FINE-GRAINED MODELING FOR CLINICAL DECISIONS VIA MACHINE LEARNING

A Dissertation
Presented to
The Academic Faculty

By

Jiaming Cui

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
College of Computing
School of Computational Science and Engineering

Georgia Institute of Technology

August 2024

FINE-GRAINED MODELING FOR CLINICAL DECISIONS VIA MACHINE LEARNING

Thesis committee:

Dr. B. Aditya Prakash, Advisor
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. Srijan Kumar
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. Anil Vullikanti
Department of Computer Science and Biocomplexity Institute
University of Virginia

Dr. Xiuwei Zhang
School of Computational Science and Engineering
Georgia Institute of Technology

Dr. Pinar Keskinocak
School of Industrial and Systems Engineering
Georgia Institute of Technology

Date approved: July 19, 2024

叹年来踪迹，何事苦淹留。

To Baoli Li and Xiao Cui

ACKNOWLEDGMENTS

First, I want to thank my Ph.D. advisor, Prof. B. Aditya Prakash, for his mentorship, guidance, and support in both completing my Ph.D. degree and pursuing an academic career. I deeply appreciate his passion, talent, and discernment in science and research, which provide life-long lessons I can always learn from.

I also want to extend my gratitude to my thesis committee members: Prof. Anil Vullikanti, Prof. Pinar Keskinocak, Prof. Xiuwei Zhang, and Prof. Srijan Kumar. Their constructive and valuable suggestions and feedback significantly strengthened my thesis. Specifically, I want to thank Prof. Anil Vullikanti, who closely collaborated with me on multiple projects and greatly helped me during my job search. Additionally, I am grateful to Prof. Bijaya Adhikari and Prof. Alexander Rodríguez for their substantial support throughout my Ph.D. studies and job search.

I am also grateful to my collaborators, labmates, and close friends: Prof. Eili Klein, Dr. Gregory Madden, Prof. Chao Zhang, Prof. Naren Ramakrishnan, Prof. Sriram Pemmaraju, Prof. Jilles Vreeken, Dr. Costi Sifri, Dr. Bryan Lewis, Dr. Lingkai Kong, Dr. Hankyu Jang, Dr. Methun Kamruzzaman, Dr. Anika Tabassum, Dr. Arash Haddadan, Jack Heavey, Harshavardhan Kamarthi, Zhiyuan Zhao, Haoxin Liu, Shangqing Xu, Jingfan Meng, Zihui Qian, Yifeng Cao, Huayi Wang, Yule Wang, Kaige Xie, Sungjun Cho, Jiajia Xie, Vivek Anand, Matthew Bielskas, Leo Lin, Javen Ho, and Pulak Agarwal. The collaborations and communications with them were invaluable, beneficial, and made for an unforgettable Ph.D. journey.

I also want to thank all the professors and staff in the School of Computational Science and Engineering at the Georgia Institute of Technology. This is indeed a great place for study, research, and growth. I also appreciate and miss my undergraduate life at Shanghai Jiao Tong University, China.

Finally, I want to thank my parents, Baoli Li and Xiao Cui, who raise me, teach me, and support me. I believe they are always with me and we are always together, wherever we are and whenever it is.

Standing at the end of my Ph.D. study and looking forward, I would like to end with the quote by Winston Churchill: “Now this is not the end. It is not even the beginning of the end. But it is, perhaps, the end of the beginning.”

Work in this dissertation was partially supported by the NSF (Expeditions CCF-1918770, CAREER IIS-2028586, RAPID IIS-2027862, Medium IIS-1955883, Medium IIS-2106961, PIPP CCF-2200269), CDC MInD program, and computing resources from Georgia Tech.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Public health challenges threaten people's lives and place a high burden on our healthcare system. For example, for infectious diseases, COVID-19 has led to 775 million cases and 7 million deaths worldwide as of July 2024, making it one of the largest public health crises in human history. Healthcare-associated infections (HAIs), such as Methicillin-resistant *Staphylococcus aureus* (MRSA) and *Clostridioides difficile* (*C. diff*), infect approximately 3% of hospitalized patients in the United States every year, resulting in more than 2.8 million cases and 35,000 deaths annually. These challenges force us to make critical clinical decisions in context of public health, such as releasing less severe patients to admit those in greater danger. In turn, such decisions may also lead to more infections in the community, and influence the hospital back. However, it is challenging to use existing epidemiological models to guide such decision-making. The spread pathways of infectious diseases are much more complicated in hospital settings when they interact with healthcare systems, and existing models cannot capture all these pathways effectively to make informed decisions. Additionally, these models also cannot digest the rich clinical datasets that provide a large amount of patient-level data, restricting them from making accurate, fine-grained decisions.

To tackle this, in this dissertation, we propose novel machine learning algorithms that can utilize these datasets to help design more detailed, fine-grained epidemiological models for more accurate infectious disease surveillance and control practices. Specifically, in Part I, we will show how such environmental factor-mediated models could better reconstruct the spread pathway in hospitals for infectious disease surveillance and achieve more cost-efficient contact precaution policies for clinical control. In Part II, we will propose new ML algorithms to better calibrate epidemiological models to learn more accurate model parameters. Moreover, we also designed new frameworks to integrate neural networks and epidemiological models simultaneously, which allows us to incorporate electronic health record (EHR) data to give patient-level predictions. Experimental results on real-world clinical datasets from large hospital systems demonstrate that our models and frameworks lead to more effective and efficient decision-making, thereby better bridging public health with clinical decisions.

CHAPTER 1
INTRODUCTION

1.1 Motivation

Public health and clinical decisions are intertwined: Public health challenges place a high burden on our medical system, forcing us to choose between maintaining quality and treating more people [1, 2, 3, 4, 5, 6]. Meanwhile, inappropriate clinical decisions also cause downstream effects on communities [7, 8, 9, 10, 11, 12]. One of the best examples is infectious diseases: During the COVID-19 pandemic, hospital burdens increased significantly, and policymakers had to increase bed capacity to save more lives within limited budgets [13, 14, 15]. Meanwhile, patients not admitted to hospitals could still spread the disease in the community, leading to further infections [16, 17, 18, 19, 20]. All these highlight the importance of making appropriate clinical decisions in the context of public health. However, it is challenging to use existing epidemiological models to guide such decision-making. On the one hand, the spread pathways of infectious diseases are more complicated in hospital settings, and existing models cannot capture these pathways effectively to make informed decisions. One example is colonized patients of healthcare-associated infections (HAIs). HAIs such as Methicillin-resistant *Staphylococcus aureus* (MRSA) and *Clostridioides difficile* (*C. diff*) are infections acquired by patients during treatment at healthcare facilities such as hospitals and long-term nursing homes [21, 22, 23]. Although many HAI-colonized patients may not show symptoms, they can still transmit pathogen loads to other patients and contaminate the environment in healthcare facilities [10, 24, 25, 26]. However, traditional epidemiological models struggle to accurately model how these colonized patients infect others via healthcare workers (e.g., doctors and nurses) and contaminated environments [27, 28, 29]. This prevents epidemiologists and clinicians from responding promptly and precisely to the spread of HAIs, making commonly used contact precautions less reliable. On the other hand, existing epidemiological models cannot digest the rich clinical datasets that provide a large amount of patient-level data, restricting them from making accurate patient-specific decisions. For example, electronic health records (EHRs) offer extensive patient-level information. However, current models cannot leverage these data, preventing them from providing accurate, patient-specific decisions.

This dissertation emphasizes using machine learning to better learn from clinical datasets to help design accurate, fine-grained modeling, which can better capture the spread dynamics in healthcare facilities and empower better decision-making. It proposes new frameworks to combine neural networks with epidemiological models to learn from EHR datasets and incorporate healthcare workers (HCWs) and locations into HAI modeling, which helps to optimize patient-specific contact tracing policies for surveillance and contact precaution policies for control. Experimental results on real-world clinical datasets from large hospital systems demonstrate that our models and frameworks can better connect the pandemic spread dynamics with clinical surveillance and control policies and lead to more effective and efficient decision-making, thereby better bridging public health with clinical decisions.

1.2 Thesis Overview

In this section, we present the thesis statement and its structure.

