 100.12          |

### 3.4.1.4 Bidirectional identity loss, missing proportions

Table 3.10 shows the results of the Monte Carlo approach for larger blocks of missing bidirectional data. Bidirectional is much more intensive computationally than unidirectional, so we present proportions only up to 10\% here. The degradation is again modest (compare with Table 3.8), and the ranking of methods is consistent.

Table 3.11 shows average energy values, normalized by the size of the missing block for comparison with Table 3.9. The values are close, and the same hierarchies are apparent.

### 3.4.2 Simulated time series

We simulate Hawkes processes on two classes of networks. First we consider some toy networks with simple structures. Then we simulate a faux IkeNet (FauxNet) using the IkeNet parameters.
3.4.2.1 Toy networks

We use three different configurations of toy networks. Like IkeNet they have 22 nodes, but a known interaction structure. We assume that \( g \) is exponential with \( \alpha = 0.5, \omega = 6 \), with the background rate \( \mu \) varying to show different levels of interaction.

- **Dense:** All nodes are connected to each other (a complete graph), with a low rate of interaction (\( \mu = 0.03 \)).

- **Sparse:** The nodes are arranged in a ring. Each node is connected to its two neighbors and to the node opposite it in the ring, so that the graph looks like a wheel with spokes (except there is no node at the axle). Interaction rates between connected nodes are high (\( \mu = 0.1 \)). Unconnected nodes do not interact.

- **Pseudosparse:** A complete graph, with high interaction (\( \mu = 0.1 \)) between the nodes connected in the sparse graph and low interaction (\( \mu = 0.03 \)) between other pairs.

Table 3.12 presents the results for Monte Carlo simulation. For each network, we adopted bidirectional identity loss for each record in succession, and then averaged the results over each Monte Carlo simulation. Table 3.12 compares with Table 3.8.

The method of modes performs very poorly here compared with IkeNet, because the toy networks lack the heterogeneity in activity levels evident in Table 3.1 and Figure 3.3. NN, SSB, and MRL perform similarly, as with IkeNet, but here MRL outperforms NN. SSB still outperforms them both. Unsurprisingly, all methods perform better on the sparse network than on the dense network, but the local methods perform very well compared to the method of modes even on the dense network. Interestingly, though the performance of the method of modes on the pseudosparse network is between its performances on the dense and sparse networks, the local methods perform worst on the pseudosparse network. This is because the local methods perform poorer as the number of pairs experiencing a burst of activity at any given
Table 3.12: Toy networks: Predictive power for bidirectional identity loss ($|I| = 1$)

<table>
<thead>
<tr>
<th>Network</th>
<th>Method</th>
<th>Top 1</th>
<th>Top 2</th>
<th>Top 3</th>
<th>Top 5</th>
<th>Top 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense</td>
<td>Modes</td>
<td>1.0%</td>
<td>1.9%</td>
<td>2.7%</td>
<td>4.3%</td>
<td>7.9%</td>
</tr>
<tr>
<td></td>
<td>NN</td>
<td>21.4%</td>
<td>36.5%</td>
<td>47.0%</td>
<td>59.3%</td>
<td>69.0%</td>
</tr>
<tr>
<td></td>
<td>SSB</td>
<td>27.4%</td>
<td>41.6%</td>
<td>50.6%</td>
<td>61.0%</td>
<td>69.7%</td>
</tr>
<tr>
<td></td>
<td>MRL</td>
<td>26.4%</td>
<td>40.9%</td>
<td>49.6%</td>
<td>57.9%</td>
<td>61.9%</td>
</tr>
<tr>
<td>Sparse</td>
<td>Modes</td>
<td>4.5%</td>
<td>8.6%</td>
<td>12.4%</td>
<td>20.1%</td>
<td>37.7%</td>
</tr>
<tr>
<td></td>
<td>NN</td>
<td>36.9%</td>
<td>55.5%</td>
<td>65.0%</td>
<td>72.6%</td>
<td>78.8%</td>
</tr>
<tr>
<td></td>
<td>SSB</td>
<td>40.8%</td>
<td>57.5%</td>
<td>65.8%</td>
<td>73.0%</td>
<td>79.6%</td>
</tr>
<tr>
<td></td>
<td>MRL</td>
<td>39.8%</td>
<td>55.9%</td>
<td>62.0%</td>
<td>63.6%</td>
<td>64.9%</td>
</tr>
<tr>
<td>Pseudospars</td>
<td>Modes</td>
<td>1.5%</td>
<td>2.8%</td>
<td>4.2%</td>
<td>6.7%</td>
<td>12.5%</td>
</tr>
<tr>
<td></td>
<td>NN</td>
<td>17.9%</td>
<td>31.4%</td>
<td>41.5%</td>
<td>54.7%</td>
<td>67.6%</td>
</tr>
<tr>
<td></td>
<td>SSB</td>
<td>23.7%</td>
<td>36.8%</td>
<td>45.8%</td>
<td>57.0%</td>
<td>68.3%</td>
</tr>
<tr>
<td></td>
<td>MRL</td>
<td>23.0%</td>
<td>36.2%</td>
<td>45.1%</td>
<td>54.9%</td>
<td>61.5%</td>
</tr>
</tbody>
</table>