1.2.1 Thesis Statement

> Machine learning empowers fine-grained models to bridge public health with clinical decisions

More specifically, I proposed several frameworks to improve the epidemiological modeling to better guide clinical decision-making. In terms of better exploiting traditional datasets such as the time series of infections, I proposed new frameworks based on information theory or end-to-end learning to derive more accurate model parameters. Moreover, I designed new frameworks to integrate neural networks and epidemiological models simultaneously, allowing us to utilize the electronic health record data of each patient and provide fine-grained, patient-level modeling results. For the epidemiological models, I propose utilizing environmental factor-mediated models to better capture HAI spread dynamics. Here, the environmental factors refer to the pathogen load at locations (e.g., patient rooms, doctor's offices, nurse stations) contaminated by patients and healthcare workers. By incorporating these environmental factors into modeling, we can better describe the spread dynamics of HAIs as well as the surveillance and control practices in healthcare facilities, thereby improving real-world HAI clinical policy-making. Following the thesis statement, the thesis is organized into two parts: (Part I) Fine-grained modeling for clinical decisions and (Part II) Machine learning for accurate, fine-grained modeling. The outline is shown in Table 1.1.

1.2.2 Thesis Structure

In chapter 2, we will discuss the current modeling works and their limitations. Completed work of fine-grained modeling for clinical decisions (Part I) is discussed in chapter 3-5 in sub-section (a) reconstruct the spread pathway for infectious disease surveillance, and (b) more cost-efficient contact precaution policies for clinical control. Completed works of machine learning for accurate, fine-grained modeling (Part II) are discussed in chapter 6-9 and are divided into two sub-sections: (c) calibrate more accurate model parameters, and (d) incorporating electronic health records (EHRs) into patient-level models.

Table 1.1: Structure of the thesis with references to the chapters.

Part I: Fine-grained Modeling for Clinical Decisions
(a) Reconstruct the Spread Pathway for Infectious Disease Surveillance • H^2ABM: Heterogeneous Agent-based Model on Hypergraphs to Capture Group Interactions (chapter 3)
(b) More Cost-efficient Contact Precaution Policies for Clinical Control • Using Spectral Characterization to Identify Healthcare-associated Infection (HAI) Patients for Clinical Contact Precaution (chapter 4) • Modeling Relaxed Policies for Discontinuation of Methicillin Resistant *Staphylococcus aureus* Contact Precautions. (chapter 5)
Part II: Machine Learning for Accurate, Fine-grained Modeling
(c) Calibrate More Accurate Model Parameters • Accurately Estimating Total COVID-19 Infections using Information Theory (chapter 6) • End-to-End Stochastic Optimization with Energy-Based Model (chapter 7)
(d) Incorporating Electronic Health Records (EHRs) into Patient-level Models • Identifying Importation and Asymptomatic Spreaders of Multi-drug Resistant Organisms in Hospital Settings (chapter 8) • Combining Agent-based Models and Neural Networks to Forecast Future Cases of Healthcare Associated Infections (chapter 9)

1.3 Contributions and Impact

1.3.1 Contributions

My work focuses on realizing appropriate clinical decisions within the context of infectious diseases for public health. To achieve this, I developed more comprehensive models to better capture the spread dynamics in healthcare facilities and make better decisions. For example, I have incorporated healthcare workers (HCWs) and locations into healthcare-associated infection (HAI) modeling to optimize contact tracing for surveillance and contact precaution policies for control. I also propose using machine learning to learn from different clinical datasets to improve modeling. For instance, I combine neural networks with epidemiological models to learn from EHR datasets to estimate each patient's probability of being infected.

1.3.2 Impact

My research solved several clinical decision-making problems raised by clinicians at the University of Virginia hospital. They greatly appreciated these insights and were eager to implement them in practical, real-world scenarios. I also actively participated in the CDC MInD (Modeling Infectious Diseases in Healthcare Network) group and presented my work in the weekly meeting several times [30, 31, 32]. Our machine learning frameworks for estimating patient-specific infection risks interested clinicians and CDC officials in the MInD meeting, and they asked nearly 10 questions after our presentation. We also received many emails asking for more details and potential extensions soon after the meeting. I am also a student member of the MIDAS network. I presented my work at its 2022 and 2023 meetings [33, 31]. Many epidemiologists showed interest in my work and discussed potential extensions with me during and after the meetings. My works have also been incorporated into graduate courses (CSE 8803 Data Science for Epidemiology in Georgia Tech). I also won the best poster award in SDM 2024 conference.

1.4 Summary of Work

1.4.1 Part I: Fine-grained Modeling for Clinical Decisions

(a) Reconstruct the Spread Pathway for Infectious Disease Surveillance

Chapter 3: H^{2}ABM: Heterogeneous Agent-based Model on Hypergraphs to Capture Group Interactions [34] This chapter proposes using hypergraphs to better capture group interactions and extend the current graph-based HABMs to hypergraphs. Specifically, we use MRSA spread in the University of Virginia hospital as an example to showcase how we extend an existing graph-based HABM, GRAPH-HETERSIS, to a hypergraph-based HABM (H^{2}ABM), HYPERGRAPH-HETERSIS. We demonstrate how hypergraphs can better capture the structural differences between contacts before and during the first wave of the COVID-19 outbreak in Virginia compared to graphs. Our experiments show that H^{2}ABM better captures the underlying group interactions and more accurately fits and forecasts MRSA cases.

(b) More Cost-efficient Contact Precaution Policies for Clinical Control

Chapter 4: Using Spectral Characterization to Identify Healthcare-associated Infection (HAI) Patients for Clinical Contact Precaution [30] This chapter focuses on two clinical problems arising from a limited budget: (a) choosing the best patients to be placed under precaution to minimize the spread (the Isolation problem), and (b) choosing the best patients to release when the limited budget requires some patients to be cleared from precaution (the Clearance problem). One of the critical challenges in solving these problems is that HAIs have multiple transmission pathways, allowing locations to accumulate "load" and spread the disease. However, standard propagation models like Independent Cascade (IC) and Susceptible-Infected-Susceptible (SIS) have difficulty capturing such mechanisms. To address this challenge, we use non-linear system theory to develop a novel spectral characterization of a recently proposed 2-MODE-SIS model on people/location networks to capture HAI spread dynamics. We formulate the two clinical problems with this spectral characterization and explain previously observed counter-intuitive behaviors like the "dilution effect". We also develop effective and efficient algorithms to solve the formulated problems.

Chapter 5: Modeling Relaxed Policies for Discontinuation of Methicillin Resistant *Staphylococcus aureus* Contact Precautions [31] Currently, the policy at the University of Virginia (UVA) hospital is to maintain contact precautions for patients with MRSA until they present three consecutive negative tests (i.e., the "3-negative policy"). One clinical problem raised by clinicians is whether relaxing the present "3-negative policy" to two or one negative test would significantly impact the epidemiology in the hospital. Hence, this chapter proposes an agent-based model to capture MRSA spread in the UVA hospital, which includes environmental contamination and contacts between patients and providers. We also propose a parameter adjustment method using the calibrated parameters for a 3-negative policy scenario to predict the outcomes for the 2- and 1-negative policy scenarios. Our results suggest that both the 2- and 1-negative policies lead to an increase in MRSA cases and treatment costs. However, they also reduce the average time patients spend under precautions, which lowers costs. Even accounting for the increased cost of therapy, the total cost (sum of both treatment and contact precaution costs) is lowest for the 1-negative policy.

1.4.2 Part II: Machine Learning for Accurate, Fine-grained Modeling

(c) Calibrate More Accurate Model Parameters

Chapter 6: Accurately Estimating Total COVID-19 Infections using Information Theory [33] This chapter proposes an information-theoretic approach for accurately estimating the number of total infections. The approach is built on top of ordinary differential equations (ODE) based models, which are commonly used in epidemiology. We demonstrate how these models can be used to estimate the number of total infections and identify the parameterization requiring the fewest bits to describe the observed dynamics of reported infections. The experiments show that this approach not only provides better estimates of the number of total infections but also offers improved forecasts compared to standard model calibration methods. Additionally, we demonstrate how this approach can be used to simulate what-if scenarios with non-pharmaceutical interventions.

Chapter 7: End-to-End Stochastic Optimization with Energy-Based Model [35] Decision-focused learning (DFL) was recently proposed for stochastic optimization problems involving unknown parameters. By integrating predictive modeling with an implicitly differentiable optimization layer, DFL has shown superior performance compared to the standard two-stage predict-then-optimize pipeline. However, most existing DFL methods are only applicable to convex problems or a subset of non-convex problems that can be easily relaxed to convex ones. Therefore in this chapter, we propose SO-EBM, a general and efficient DFL method for stochastic optimization using energy-based models. Instead of relying on KKT conditions to induce an implicit optimization layer, SO-EBM explicitly parameterizes the original optimization problem using a differentiable optimization layer based on energy functions. We evaluate SO-EBM in three applications, demonstrating its effectiveness and efficiency.