Table 3.13: FauxNet: Predictive power for bidirectional identity loss ($|I| = 1$)

<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 2</th>
<th>Top 3</th>
<th>Top 5</th>
<th>Top 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modes</td>
<td>11.7%</td>
<td>17.5%</td>
<td>21.1%</td>
<td>27.5%</td>
<td>37.0%</td>
</tr>
<tr>
<td>NN</td>
<td>49.4%</td>
<td>60.2%</td>
<td>63.9%</td>
<td>66.8%</td>
<td>70.3%</td>
</tr>
<tr>
<td>SSB</td>
<td>53.6%</td>
<td>63.2%</td>
<td>66.8%</td>
<td>70.1%</td>
<td>74.3%</td>
</tr>
<tr>
<td>MRL</td>
<td>48.5%</td>
<td>60.6%</td>
<td>64.5%</td>
<td>65.9%</td>
<td>66.0%</td>
</tr>
</tbody>
</table>

time increases. This strength of this effect decreases as we move from top 1 to top 10, and indeed this is reflected in Table 3.12.

### 3.4.2.2 FauxNet

As with the toy networks, we took a Monte Carlo approach to FauxNet, the simulated IkeNet, and present results for bidirectional identity loss in Tables 3.13 and 3.14. The method of modes performs almost the same as in IkeNet (see Tables 3.8 and 3.10). The other methods perform better here by several percentage points.
<table>
<thead>
<tr>
<th>Method</th>
<th>Top 1</th>
<th>Top 2</th>
<th>Top 3</th>
<th>Top 5</th>
<th>Top 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modes</td>
<td>11.7%</td>
<td>17.4%</td>
<td>21.0%</td>
<td>27.3%</td>
<td>36.8%</td>
</tr>
<tr>
<td>NN</td>
<td>48.9%</td>
<td>59.4%</td>
<td>63.0%</td>
<td>66.0%</td>
<td>69.4%</td>
</tr>
<tr>
<td>SSB</td>
<td>52.4%</td>
<td>62.0%</td>
<td>65.7%</td>
<td>69.1%</td>
<td>73.4%</td>
</tr>
<tr>
<td>MRL</td>
<td>47.9%</td>
<td>59.8%</td>
<td>63.6%</td>
<td>65.0%</td>
<td>65.1%</td>
</tr>
</tbody>
</table>

### 3.4.3 Discussion

In all our results, the local methods (nearest-neighbor, SSB, and MRL) strongly outperform the purely global method of modes. This suggests that most of the information in these sorts of records is local. Meanwhile, with IkeNet the model-free nearest-neighbor method performs comparably to the variational methods (SSB and MRL) developed in section 3. With the simulated Hawkes process data it underperforms SSB and, in some places, MRL, but not by nearly the margin that the method of modes does. This suggests that the Hawkes process is an imperfect model for real human communication like the IkeNet data, but the loss incurred from these assumptions is modest. On the other hand, the loss in assuming no model at all (i.e. using nearest-neighbor) is also modest and has the virtue of being simpler to implement, understand, and communicate outside technical literature.

The improvement in MRL’s performance as it moves from top 5 to top 10 is considerably lower than it is for the other methods. Figure 3.7 reveals why. It shows a histogram of \( \|x_{MRL}\|_0 \), the number of nonzero components of \( x_{MRL} \), for each bidirectional \( |I| = 1 \) case. The median is 6, and \( \|x_{MRL}\|_0 \leq 5 \) in about 44% of cases. In these cases, if the correct pair is not in the top 5 then it will not be in the top 10, either. SSB, by contrast, always has full \( \ell^0 \) norm (see the appendix for a proof), and even if the correct pair has only a small positive weight it is often larger enough than the other small positive weights to make it to the top 10. Of course, MRL has even fewer positive components in the unidirectional case, explaining why it underperforms less in bidirectional identity loss. Thus SSB’s density is capturing some faint information.
that MRL misses by being so sparse. If a likelihood approach like MRL is to beat SSB it will likely have to mimic this ability.

All the methods except the method of modes perform better on FauxNet than on IkeNet. Furthermore, SSB and MRL perform better relative to nearest-neighbor on the simulated time series than they do on IkeNet data. Both these observations suggest that the Hawkes process is an imperfect model for the behavior driving IkeNet.

3.5 Conclusion

We demonstrated that, when estimating the parameters of a Hawkes process from data, choosing a parameterization for the triggering function is less important than using the correct values of the parameters. We then developed a method for filling in missing data for interactions within social networks and presented some results from the IkeNet data set. The method’s power even when the proportion of missing data increases has implications for security, surveillance, and privacy. In particular, it suggests that access to even a fraction of a complete record can reveal a great deal of information about the remainder, emphasizing the need for robust access controls.

Future work should address how network structure impacts the ability to fill in
missing data. Exogenous information (for example, the leadership relationships among the IkeNet officers) may also be able to boost the method’s power. Future work might also seek an objective function combining MRL’s fidelity to the original likelihood with SSB’s solution density. However, as noted, modeling IkeNet’s email behaviors with Hawkes processes has its limits, so consideration of other classes of self-exciting point processes may be warranted.
CHAPTER 4

Information spread in social networks

4.1 Introduction

The spread of information through social networks, long a topic of interest, is receiving increasing attention as larger data sets become available. Relying in some cases on approaches and results from epidemiology [12, 40, 43, 55, 72], emerging research has begun to identify the important factors. Two key components in determining the dynamics of information spread are the structure of the network [18, 24, 45, 46] and the behavior of the network’s members [26, 37, 45, 77].

Social networks are commonly characterized by the existence of a “core” of densely connected nodes and a “periphery” of relatively remote nodes that are nonetheless connected to the core [20, 34, 43]. Network structure can impact the spread of information or disease in sometimes surprising ways. For example, the highest-degree and highest-centrality nodes are not necessarily the most efficient spreaders; more subtle metrics dominate degree and centrality [43]. Even so-called “superspreaders” may have their relative influence diminished in periods of high activity [6] or negated entirely if some agents within the network act to block the spread [5]. In these cases “gatekeepers” can take on a determinative role [80]. Gatekeepers may want to block diseases, untrue rumors [5], or opinions with which the gatekeepers disagree [67].

The behavior of the nodes can also have important effects on spreading. Human interaction tends to occur in bursts rather than with the sort of even spacing emerging from a Poisson process [13, 37, 53, 66], perhaps because of the way people prioritize
tasks [1]. Burstiness has been observed in both face-to-face [13, 37] and electronic [35, 36, 66] communication and has been studied as a behavior in agent-based models [71]. This factor alone has a large impact on spreading [53, 72, 77]. Also, conversations tend to occur in groups, which in some ways counteracts burstiness [53].

Network structure and human behavior can combine to impact spreading. For example, disease and information are often prevented from taking the most efficient path through the network. The spreading can get trapped in the core, isolated from the periphery by a paucity of contacts. At the same time, the burstiness of communication can create long intervals between opportunities to spread to and through the periphery [40].

4.2 Model specification

We want to model information spread within a social network, where the only source of information is within the network. This could represent friends sharing news or personal information through social media, or a covert network that relies on member-to-member interactions to keep its information secret.

Model the network as a graph with $N$ nodes and $M$ edges, undirected and unweighted (for now). At time $t$, node $n$ has knowledge $k_n(t)$, which for now we take in $[0, 1]$. The (column) vector $k(t)$ has $n^{th}$ component $k_n(t)$. Each node interacts stochastically with the nodes to which it is connected through edges. Namely, interactions along each edge follow an i.i.d. Poisson process with parameter $\lambda$. If we label the $M$ edges 1 to $M$, then edge $m$ has Poisson process $Y_m(t)$.

When two nodes interact, the more informed node shares information with the less informed node. The information transfer is not perfect: the less informed node receives a convex combination of its previous knowledge with its partner’s knowledge.
If nodes $n_1$ and $n_2$ interact at time $t$, and $n_1$ is the less informed, then

$$k_{n_1}(t) = (1 - \alpha)k_{n_1}^-(t) + \alpha k_{n_2}^-(t),$$

where $k_n^-(t) = \lim_{s \to t^-} k_n(s)$ is node $n$’s knowledge just before $t$, and $\alpha \in (0, 1)$ is a global constant.