(d) Deciding HAIs Releasing Policies

Chapter 8: Identifying Importation and Asymptomatic Spreaders of Multi-drug Resistant Organisms in Hospital Settings Patients are sometimes already infected with HAI at the time of admission to the hospital (referred to as "importation"), and additional patients might get infected in the hospital through transmission ("nosocomial infection"). Since testing is limited and incurs significant delays, a major challenge for hospital infection control is to rapidly identify importation and nosocomial infection cases. In this chapter, we propose NEURABM, a new framework that integrates both neural networks and agent-based models (ABM) to combine the advantages of both modeling-based and machine learning-based methods. NEURABM simultaneously learns a neural network model for patient-level prediction of importation and uses the ABM model for identifying infections. Our results demonstrate that NEURABM identifies importation and nosocomial infection cases more accurately than existing methods.

Chapter 9: Combining Agent-based Models and Neural Networks to Forecast Future Cases of Healthcare Associated Infections Agent-based models (ABMs) are commonly used in infectious disease forecasting due to their ability to capture agent interactions in contact networks. However, these models often assume static networks, limiting their accuracy for dynamic environments like hospitals. Machine learning methods, while not reliant on contact networks, often lack the epidemiological insights of ABMs. The proposed FORECASTABM framework integrates ABMs and neural networks to overcome these limitations. By using an adapter neural network, it forecasts future healthcare-associated infection cases based on current patient states without needing future contact networks, ensuring higher accuracy and practical applicability in real-world scenarios.

1.4.3 Publication List

The following are the author's finished works which are discussed in this thesis.

1. Vivek Anand*, Jiaming Cui*, Jack Heavey, Anil Vullikanti, B. Aditya Prakash. H^2ABM: Heterogeneous Agent-based Model on Hypergraphs to Capture Group Interactions. SDM 2024 (Houston). **[Best poster award]**

2. Jiaming Cui*, Sungjun Cho*, Methun Kamruzzaman, Matthew Bielskas, Anil Vullikanti, B. Aditya Prakash. Using Spectral Characterization to Identify Healthcare-associated Infection (HAI) Patients for Clinical Contact Precaution. Scientific Reports 2023.

3. Jiaming Cui, Jack Heavey, Leo Lin, Eili Y. Klein, Gregory R. Madden, Costi D. Sifri, Bryan Lewis, Anil K. Vullikanti, B. Aditya Prakash. Modeling Relaxed Policies for Discontinuation of Methicillin Resistant Staphylococcus aureus Contact Precautions. ICHE 2024.

4. Jiaming Cui, Jack Heavey, Leo Lin, Eili Y. Klein, Gregory R. Madden, Costi D. Sifri, Bryan Lewis, Anil K. Vullikanti, B. Aditya Prakash. Modeling Relaxed Policies for Discontinuation of Methicillin Resistant Staphylococcus aureus Contact Precautions. (poster). MIDAS 2023 (Atlanta).

5. Jiaming Cui, Arash Haddadan, A S M Ahsan-Ul Haque, Jilles Vreeken, Bijaya Adhikari, Anil Vullikanti, B. Aditya Prakash. Accurately Estimating Total COVID-19 Infections using Information Theory. (SDM 2025, in preparation)

6. Jiaming Cui, Arash Haddadan, A S M Ahsan Ul Haque, Bijaya Adhikari, Anil Vullikanti and B. Aditya Prakash. Information Theoretic Model Selection for Accurately Estimating Unreported COVID-19 Infections (poster). MIDAS 2022 (Washington DC).

7. Lingkai Kong, Jiaming Cui, Yuchen Zhuang, Rui Feng, B. Aditya Prakash, Chao Zhang. End-to-end Stochastic Programming with Energy-based Model. NeurIPS 2022 (New Orleans).

8. Jiaming Cui, Jack Heavey, Eili Y. Klein, Gregory R. Madden, Anil K. Vullikanti, B. Aditya Prakash. Identifying Importation and Asymptomatic Spreaders of Multi-drug Resistant Organisms in Hospital Settings. (Nature Communications, under review)

1.4.4 Excluded Work

For better coherence, some of the author's Ph.D. work have been excluded from this thesis. The excluded work includes the following.

1. Haoxin Liu, Shangqing Xu, Zhiyuan Zhao, Lingkai Kong, Harshavardhan Kamarthi, Aditya B. Sasanur, Megha Sharma, Jiaming Cui, Qingsong Wen, Chao Zhang, B. Aditya Prakash. Time-MMD: A New Multi-Domain Multimodal Dataset for Time Series Analysis. (NeurIPS 2024, under review)

2. Lingkai Kong, Wenhao Mu, Jiaming Cui, Yuchen Zhuang, B. Aditya Prakash, Bo Dai, Chao Zhang. DF^2: Distribution-Free Decision-Focused Learning. (NeurIPS 2024, under review)

3. Lingkai Kong, Jiaming Cui, Haotian Sun, Yuchen Zhuang, B. Aditya Prakash, Chao Zhang. Autoregressive Diffusion Model for Graph Generation. ICML 2023 (Hawaii).

4. Hankyu Jang, Andrew Fu, Jiaming Cui, Methun Kamruzzaman, B. Aditya Prakash, Anil Vullikanti, Bijaya Adhikari, Sriram Pemmaraju. Detecting Sources of Healthcare Associated Infections. AAAI 2023 (Washington DC).

5. Alexander Rodríguez, Jiaming Cui, Naren Ramakrishnan, Bijaya Adhikari and B. Aditya Prakash. EINNs: Epidemiologically-Informed Neural Networks. AAAI 2023 (Washington DC).

6. Jack Heavey, Jiaming Cui, Chen Chen, B. Aditya Prakash. Provable Sensor Sets for Epidemic Detection over Networks with Minimum Delay. AAAI 2022 (Vancouver BC).

7. Cramer, Estee Y., et al. [multiple co-authors, including Jiaming Cui] Evaluation of individual and ensemble probabilistic forecasts of COVID-19 mortality in the United States. PNAS 2022.

8. Cramer, Estee Y., et al. [multiple co-authors, including Jiaming Cui] The united states covid-19 forecast hub dataset. Scientific data 2022.

9. Alexander Rodríguez, Anika Tabassum, Jiaming Cui, Jiajia Xie, Javen Ho, Pulak Agarwal, Bijaya Adhikari and B. Aditya Prakash. DeepCOVID: An Operational Deep Learning-driven Framework for Explainable Real-time COVID-19 Forecasting. IAAI 2021 (virtual).

1.5 Thesis Outline

The outline of the thesis is as follows. First, we discuss the current modeling works in chapter 2. Next, we present Part I: Fine-grained modeling for clinical decisions in chapters 3 to 5. Then, we present Part II: Machine learning for accurate, fine-grained modeling in chapters 6 to 9. Chapter 10 will discuss the conclusion and future works, followed by the appendix.

CHAPTER 2
SURVEY

This chapter surveys the existing literature on healthcare-associated infections, explaining why they are important, what people do for HAI surveillance and control, and why they are challenging. Next, we introduce the current methods to capture the HAI spread and guide decision-making. We first focus on the mechanistic models, such as ordinary differential equation models and agent-based models. We then introduce graph and hypergraph analysis methods, such as information diffusion and optimization problems. Next, we list the statistical and machine learning models, covering regression models, network-based models, CV/NLP models, and density estimation models. Finally, we highlight the recent hybrid models that attempt to integrate both mechanistic modeling and machine learning together.

2.1 Healthcare-associated Infections

Healthcare-associated infections such as Methicillin-resistant *Staphylococcus aureus* (MRSA) and *Clostridioides difficile* (*C. diff*) are infections acquired by patients during treatment at healthcare facilities like hospitals and long-term nursing homes [21, 22, 23]. They impose a tremendous burden on hospitals since they may lead to severe outcomes, longer hospital stays for patients, and increase the risk of mortality [36, 37, 38, 3, 39, 40]. Because of its severe outcome, epidemiologists and clinicians have proposed multiple approaches for HAI surveillance and control.

One of the most significant challenges in HAI surveillance is identifying asymptomatic infections. These infections present no symptoms but are still able to infect other healthy patients. For example, many patients have already been infected but present no symptoms at admission (i.e., importation cases). The European Centre for Disease Prevention and

Control (ECDC) estimates that importation cases contribute to 13% of HAI cases in Germany and 18.9% in Spain [41]. These importation cases can spread HAI-causing pathogens and lead to nosocomial infection cases, which can also be asymptomatic but further spread pathogens to additional healthy patients [42]. Surveillance tests [43, 44] such as culture or PCR tests are common methods in hospitals to identify importation and nosocomial infection cases. However, they are costly, require time to process, and are not 100% accurate [45]. Additionally, these are typically only able to be used for a subset of multidrug-resistant organisms (MDROs), and therefore not applicable for all kinds of HAIs.

Regarding to the control, epidemiologists and clinicians have also proposed several approaches to control HAI spread in hospitals, such as contact precautions, antimicrobial stewardship, and contact tracing [7, 8, 9, 10, 11, 12]. Among these approaches, contact precautions have been shown to be more effective in preventing further nosocomial infections [46, 47], and easier to implement [48]. Therefore, they are most commonly used in most hospitals to control HAI spread [9]. However, real-world contact precautions are expensive [1, 2, 3, 4, 5, 6] and therefore they can only be applied to a small fraction of patients (a "budget"). Further, there are several downsides to contact precautions on patient health outcomes, e.g., [6, 5]. For example, if MRSA testing are always accurate, a single-negative test would be optimal to release patient out of contact precaution. However, several factors, including sensitivity of the test, low bacterial load at time of collection, and sampling error, abrogate the accuracy of testing. Repeated testing improves the sensitivity and thus can extend precautions in patients that may otherwise be assumed clear. Moreover, isolating MRSA patients is expensive since it requires gowns and gloves and other resources, with an average estimated cost of more than $400 per day per patient [49]. In addition, although the extent of side-effects of contact precautions remains uncertain [5, 50], there are potential harms due to extended contact precautions [51, 37], including reductions in provider visits to patient rooms [4], which can increase risk of depression. Thus, releasing policies must balance the potential harms from contact precautions against the potential prevention of transmission. To-date there are few systematic analyses of the optimal length of releasing policies. Prior work on evaluating releasing policies [52, 53] have focused on individual-level risk estimation using EHR data. However, there is no evaluation at the hospital level yet.

2.2 Mechanistic Models

Ordinary differential equation-based models are widely used in epidemiology to identify the asymptomatic infections and forecast future number of cases. Data scientists and epidemiologists have devoted much time and effort to estimate the reported rate through ODE models [20, 58, 59, 60, 61, 62, 63, 64, 65, 54, 55, 56, 57]. In general, the model has a set of parameters that we estimate from *observed data* using a so-called calibration procedure. In practice, the data we use for calibration can be the time series of the number of reported infections. To estimate the number of total infections, these models often explicitly include reported rate as one of their parameters, or include multiple parameters that jointly account for it. There are many calibration procedures commonly used in literature, such as RMSE-based [66] or Bayesian approaches [65, 64]. Moreover, by assuming different parameters for different level of non-pharmaceutical interventions, people could also use ODE models to evaluate how different control measures can reduce the number of cases.

In additional to ODE models, agent-based models are also widely used in clinical practice. Compared with ODE models, ABMs could use information about contact networks between patients and providers within healthcare facilities to model the infection status of an individual patient. For example, many models have been developed [67, 10, 68]) to capture the transmission dynamics of HAIs within a healthcare facility. They are calibrated to infections in the hospital and use projections from such models for prediction. Moreover, ABMs built on heterogeneous contact networks have been widely used in infectious disease modeling [68]. One of the most important applications among them is for HAI modeling [67, 68, 69]. Since both people (e.g., patients or HCWs) and contaminated surfaces can spread pathogens [24, 26] and their dynamics can be significantly different (e.g., patients or HCWs can move in hospital, while locations cannot move), heterogeneous ABMs are more suitable than conventional homogeneous ABMs. However, they still cannot directly incorporate the risk factor of each patient from the EHR data, such as medications, lab results, vital signs, and device use history into modeling.

2.3 Graph and Hypergraph Analysis

Starting from the contact networks, people also study the spread dynamics from the information diffusion prospective. For example, propagation models have been investigated in the context of information spreading on graphs, with applications include viral marketing [70], spread of topics across blogs [71], and network security [72]. Moreover, in terms of the optimization problems, a particularly well-studied problem is influence maximization, an optimization problem of which the goal is to find a set of nodes that can initiate the largest spread of information throughout the network [73, 74, 75]. In additional to the information diffusion, epidemic thresholds are also stuided on the networks. For example, [76, 77] studied thresholds for multiple competing pathogens on random graphs by mapping the SIR model to a percolation problem, and [78, 79] gave the threshold for the SIS model on any arbitrary undirected networks.

Recently, hypergraphs have been used for many applications, such as rumor detection [80], business modeling [81], recommendation system design [82], and financial transactions [83]. Hypergraphs have also been used for infectious disease modeling, most notably for COVID-19 [84]. However, in their work, they assume that every vertex is of the same type (i.e., the model is homogeneous), while the entities involved in HAI spread could be heterogeneous (e.g., patients, healthcare workers, and locations).

2.4 Statistical and Machine Learning Models

Recently, machine learning and statistical techniques are also explored in epidemiology [85, 86]. One main category of these models is regression models, where they use patient-specific information (e.g., age, occupation) and risk factors (e.g., previous surgery history, chronic diseases) as input, and predict patients' infection probabilities. For example, there are multiple types of regression models, including linear regression models, auto-regressive models, traditional ML models, and hierarchical models [87]. Examples of linear regression models include [88] and [89], which use lasso and ridge regularizations. Examples of auto-regressive models include autoregressive integrated moving average (ARIMA) and its variations like SARIMA, ARIMAX, and SARIMAX [90]. Traditional ML models are support vector machine (SVM), decision tree, and Naive Bayes [91]. Example hierarchical models are ARGO2 and ARGOX [92, 93].

In addition to the regression models, neural network-based models are also widely used. For example, sequential models like long short-term memory (LSTM)[94], gated recurrent units (GRU) [95], and transformers [96] are widely used for forecasting. Additionally, temporal dynamics are also used to find the similarity for forecasting [97]. Other attempts also explore transfer learning techniques [98], leverage multi-modal data [99], and incorporate spatial structure [100]. Recently, computer vision (CV) [101] and natural language processing (NLP) [102] models built on deep neural networks have also been widely explored.

Another category of machine learning models is density estimation models. For example, [103] uses parametric Bayesian inference, and [104] uses kernel density estimation. Besides, uncertainty quantification is also being explored recently [105]. In addition, decision-focused learning based on end-to-end models is also widely used. For example, [106] relaxes the discrete decision into its continuous counterpart and adds a quadratic regularization term to avoid vanishing gradients in DFL. [107] proposes a log barrier regularizer and differentiates through the homogeneous self-dual embedding. [108] proposes a noise contrastive objective by maximizing the distance between the optimal solution and noisy samples.

However, the performance of machine learning methods has not proven to be high enough due to many reasons, including imbalance (since infections usually only account for a very small fraction of the entire patient population), bias in the data (since testing is generally not done in a systematic manner), and the fact that machine learning methods do not incorporate epidemiological knowledge in their frameworks.

2.5 Hybrid Models

Recently, there have also been works aiming to integrate the advantages of both mechanistic models and machine learning models. For example, to learn the mechanistic parameters (e.g., transmission rate, recovery rate), people explore statistical techniques [109], learning from simulation-generated data [110], and using ODE regularizations [111]. People also explore integrating search volume data into ODE models via data assimilation [112]. However, such data assimilation is non-trivial and requires a lot of domain knowledge. Recently, physics-informed neural networks (PINNs) have been proposed to approximate the output of ODEs with universal functions, which use unsupervised loss functions designed with ODE equations as regularizers [113]. They have been widely used in epidemic forecasting and control and show good performance in various tasks [114].