Total knowledge is not conserved; it can only increase. However, the following maximum principle applies.

**Proposition 4.1.** For all $t > 0$,

1. $\max k(t) = \max k(0)$.
2. If $k_n(0) = \max k(0)$, then $k_n(t) = k_n(0)$.

**Proof.** For part 1, the only mechanic for $k_n(t)$ to change is through interacting with another node. During interactions, the greater node’s knowledge does not change. Therefore, $\max k(t) \geq k(0)$. The reverse inequality follows from the fact that knowledge spreads through convex combinations with parameter $\alpha \in (0, 1)$. This also gives part 2. \qed

### 4.3 Examples

#### 4.3.1 1D nearest-neighbor information sharing

The simplest configuration is to line up the $N$ nodes from node 0 to node $N - 1$ and place edges in between nearest neighbors, so that $M = N - 1$. Let’s have a deterministic initial condition: with probability 1, $k_0(0) = 1$, and $k_n(0) = 0$ for $n > 0$. That is, at time 0, node 0 has all the information, and nobody else knows anything.

For this configuration we can fully characterize $E(k_n(t))$. First, we have a lemma.

**Lemma 4.2.** For all $t$, if $n_1 < n_2$, then $k_{n_1}(t) \geq k_{n_2}(t)$.  

91
Proof. It suffices to prove that \( k_n(t) \geq k_{n+1}(t) \) for all \( n \). Proceed by backward induction on \( n \). The last node, node \( N-1 \) has no information source other than node \( N-2 \), and \( k_{N-1}(0) = k_{N-2}(0) = 0 \). Thus \( k_{N-2}(t) \geq k_{N-1}(t) \) for all \( t \). Now assume \( k_n(t) \geq k_{n+1}(t) \). Then node \( n \) has no information source other than node \( n-1 \). Because \( k_{n-1}(0) \geq k_n(0) \) (with equality holding unless \( n = 1 \)), then, \( k_{n-1}(t) \geq k_n(t) \). \( \square \)

**Proposition 4.3.** Let \( \overline{k}_n(t) = E(k_n(t)) \). Then \( \overline{k}_0(t) = 1 \) and, for all \( n > 0 \),

\[
\overline{k}_n(t) = 1 - e^{-\alpha \lambda t} \sum_{j=0}^{n-1} \frac{(\alpha \lambda t)^j}{j!}.
\]

**Proof.** Proposition 4.1 gives \( \overline{k}_0(t) = 1 \). For \( n > 0 \), we derive a system of coupled differential equations and perform induction on \( n \). By Lemma 4.2, \( k_n(t) \leq k_{n-1}(t) \), so any interaction between \( n \) and \( n+1 \) can only increase \( k_n \). The interactions follow a Poisson process, so

\[
E(k_n(t+\Delta t)|k_n(t)) = (1 - h \lambda)k_n(t) + h \lambda((1 - \alpha)k_n(t) + \alpha k_{n-1}(t)) + O(h^2)
\]

\[
E(k_n(t+\Delta t)|k_n(t)) = k_n(t) + h \alpha \lambda (k_{n-1}(t) - k_n(t)) + O(h^2)
\]

\[
E\left(\frac{k_n(t+\Delta t) - k_n(t)}{\Delta t}\,|\,k_n(t)\right) = \alpha \lambda (k_{n-1}(t) - k_n(t)) + O(\Delta t).
\]

Taking expectations,

\[
\frac{\overline{k}_n(t+\Delta t) - \overline{k}_n(t)}{\Delta t} = \alpha \lambda (\overline{k}_{n-1}(t) - \overline{k}_n(t)) + O(\Delta t).
\]

Taking limits as \( \Delta t \to 0 \),

\[
\overline{k}_n'(t) = \alpha \lambda (\overline{k}_{n-1}(t) - \overline{k}_n(t)).
\]