In addition to ODEs, people have also explored how to integrate deep neural networks with ABMs. For example, [115, 116, 117] explores using multi-agent reinforcement learning (MARL) methods, which considers ABM as a reinforcement learning environment and uses deep neural networks to learn different policies. Other works also explore using ABM to generate synthetic datasets to train neural networks for contact tracing [118] or calibration [119]. However, these methods all suffer from the non-differentiability of the ABM and cannot perform automatic differentiation. Recently, another framework named GRADABM [120] proposes to train deep neural networks and ABMs jointly, integrating heterogeneous data to infer latent micro variables. By developing a differentiable agent-based model, they enable gradient-based learning with automatic differentiation to calibrate simulation parameters and show good performance.

Part I

Fine-grained Modeling for Clinical Decisions

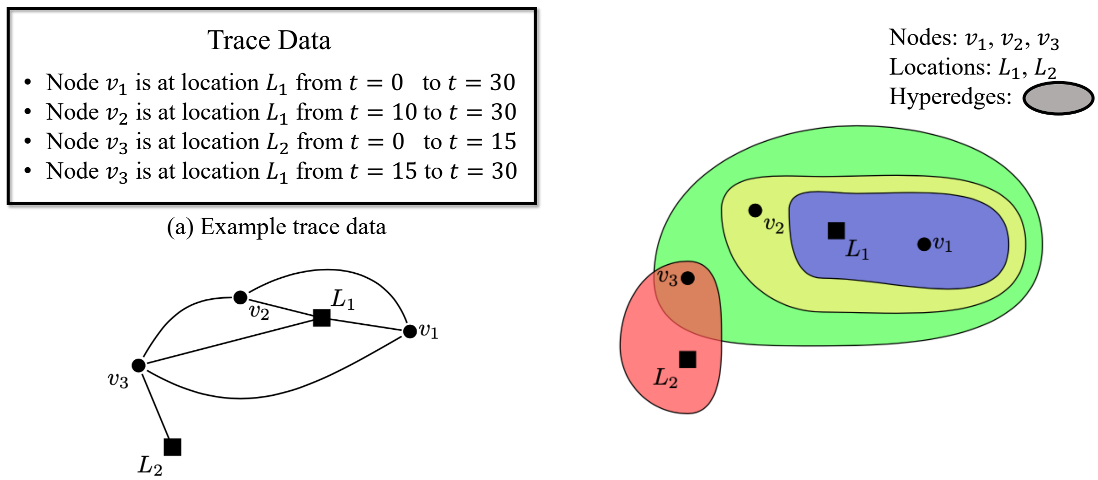
CHAPTER 3

H^{2}ABM: HETEROGENEOUS AGENT-BASED MODEL ON HYPERGRAPHS TO CAPTURE GROUP INTERACTIONS

Heterogeneous agent-based models (HABMs) can simulate the dynamics of multiple types of entities and their interactions on contact networks. In recent years, they have gathered great interest and are widely applied in multiple fields, such as personalized recommendations, publication ranking, and epidemic modeling. Nevertheless, conventional HABMs on graphs can only capture pair-wise interactions between agents but fail to capture the more complex dynamics of group interactions (e.g., multiple people in the same location simultaneously), consequently leading to suboptimal performance. To address this, we propose using hypergraphs to capture such group interactions better and extend the current graph-based HABMs to hypergraphs. Specifically, we use MRSA (Methicillin-resistant *Staphylococcus aureus*, a kind of infectious disease acquired by patients during treatment at healthcare facilities) spread in the University of Virginia hospital as an example to showcase how we extend an existing graph-based HABM, GRAPH-HETERSIS, to a hypergraph-based HABM (H^{2}ABM), HYPERGRAPH-HETERSIS. We show how the hypergraphs can capture the structural difference between contacts before and during the first wave of COVID-19 outbreak in Virginia better than graphs. Our experiments show that H^{2}ABM better captures the underlying group interactions and better fits and forecasts MRSA cases.

3.1 Introduction

A **<u>H</u>**eterogeneous **<u>A</u>**gent-**<u>b</u>**ased **<u>M</u>**odel (HABM) is a computational model used to simulate the actions and interactions of multiple types of entities [121, 122, 69, 30, 31]. They are "heterogeneous" since they explicitly account for differences in characteristics, behaviors, or decision-making rules among different kinds of entities. For example, a HABM to model



(a) Example trace data

(b) Graph generated from example trace data

(c) Hypergraph generated from example trace data

Figure 3.1: **Example of graph and hypergraph generated from trace data to capture the interactions of vertices and locations.** (a) An example trace data of vertices. It traces the location (L_1 and L_2) of each vertex (v_1 to v_3) for every timestep t. (b) The graph generated from example trace data in (a). Dots and squares represent vertices and locations, respectively. An edge between two entities represents the pair-wise relation between them. For example, only v_1 and v_2 are at the same location L_1 from $t = 10$ to $t = 15$. Therefore, there is an edge between v_1 and v_2. (c) The hypergraph generated from example trace data in (a). Each colored circle corresponds to a hyperedge, which represents a group interaction between the vertices and locations inside it. For example, only v_1 and v_2 are at the same location L_1 from $t = 10$ to $t = 15$, therefore there is a hyperedge (yellow one) among v_1, v_2, and L_1.

Healthcare-associated infections (HAI, infections acquired by patients during treatment at healthcare facilities such as hospitals and nursing homes [21, 22, 23]) may model the actions of patients, healthcare workers, locations as well as their interactions [68]. In recent years, HABMs have gathered great interest with applications in personalized recommendation [123], publication ranking [124], and drug design [125].

Current HABMs usually use graphs to capture the interactions between entities. However, graphs can only represent pair-wise interactions, while in the real world, many interactions are in groups. For example, as shown in Figure 3.1(a), only vertices v_1 and v_2 are at location L_1 from $t = 10$ to $t = 15$. However in Figure 3.1(b), the graph can only break the co-locations between v_1, v_2, and L_1 as 3 pair-wise interactions (edges between v_1 and v_2, v_1 and L_1, v_2 and L_1). Additionally, one cannot determine if v_2 interacted with both v_1

and v_3 together or with each of them separately from the graph. This makes the graphs less preferable in capturing complex dynamics of group interactions.

In this work, we propose to use hypergraphs to better capture such group interactions. As shown in Figure 3.1(c), the interactions between v_1, v_2, and L_1 from $t = 10$ to $t = 15$ can be easily represented by the yellow hyperedge. Similarly, the interactions between v_1, v_2, v_3, and L_1 from $t = 15$ to $t = 30$ can also be represented by the green hyperedge. From these two hyperedges, one can easily determine that v_2 interacted with v_1 and v_3 together once and interacted with only v_1 once. This shows that hypergraphs can capture the more complex group interactions more effectively. We also provide a more detailed explanation for this example in subsection 3.2.2.

However, switching from graphs to hypergraphs also requires us to extend the HABMs to hypergraphs. This is non-trivial since the dynamics of group interactions cannot simply be considered as the linear combination of each pair-wise interaction. In this work, we use MRSA (Methicillin-resistant *Staphylococcus aureus*, a kind of HAI) spread at the University of Virginia hospital and a widely-used graph-based HABM for HAI modeling, GRAPH-HETERSIS model [68], as an example to showcase how we can easily extend from a graph-based HABM to a **H**ypergraph-based **HABM** (H^{2}ABM). We refer to this H^{2}ABM as HYPERGRAPH-HETERSIS model in later sections.

Our main contributions are summarized below.

1. We use the widely-used GRAPH-HETERSIS model as a simple example to show how we can extend from a graph-based HABM to a hypergraph-based HABM (H^{2}ABM) and propose the HYPERGRAPH-HETERSIS model which incorporates group level nonlinear load discounting.
2. We show that hypergraphs can capture the structural difference better than graphs can. Specifically, we analyze the contact networks in the University of Virginia (UVA) hospital before and during the first wave of COVID-19 outbreak in Virginia and show that hypergraphs are more effective in capturing structural differences be-

tween the two contact networks than graphs.

3. We also show that the HYPERGRAPH-HETERSIS leads to better fitting and forecasting of MRSA cases in the UVA hospital than GRAPH-HETERSIS. It also provides good interpretability of the MRSA outbreak in UVA hospital.

3.2 Methods

3.2.1 GRAPH-HETERSIS Model

First, we will introduce a widely-used graph-based HABM, for HAIs, GRAPH-HETERSIS.

GRAPH-HETERSIS is a heterogeneous agent-based load-sharing SIS model over a graph [68]. It has been widely used to model HAIs such as MRSA in hospital settings. There are three different types of entities, each with different behaviors: Patients are mobile, can accumulate, transmit and get infected by pathogen; Healthcare Workers (HCWs) are mobile, can accumulate and transmit pathogen but cannot get infected; Locations cannot move but can accumulate and share pathogen.

GRAPH-HETERSIS is described in Algorithm 1. A vertex i can be either *susceptible* ($\mathbf{x}_t(i) = 0$) or *infected* ($\mathbf{x}_t(i) = 1$). In addition, the model keeps track of pathogen loads on all people and locations using a load vector $\boldsymbol{l}_t$ at each timestep t. Using the adjacency matrix $\mathbf{A}_t$ ($\mathbf{A}_t(i, j) = 1$ vertex if i and j interacted at time t and $\mathbf{A}_t(i, j) = 0$ otherwise) and the load transfer ratios τ_{ijt} (τ_{ijt} represents the ratio of loads transferred from vertex j to vertex i at time t), we can compute the pathogen transfer matrix $\boldsymbol{R}_t$ and write the pathogen load updates as a linear operation as seen in line 4 in Algorithm 1 (note that the column-sums of $\boldsymbol{R}_t$ are restricted to be less than or equal to 1, which implies that the total amount of pathogen cannot increase after transfer (i.e., $\|\boldsymbol{R}_t\boldsymbol{l}_t\|_1 \leq \|\boldsymbol{l}_t\|_1$)). The probability that a patient i gets infected from susceptible is a dose-response function proportional to $\boldsymbol{l}_t(i)$, as shown in line 7 in Algorithm 1 with β as the disease infectivity. Once infected, the patient i sheds α additional pathogen per timestep to his own load, which can later be transferred to neighbors (both people and locations) via edges; this shedding continues until the patient

Table 3.1: List of notations

Notation	Description
α	Pathogen shedding rate
β	Disease infectivity
δ	Recovery probability
τ_{ijt}	Pathogen transfer ratio from node j to node i at time t (GRAPH-HETERSIS model only)
Υ_{ijt}	Pathogen transfer ratio from node j to node i at time t (HYPERGRAPH-HETERSIS model only)
$\boldsymbol{l}_t$	Pathogen load vector at time t sized $N \times 1$
$\boldsymbol{x}_t$	Infection state vector at time t sized $P \times 1$
$\boldsymbol{A}_t$	Adjacency matrix at time t sized $N \times N$
$\boldsymbol{R}_t$	Pathogen transfer matrix at time t sized $N \times N$
$\boldsymbol{H}_t$	Hypergraph matrix at time t sized $M_t \times N$ (HYPERGRAPH-HETERSIS model only)
M_t	Number of hyperedges
$g(\cdot)$	Nonlinear discounting load accumulation function (HYPERGRAPH-HETERSIS model only)
P	Total number of patients
H	Total number of healthcare workers (HCWs)
L	Total number of locations
N	Total number of entities ($P + H + L$)

recovers with recovery probability δ. We list the notations in Table 3.1.

3.2.2 HYPERGRAPH-HETERSIS Model

In this section, we motivate the need for hypergraphs in modeling HAIs and subsequently introduce HYPERGRAPH-HETERSIS. We then showcase the additional expressivity of HYPERGRAPH-HETERSIS compared to GRAPH-HETERSIS due to the presence of non-linearity.

Hypergraphs

Hypergraphs are generalizations of graphs, where each hyperedge can include any number of vertices [126]. Compared with the edges of a graph that can only connect exactly two vertices, hyperedges are more flexible in capturing complex, multiple-entity relations in

Algorithm 1 GRAPH-HETERSIS model procedure

Inputs: $\alpha, \beta, \delta, \{\boldsymbol{A}_t\}_{t=1}^T, \{\tau_{ijt}\}_{i,j,t}$
Compute $\boldsymbol{R}_t(i,j) = \begin{cases} \tau_{ijt}\boldsymbol{A}_t(i,j) & \text{if } i \neq j \\ \tau_{ijt} & \text{if } i = j \end{cases}$
for $t = 1, \ldots, T$ **do**
    Update loads $\boldsymbol{l}_{t+1} = \boldsymbol{R}_t\boldsymbol{l}_t + \alpha\boldsymbol{x}_t$
    for each patient i **do**
        if i is susceptible at time t (i.e., $\boldsymbol{x}_t(i) = 0$) **then**
            i gets infected (i.e. $\boldsymbol{x}_{t+1}(i) = 1$ with probability min$\{1, \beta\boldsymbol{l}_t(i)\}$
        else
            i gets susceptible (i.e. $\boldsymbol{x}_{t+1}(i) = 0$ with prob. δ
        end if
    end for
end for

addition to the pairwise relations.

Hypergraph Formulation

Given a set of N vertices $V = \{v_1, v_2, \ldots v_N\}$ that interact with each other ($\|V\| = N$) at each timestep t, we use a incidence matrix $\boldsymbol{H}_t$ to represent the interactions between vertices at time t (i.e., for T timesteps, we have $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, $\cdots$, $\boldsymbol{H}_T$, and we use the same vertex set V for all $\boldsymbol{H}_t$). The dimension of hypergraph matrix $\boldsymbol{H}_t$ is $M_t \times N$, where M_t is the number of hyperedges at time t. Each row in $\boldsymbol{H}_t$ is a separate hyperedge h, and nonzero entries indicate membership of the respective vertices. For example, for the hypergraph shown in Figure 3.1(c), the yellow hyperedge between v_1, v_2 and L_1 can be written as $[1, 1, 0, 1, 0]$, where the five elements correspond to v_1, v_2, v_3, L_1, and L_2 respectively.

Limitations of Graph-based HABMs

Although graph-based HABMs can capture the pair-wise interactions between entities well, they are less suitable to model complex group interactions. This is because they can only break group interactions into multiple pair-wise interactions (e.g., breaking an interaction among three people to 3 pair-wise edges between each of two), which is computationally expensive (since it needs n^2 pair-wise edges to capture a group interaction between n

people). Meanwhile, graph-based HABMs are also less suitable for many real-world applications. For example, for MRSA spread and GRAPH-HETERSIS model, indirect spread pathways (e.g., patient-HCW-patient) also play a significant role in the MRSA spread, as suggested by clinical studies [10]. However, although GRAPH-HETERSIS model can capture the pair-wise interactions well (e.g., from HCW to patients), they still struggle to model such indirect spread pathways.

Why HYPERGRAPH-HETERSIS *Model*

To overcome these limitations, we propose to use a hypergraph-based HABM. Here is an example to illustrate why hypergraphs can better capture complex group interactions compared to graphs. Consider an example trace data with 3 people (v_1, v_2, v_3) and 2 locations (L_1, L_2) as shown in Figure 3.1(a). In this example, the vertices are $V = \{v_1, v_2, v_3, L_1, L_2\}$. For the hypergraph representation, vertices v_1 and v_2 are at location L_1 between $t = 10$ to $t = 15$, which can be represented by the hyperedge in yellow. Instead, for the graph representation, we can only connect vertices v_1, v_2, and L_1 as a 3-clique. Here, if we focus on vertex v_2, we can clearly see from the hypergraph that v_2 has interacted with both v_1 and (v_1, v_3) at location L_1 across different times. However, from the graph representation, we cannot determine if v_2 interacted with both v_1 and v_3 together or with each of them separately. This example clearly indicates that hypergraphs are more suitable in capturing the group interactions than graphs.

Although people can also use more fine-grained graphs with much smaller intervals between each timestep, the computational cost also increases significantly, which makes this approach less applicable. Consequently, this motivates us to extend the graph-based HABMs to hypergraph-based HABMs. Next, we will extend GRAPH-HETERSIS introduced in subsection 3.2.1 to utilize hypergraphs in HYPERGRAPH-HETERSIS.

Overview of HYPERGRAPH-HETERSIS *Model*

Intuitively, the GRAPH-HETERSIS shown in Algorithm 1 contains two steps: (i) accumulate pathogen load from both contacts and/or self-shedding and (ii) stochastically transitioning patients between the susceptible and infected states. Specifically, we do not seek to drastically change GRAPH-HETERSIS but only modify it to include group interactions. Therefore, in HYPERGRAPH-HETERSIS, we seek to maintain these two steps, but only change how we accumulate load, keeping the state transition procedure exactly the same as before.