Now we begin our induction. For \( n = 1 \),

\[
\overline{k}_1'(t) = \alpha \lambda (1 - \overline{k}_1(t)).
\]
The solution with the initial condition $k_1(0) = 0$ is $k_1(t) = 1 - e^{-\alpha \lambda t}$, which corresponds to (4.1). Now assume (4.1) holds for a given $n$. Then

$$k'_{n+1}(t) = \alpha \lambda \left(1 - e^{-\alpha \lambda t} \sum_{j=0}^{n-1} \frac{(\alpha \lambda t)^j}{j!} - k_{n+1}(t)\right)$$

$$e^{\alpha \lambda t} (k'_{n+1}(t) + \alpha \lambda k_{n+1}(t)) = \alpha \lambda \left(e^{\alpha \lambda t} - \sum_{j=0}^{n-1} \frac{(\alpha \lambda t)^j}{j!}\right)$$

$$\frac{d}{dt}(e^{\alpha \lambda t} k_{n+1}(t)) = \alpha \lambda \left(e^{\alpha \lambda t} - \sum_{j=0}^{n-1} \frac{(\alpha \lambda t)^j}{j!}\right).$$

Taking antiderivatives,

$$e^{\alpha \lambda t} k_{n+1}(t) = C_{n+1} + e^{\alpha \lambda t} - \sum_{j=0}^{n-1} \frac{(\alpha \lambda t)^{j+1}}{(j+1)!}$$

$$k_{n+1}(t) = 1 - e^{-\alpha \lambda t} \left(-C_{n+1} + \sum_{j=1}^{n} \frac{(\alpha \lambda t)^j}{j!}\right).$$

The initial condition $k_{n+1}(0) = 0$ gives $C_{n+1} = -1$ for the constant of integration. Thus

$$k_{n+1}(t) = 1 - e^{-\alpha \lambda t} \left(1 + \sum_{j=1}^{n} \frac{(\alpha \lambda t)^j}{j!}\right) = 1 - e^{-\alpha \lambda t} \sum_{j=0}^{n} \frac{(\alpha \lambda t)^j}{j!},$$

which was to be shown.

Figure 4.1 compares the analytical solution of Proposition 4.3 with Monte Carlo simulations of the nearest-neighbor graph. Here there are 101 nodes (at 0 through 100), $\alpha = 0.5$, and $\lambda = 0.5$. Theory and simulation match. By $t = 50$ a "knowledge front" has emerged, with a diffuse boundary between a left-region of knowledge near 1 and a right-region of knowledge near 0. This front moves rightward and becomes more diffuse over time, until the low-knowledge region is gone and the front is slowly squeezed out by the high-knowledge region.

We now find validation for this behavior in a continuum limit. Return for a moment
Figure 4.1: Plots of the expectation value of knowledge $E(k_n(t))$ for nearest-neighbor information sharing, with parameters $N = 101$, $\alpha = 0.5$, $\lambda = 0.5$. The solid curves are analytical solutions from Proposition 4.3, and the circles are Monte Carlo values taken from 10,000 simulations.

\[(4.2)\], before we’ve taken limits to derive the ODE system \[(4.3)\):

\[
\frac{\bar{k}_n(t + h) - \bar{k}_n(t)}{h} = \alpha \lambda (\bar{k}_{n-1}(t) - \bar{k}_n(t)) + O(h).
\]

Now let’s think about the line of nodes being embedded in a physical space. As we send the time step $h \to 0$, we also want to send the “distance” between nodes to 0, prompting more frequent interactions. This corresponds to increasing $\lambda$. Our hydrodynamic limit will be that $\alpha \lambda h$ will stay constant, say at $\theta$. We thus get an equation for $u(x, t) = \bar{k}_{nh}(t)$ as $h \to 0$:

\[
\frac{u(x, t + h) - u(x, t)}{h} = \theta \frac{u(x - h, t) - u(x, t)}{h} + O(h),
\]

which of course limits to

\[ u_t = -\theta u_x, \]

the transport equation. The unique solution with our initial condition is

\[
u(x, t) = \begin{cases} 
1 & \text{if } x \geq \theta t, \\
0 & \text{if } x < \theta t.
\end{cases}
\]
The limiting behavior is thus a front with a sharp boundary, moving at speed $\theta = \alpha \lambda h$. In Figure 4.1, in which $\alpha = 0.5$, $\lambda = 0.5$, and $h = 1$, the front moves at a speed of approximately 0.25, as predicted by the continuum limit. However, before taking the limit the front is diffuse and becomes more so as it moves. In the following section we investigate the behavior of this growing boundary region.