Instead of modeling the pathogen load spread using pairwise interactions, we aggregate load across all the group interactions for each vertex using the hypergraph matrix $\boldsymbol{H}_t$ at time t. After we aggregate loads within each hyperedge, we discount the accumulated load by a nonlinear yet invertible function $g(\cdot)$ to simulate the lost load between various interactions, which has the same size as $\boldsymbol{l}_t$. Intuitively, this makes sense since if a person moves from one location to another, some load is inevitably shed. Once we have discounted the load obtained by each vertex from each group interaction, we sum up the total load obtained across every group interaction to obtain the updated pathogen load vector. Then, we stochastically state transit patient states like GRAPH-HETERSIS model.

Design

A concise summary of the notation can also be seen in Table 3.1 and the detailed procedure can be seen in Algorithm 2. Specifically, we first calculate the pathogen transfer matrices at each time step based on the various transfer ratios Υ_{ijt} and their corresponding adjacency matrices. Then at each timestep, we sum up the loads transfered via each hyperedge together using the matrix operations, and update the pathogen load in line 4 of Algorithm 2. Specifically, Next, patients stochastically undergo state transitions (*susceptible* to *infected* or *infected* to *susceptible*) respectively.

Algorithm 2 Hypergraph-HeterSIS model procedure

Inputs: $\alpha, \beta, \delta, \{\boldsymbol{H}_t\}_{t=1}^{T}, \{\Upsilon_{ijt}\}_{i,j,t}$

Compute $\boldsymbol{\Upsilon}_t(i,j) = \begin{cases} \Upsilon_{ijt}\boldsymbol{A}_t(i,j) & \text{if } i \neq j \\ \Upsilon_{ijt} & \text{if } i = j \end{cases}$

for $t = 1, \dots, T$ **do**

  $\boldsymbol{l}_{t+1} = \alpha \boldsymbol{x}_t + \Sigma_{M_t} \boldsymbol{H}_t \cdot g(\Sigma_N(\boldsymbol{H}_t \circ \boldsymbol{l}_t \otimes \boldsymbol{\Upsilon}_t))$

  for each patient i **do**

    if i is susceptible at time t (i.e., $\boldsymbol{x}_t(i) = 0$) **then**

      i gets infected (i.e. $\boldsymbol{x}_{t+1}(i) = 1$ with probability min$\{1, \beta \boldsymbol{l}_t(i)\}$

    else

      i gets susceptible (i.e. $\boldsymbol{x}_{t+1}(i) = 0$ with prob. δ

    end if

  end for

end for

Expressivity of Hypergraph-HeterSIS *Model*

In this section, we will elaborate on how Hypergraph-HeterSIS model is more expressive than Graph-HeterSIS model and why $g(\cdot)$ is necessary. Recall that τ_{ijt} and $\boldsymbol{\Upsilon}_{ijt}$ represent the load transfer ratio from vertex j to i at time t for Graph-HeterSIS and Hypergraph-HeterSIS, respectively. Further assume m_{ijt} to be the number of hyperedges that vertex i and j colocate in at time t.

Theorem 1 (Linear Equivalence)**.** *If $g(\cdot)$ is linear,* Graph-HeterSIS *and* Hypergraph-HeterSIS *are equivalent if and only if* $\forall\ i,j \in V$, $\boldsymbol{\Upsilon}_{ijt} \propto \frac{\tau_{ijt}}{m_{ijt}}$.

Proof. Assume that pathogen loads are equal for both Graph-HeterSIS and Hypergraph-HeterSIS at time t. Then, inductively solve for parameters (τ and $g(\cdot)$) needed to make pathogen loads at time $t+1$ equal. Note that all the information necessary to make the two models equivalent is known *statically* i.e., without knowing the state of the system and only relying on external information. This is because, even though $\boldsymbol{x}_t$ and $\boldsymbol{l}_t$ are unknown, if we know τ_{ijt} and m_{ijt}, we can reweight the Hypergraph-HeterSIS (and vice versa) to make $\boldsymbol{\Upsilon}_{ijt}$ identical to the Graph-HeterSIS. □

Theorem 2 (Nonlinear Equivalence)**.** *If $g(\cdot)$ is nonlinear, then* Graph-HeterSIS *and* Hypergraph-HeterSIS *are equivalent if and only if $g(\cdot)$ is invertible and* $\forall i, j \in V$

$\boldsymbol{\Upsilon}_{ijt} = g^{-1}(\frac{\tau_{ijt}\boldsymbol{l}_t(j)}{m_{ijt}}) \;/\; \boldsymbol{l}_t(j)$.

Proof. Assume that pathogen loads are equal for both GRAPH-HETERSIS and HYPERGRAPH-HETERSIS at time t. Although we know τ_{ijt}, m_{ijt} and $g^{-1}(\cdot)$ statically, we do not know $\boldsymbol{l}_t(j)$ until we run our simulation. As this information is hidden, we cannot reweight HYPERGRAPH-HETERSIS to be equivalent to GRAPH-HETERSIS and vice versa. It is easy to see that this is because of $g^{-1}(\cdot)$, which prevents cancellation of $\boldsymbol{l}_t$. □

To summarize, if g is linear, we can map HYPERGRAPH-HETERSIS model to GRAPH-HETERSIS model directly with a transformation of the pathogen transfer ratios. However, if $g(\cdot)$ is nonlinear, we cannot. This shows that the nonlinearity of $g(\cdot)$ is what allows HYPERGRAPH-HETERSIS model to be more expressive than GRAPH-HETERSIS model. In this work, we use the $tanh(\cdot)$ function as the nonlinear function $g(\cdot)$.

3.3 Results

In section 3.3, we answer the following research questions:

- **Question 1:** Can hypergraphs better capture the differences of structural patterns of group interactions better than graphs in a real-world, clinical hospital setting?
- **Question 2:** Can HYPERGRAPH-HETERSIS better fit the number of MRSA cases (i.e., better calibration) in UVA hospital than GRAPH-HETERSIS?
- **Question 3:** After calibration, can HYPERGRAPH-HETERSIS better forecast the number of MRSA cases in UVA hospital than GRAPH-HETERSIS?
- **Question 4:** Can HYPERGRAPH-HETERSIS provide a good interpretation to help explain MRSA cases in UVA hospital?

3.3.1 Setup

Dataset

We evaluate the performance of graphs and hypergraphs on the UVA dataset. It contains a large suite of clinical metadata from the Epic-based SQL database at the University of Virginia (UVA) hospital and the weekly number of MRSA cases in the hospital. From the dataset, we construct the graphs and hypergraphs for two time periods, one representing 28 weeks before COVID-19 (Jun 18, 2019, to Dec 31, 2019), referred as the UVA-PRECOVID time period, and another representing 28 weeks since the pandemic (Jun 18, 2020, to Dec 31, 2020), referred as the UVA-COVID time period.

The interactions are derived from Electronic Healthcare Records (EHRs), which record the time and location of interactions between patients and HCWs. We aggregate these into daily graphs and hypergraphs such that vertices, patients and HCWs share an edge in a graph if a colocation event occurs between a patient and HCW on that day or, in rare instances, if two patients are forced to share a room for a given day. Edges between location vertices and either patients or HCWs occur if the HCW visits that room on a given day or a patient is located in that room on a given day. Hypergraph edges are formed between a collection of vertices if they all overlap in the same location at a given time.

Timesteps for patients within a given location are created when an admit/discharge/ transfer (ADT) action occurs within the hospital, and timesteps for HCWs are created when a service provider scans their ID upon entrance and exit of a patient's room. Sources of uncertainty arise around instances such as hospital guests, staff such as janitorial staff, and movement within rooms themselves, which are all untracked.

We run experiments on UVA's Rivanna server, which has 595 vertices with over 22598 cores and 8 PB of RAM. On each dataset, we run 100 simulation runs and report the average to evaluate the performance of graphs and hypergraphs. The basic statistics of two time periods are listed in Table 3.2.

Table 3.2: Summary statistics of UVA-PRECOVID and UVA-COVID

Time periods	UVA-PRECOVID	UVA-COVID
Number of patients	6520	6102
Number of HCWs	6614	6393
Number of Locations	586	579
Number of edges	786767	800278
Number of hyperedges	762135	790060

Calibration

Specifically, we use the Ensemble Adjustment Kalman Filter (EAKF) [127] to estimate the parameters in GRAPH-HETERSIS and HYPERGRAPH-HETERSIS model (i.e., α, β, δ, τ_{ijt}, Υ_{ijt} in Algorithm 1 and Algorithm 2). EAKF has been widely used for heterogeneous agent-based model calibration on various problems, including healthcare-associated infections [68, 121] (e.g., MRSA studied in this work), demonstrating high fidelity in recovering model parameters. We calibrate the number of patients in infected state to the weekly number of calibration MRSA cases collected from EHR data in the UVA hospital.