Figure 4.1 makes clear that, in contrast to the sharp boundary of the continuum limit, before the limit the boundary is diffuse and degrades as it moves. First, write (4.1) as

$$k_n(t) = e^{-\alpha \lambda t} \sum_{j=n}^{\infty} \frac{(\alpha \lambda t)^j}{j!},$$

and notice how the asymptotics fall right out of this form. For fixed $t$, increasing $n$ means including fewer terms in the series, so $k_n(t) \to 0$. For fixed $n$, increasing $t$ means the series approaches $e^{\alpha \lambda t}$ (since the full Taylor series’s first $n$ terms contribute less and less), so $k_n(t) \to 1$.

We want to understand the dynamics of the diffuse boundary between the mostly-1 region to the left and the mostly-0 region to the right. One approach is to track the value of $n$ that comes closest to solving $k_n(t) = 0.5, 0.1, \text{ and } 0.9$. This will be difficult analytically because as $t$ grows, the bounds provided by Taylor’s theorem become less precise.

However,

$$k_n(t) - k_{n-1}(t) = e^{-\alpha \lambda t} \frac{(\alpha \lambda t)^n}{n!}.$$

This has a continuous analog, namely

$$f_t(x) = e^{-\alpha \lambda t} \frac{(\alpha \lambda t)^x}{\Gamma(x+1)}.$$

The critical point of $f_t$ is where $k$ is decreasing the fastest for a given $t$. Its inflection points will roughly mark the boundary of the transition region, or at least the distance between the inflection points give an idea of the growth of this region. So doing
calculus on $f_t$ will not track the movement of specific values of $\bar{k}$, but it is more amenable to analysis and is really no less arbitrary.

The critical point of $f_t(x)$ occurs where

$$
\psi_0(x + 1) = \log(\alpha \lambda t),
$$

(4.4)

where $\psi_0$ is the digamma function that is the logarithmic derivative of the gamma function ($\Gamma' = \psi_0 \Gamma$). Its inflection points occur where

$$
(\psi_0(x + 1) - \log(\alpha \lambda t))^2 - \psi_1(x + 1) = 0,
$$

(4.5)

where $\psi_1$ is the trigamma function, $\psi_1 = \psi_0'$. Given $t$, (4.4) and (4.5) are easily solved numerically, say with Newton’s method. Figure 4.2 shows these numerically determined values. As expected, the critical point (circle) roughly indicates the transition region’s center, and the inflection points (squares) roughly indicate its boundaries.

There is no non-negative solution $x$ of (4.4) if $\log(\alpha \lambda t) < \psi_0(1) = -\gamma$, where $\gamma \approx 0.577$ is the Euler gamma. Thus we require

$$
t \geq \frac{1}{\alpha \lambda} e^{-\gamma}
$$

(4.6)

for our analysis to work. $\psi_1$ is positive and decreasing, with $\psi_1(1) = \frac{\pi^2}{6}$, so a necessary
condition for $x \geq 0$ to satisfy (4.5) is
\[
|\psi_0(x + 1) - \log(\alpha \lambda t)| \leq \frac{\pi}{\sqrt{6}}.
\]

For one solution to exist, the condition (4.6) suffices. But we want two solutions, and for this we need the stronger condition
\[
t \geq \frac{1}{\alpha \lambda} e^{-\gamma + \pi/\sqrt{6}}.
\]

The critical point $x_c(t)$ obeys $\psi_0(x_c(t) + 1) = \log(\alpha \lambda t)$. Letting $\tau = \frac{1}{\alpha \lambda} e^{-\gamma + \pi/\sqrt{6}}$ and centering a Taylor approximation at $\tau$ yields
\[
x_c(t) = x_c(\tau) + \frac{1}{\tau \psi_1(x_c(\tau) + 1)} (t - \tau) + O((t - \tau)^2). \tag{4.7}
\]

The inflection points $x_{\pm}(t)$ have the Taylor approximation
\[
x_{\pm}(t) = x_{\pm}(\tau) + x'_{\pm}(\tau)(t - \tau) + O((t - \tau)^2), \tag{4.8}
\]
where
\[
x'_{\pm}(\tau) = \frac{2}{\tau} \frac{\log(\alpha \lambda \tau) - \psi_0(x_{\pm}(\tau) + 1)}{2 \psi_1(x_{\pm}(\tau) + 1) \log(\alpha \lambda \tau) - 2 \psi_0(x_{\pm}(\tau) + 1) \psi_1(x_{\pm}(\tau) + 1) + \psi_2(x_{\pm}(\tau) + 1)}.
\]

$\tau$ depends only on $\alpha$ and $\lambda$, so we can determine $x_c(\tau)$ and $x_{\pm}(\tau)$ numerically by solving (4.4) and (4.5) at $t = \tau$.

Figure 4.3 shows the behavior of the critical points and inflection points over time. For the critical point $x_c(t)$, the circles are analytical values (calculated by Newton’s method on (4.4)), and the solid line is the linear approximation (4.7). There is strong agreement here, as suggested by the transport equation limit. On the other hand, there is modest disagreement between the inflection points’ analytical values (crosses) and linear approximations (dashed line). In particular, they appear to diverge from the
critical point slower than the linear approximations suggest. This may be a second-order effect.

4.3.2 The 4-regular graph

We can also consider what happens when the nodes are arranged into a standard 4-regular graph. Now let the node at (0,0) have knowledge 1 and see how knowledge spreads.

The analysis becomes much harder because each node now has multiple possible sources of information. In one dimension, knowledge always flowed from the left. In two dimensions, it is likely that node (1,0) will get its first knowledge increase from (0,0). However, there is a 12.5% probability that knowledge will flow like (0,0) → (0,1), (0,1) → (1,1), (1,1) → (1,0) before (0,0) and (1,0) interact directly. Other paths are less likely, but their probability does not vanish as $h \to 0$. This means we get many cross-terms in $E(k_n(t))$ of the form $E(|k_n(t) - k_n'(t)|)$, and resolving the absolute values is far from trivial. More on this in the next section.
Figure 4.4: Plots of the expectation value of knowledge $E(k_n(t))$ for nearest-neighbor information sharing on the 4-regular graph, at times $t = 50, 150, \text{ and } 250$, with parameters $N = 16^2$, $\alpha = 0.5$, $\lambda = 0.1$. Shown is the average knowledge taken from 10,000 Monte Carlo simulations.

While analysis is difficult, we can at least do Monte Carlo simulations as in the previous section. This is more expensive computationally than the 1D case because the expense grows with the number of edges. Figure 4.4 shows the results from 10,000 simulations. Again the behavior is characterized by a diffuse interface between a region of almost-1 knowledge (red) and region of almost-0 knowledge, with the interface moving as a front from red to blue.

4.3.3 Erdos-Renyi random graphs: average dynamics

Despite the difficulty of the analysis and of simulating $p$ directly, we can at least approximate it with more Monte Carlo simulations. Figure 4.5 shows the average knowledge in a system as a function of time, from 10,000 Monte Carlo simulations. The graph was determined randomly at the start of each simulation. Knowledge spreads slowly at first, but then the speed picks up as more edges have steeper knowledge gaps. Then the knowledge spread levels off. Unsurprisingly it resembles a logistic curve for the growth of a population, but as we’ve talked about before the underlying dynamics are quite different.
4.4 The Kolmogorov forward equation

If we want to analyze more general graph configurations than 1D nearest-neighbor, we need to understand how the probability density of $k_n(t)$ evolves in time. It will depend on the parameters $\alpha$ and $\lambda$, on the configuration of graphs, and on the initial density.

The Kolmogorov forward equation, or Fokker-Planck equation, is a PDE that describes the evolution of a probability density arising from a random process. Usually the random process is an Ito process (deterministic drift plus Brownian diffusion) but in principle such an equation can be derived for any process, as Denisov et al. [21] do. Their approach yields a PDE for the Fourier transform of the probability density. Following their development for our model yields a differential equation for the Fourier series coefficients of $p : ([0,1]^N \times \mathbb{R}^+) \rightarrow \mathbb{R}^+$, the probability distribution of $k$. Namely,

$$\frac{\partial \hat{p}}{\partial t}(q,t) = -\lambda M \hat{p}(q,t) + \lambda \sum_{m=1}^{M} \int_{[0,1]^N} e^{-2\pi i q \cdot \beta_m(x)} p(x,t) e^{-2\pi i q \cdot x} dx,$$

(4.9)
where $q \in \mathbb{R}^N$ is the Fourier index and $\beta_m$ is the $m^{th}$ column of an $N \times M$ matrix I call the interaction kernel. Its $(n,m)^{th}$ entry specifies the effect on node $n$ if an interaction occurs across edge $m$. For our model, this is

$$
\beta_{n,m}(x) = \begin{cases} 
\alpha \max(0, x_{n'} - x_n) & \text{if edge } m \text{ connects nodes } n \text{ and } n', \\
0 & \text{otherwise.}
\end{cases}
$$

In principle, (4.9) could work for other interaction laws; we need only specify the correct $\beta$.

However, this approach presents problems for both analysis and numerics. Analytically, the integral in (4.9) is challenging even when the regularity of the graph (as in the 4-regular graph) gives $\beta$ a (relatively) regular form. Numerically, we can nominally extract a spectral scheme from this setup. However, because $p(x,t)$ is not periodic in $x$, its Fourier coefficients will decay very slowly, and any partial sum will exhibit significant oscillation.

Let us instead follow the classical derivation of the Kolmogorov forward equation. However, instead of an Ito process we now have the Poisson process

$$
dk(t) = \beta(k^-(t))dY(t),
$$

where $\beta$ is the full matrix interaction kernel and $Y : \mathbb{R}^+ \to \mathbb{R}^M$ is the vector of i.i.d. Poisson processes governing the interactions. The following definition will be useful.

**Definition 4.4.** Let $G$ be a graph defining a social network, as in the foregoing. The graph of states $S(G)$ is a directed graph whose nodes are possible values of $k$. From node $x$, $S(G)$ has precisely $M$ directed edges. The $m^{th}$ edge connects $x$ to node $x + \beta_m(x)$, the node represented by the state $k$ would be in if it started at $x$ and then had an interaction on edge $m$.

Under this definition, nodes can be connected to each other multiply (if a low-
knowledge node of $G$ has multiple high-knowledge neighbors with the same knowledge) or even to themselves (if two neighbors in $G$ have the same knowledge). For $x, y \in S(G)$, let $e(y, x)$ be the set of edges from $y$ to $x$. Then for a small $h > 0$,

$$P(\{k(t + h) = x\}|\{k(t) = y\}) = \begin{cases} 1 - \lambda M h + \lambda |e(x, x)|h + O(h^2) & \text{if } x = y, \\ \lambda |e(y, x)|h + O(h^2) & \text{otherwise.} \end{cases}$$

Thus

$$p(x, t + h) = (1 - \lambda M h)p(x, t) + \sum_{m=1}^{M} \lambda h p(y_m(x), t) + O(h^2),$$

where $y_m(x)$ is the node of $S(G)$ one edge backward from $x$, obtained by an interaction on edge $m$. Thus

$$\frac{\partial p}{\partial t} (x, t) = -\lambda M p(x, t) + \lambda \sum_{m=1}^{M} p(y_m(x), t). \quad (4.10)$$

Equation (4.10) makes a great deal of intuitive sense: Over time, node $x$ receives probability mass from the $M$ nodes directly upstream from it and gives probability mass evenly to the $M$ nodes downstream from it.\footnote{When we enumerate the $M$ nodes upstream or downstream, we may be counting nodes multiply; remember, a node can have multiple connections to another node and even be connected to itself. In these cases some terms in the sum in (4.10) will be repeated, or they will be counteracted by the leading negative term, effectively reducing that term’s $M$ coefficient.}

This formulation of the Kolmogorov forward equation also presents problems. It makes clear that $p$ is supported on a measure-0 subset of $[0, 1]^N$. Analytical and numerical solutions will likely require enumerating the possible states (that is, the nodes of $S(G)$), whose combinatorics are nontrivial because of the multiple- and self-connection properties of $S(G)$. Enumerating the states of the even simplest non-line case, where $G$ is a triangle and $S(G)$ embedded in $[0, 1]^2$, is nontrivial.
4.5 Future work

Here we have sought primarily to introduce this framework and describe some dynamics that arise in the elementary model. However, we believe that the model can support some promising extensions. We discuss several below.

As noted in chapter 3, human interaction is often burstier than a Poisson process [1]. The Poisson process could be replaced with a self-exciting point process like a Hawkes process, so that each edge would have its own interaction rate $\lambda_m(t)$ depending on the pattern of past interactions. This would complicate analysis further, possibly making some analysis intractable, but could produce interesting dynamics that better reflect human activity patterns.

Different graph structures could also provide some rich behavior. Suppose the graph were directed, so that information could flow only one way on an edge. This is effectively the case in the 1D nearest-neighbor configuration we consider above. It may lead to more tractable analysis, and it may also be a way to incorporate social hierarchies into the model. Another variation on graph structure would be to consider dynamic graphs, in which links can form or be broken over time.

Finally, we could consider the inverse problem of guessing the graph based on the spread of knowledge. This is analogous to the missing data problem discussed in chapter 3.
REFERENCES