Metrics

To better showcase which model leads to a more accurate fit that is well-correlated to real-world MRSA outbreak trends, we use two types of measures to evaluate the performance of GRAPH-HETERSIS and HYPERGRAPH-HETERSIS: Error and trend correlation. For error metrics, following the previous work in HAI modeling [114, 128, 129, 130], we use both Normalized Root Mean Squared Error (NRMSE) and Normal Deviation (ND) as metrics for evaluation. For correlation metrics, we use Pearson correlation following the previous works [114]. Note that for both categories of measures, we use them to evaluate the performance on both calibration and forecasting period.

$$NRMSE = \frac{\sqrt{\sum_t (\hat{y}_t - y_t)^2}}{\sum_t |y_t|} \tag{3.1}$$

$$ND = \frac{\sum_t |\hat{y}_t - y_t|}{\sum_t |y_t|} \tag{3.2}$$

$$\textit{Pearson correlation} = \frac{\sum_t (\hat{y}_i - \overline{\hat{y}})(y_i - \overline{y})}{\sqrt{\sum_t (\hat{y}_i - \overline{\hat{y}})^2}\sqrt{\sum_t (y_i - \overline{y})^2}} \tag{3.3}$$

where $\hat{y}_t$ is the real-world observed number of MRSA cases in UVA hospital, y_t is the estimated value of HYPERGRAPH-HETERSIS or GRAPH-HETERSIS.

3.3.2 Q1: Capturing Structural Differences

In this section, we show that hypergraphs better capture the structural difference than graphs can. Specifically, we compare the contact networks in the University of Virginia (UVA) hospital before and during the first wave of COVID-19 outbreak in Virginia.

We focus on the difference between the contact networks for April 2019 and April 2020. For graphs, we compare basic statistics like the number of entities (patients, HCWs, and locations) and edges. For hypergraphs, we quantify hypergraphs using their fine-grained structural signatures following recent research by [131]. In their work, inspired by *graph motifs*, a combination of repeatable edge connection patterns, they propose a corresponding set of *hypergraph motifs*. Each motif represents all the connectivity patterns possible for 3 hyperedges h_1, h_2, h_3 and the emptiness or not of the seven subsets of their overlap, $h_1 \setminus h_2 \setminus h_3$, $h_2 \setminus h_1 \setminus h_3$, $h_3 \setminus h_2 \setminus h_1$, $h_1 \cap h_2 \setminus h_3$, $h_2 \cap h_3 \setminus h_1$, $h_3 \cap h_1 \setminus h_2$ and $h_1 \cap h_2 \cap h_3$. In total, there are 26 possible motifs from all possible combinations of these 7 sets (See Appendix A for a visualization of these). The relative counts for all 26 motifs serve as a signature for a particular type of hypergraph and has been effective in distinguishing different kinds of contact networks, such as co-authorship hypergraphs and email hypergraphs solely based on relative frequencies of the motifs. We use these motifs to analyze the UVA hypergraphs.

In Figure 3.2 we see that the relative counts of the 26 motifs are vastly different for

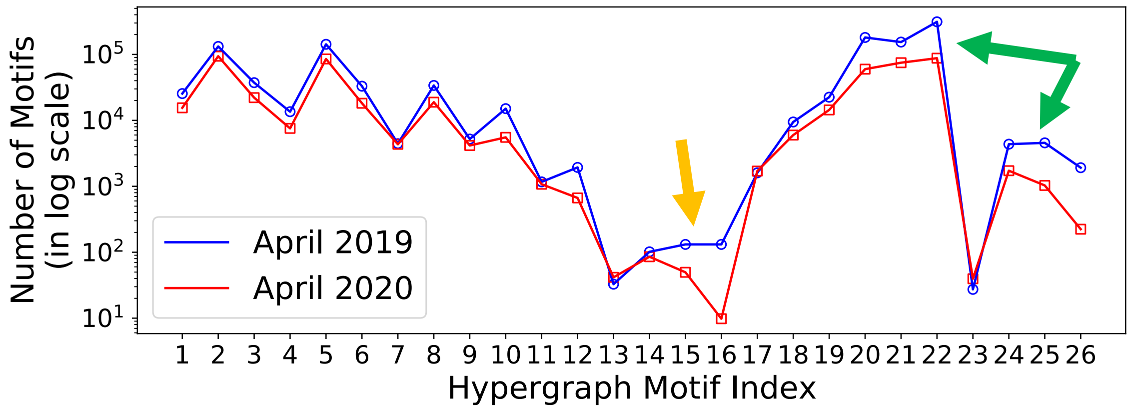


Figure 3.2: **Number of motifs for April 2019 (blue line) and 2020 (red line).** The x-axis corresponds to the index of hypergraph motifs, and the y-axis is the daily averaged number of motifs in the log scale. Note that April 2020 corresponds to the first wave of COVID-19 outbreak in Virginia. We see that motifs 15 and 16 (yellow arrow) are significantly lower in 2020 which is explained by the fact that people reduce their interactions with each others. Also note that Motifs 20-22 and 24-26 (green arrows) are also significantly lower due to fewer indirect contacts.

April 2019 in blue and April 2020 in red. Here, the x-axis corresponds to 26 different kinds of hypergraph motifs, and the y-axis is the daily averaged number of motifs (in log scale) which implies that even small deviation in the plot is highly significant. The reason for choosing these months in particular, is because April 2020 corresponds to the first wave of COVID-19 outbreak in Virginia and the previous April serves as a useful baseline to compare against.

For example, the counts of motifs 15 and 16 (yellow arrow) are significantly lower in 2020 (62.2% and 92.5%) compared to 2019. These motif IDs are those with only one or no empty subsets which implies high interconnectivity between the vertices in h_1, h_2 and h_3. This means that the number of individuals who met in groups was much smaller during COVID than before. This is easily explained by the fear of the pandemic and social distancing guidelines. Additionally, the counts of Motif 20-22 and 24-26 (green arrows) also decrease significantly (66.9%, 51.4%, 71.9% for motif 20-22, and 60.4%, 77.4%, 88.3% for motif 24-26). These motifs correspond to those with 1-2 empty sets in addition to an empty $h_1 \cap h_2 \cap h_3$. These indicate fewer indirect (e.g., patient-patient-patient) con-

Table 3.3: Basic statistics of graphs for UVA hospital in April 2019 and April 2020

Dataset	April 2019	April 2020
Number of patients	1306	795
Number of HCWs	3998	3685
Number of Locations	467	421
Number of edges	109418	77353

tacts. As shown in later experiments, HYPERGRAPH-HETERSIS captures these structural differences in the contact patterns between UVA-PRECOVID to UVA-COVID leading to better fits and forecasts. Both models are equivalent if $g(.)$ is linear, warranting the use of nonlinear group load aggregation in HYPERGRAPH-HETERSIS.

The fact that we are able to identify such complex dynamics using a simple characterization of the connection patterns showcases the effectiveness of hypergraphs in teasing out detailed structural patterns. However, it is hard to observe such dynamics intuitively from the graphs for April 2019 and 2020. We list summary statistics in Table 3.3. As shown in the table, we see that the number of patients decreases by around 40% (from 1306 to 795), while the number of HCWs and locations decreases by around 8% and 10% (from 3998 to 3685, and from 421 to 467, respectively). The edges (pair-wise interactions) decrease by around 30% (from 77353 to 109418). However, whether these pair-wise interactions are from group or one-to-one interactions are hard to identify only from the graphs.

3.3.3 Q2: MRSA Cases Calibration

We have already shown that hypergraphs can capture the structural difference better than graphs can. Next, we show that hypergraph-based HABMs can better fit and forecast MRSA cases in the UVA hospital.

To compare the performance of GRAPH-HETERSIS and HYPERGRAPH-HETERSIS in calibration, we use the MRSA spread in the UVA hospital. As mentioned in subsubsection 3.3.1, the UVA dataset also includes the weekly number of MRSA cases in the hospital. Here, we follow a real-world prediction scenario that make 4-week ahead predictions every

Table 3.4: Performance of GRAPH-HETERSIS and HYPERGRAPH-HETERSIS calibration on UVA MRSA cases

Calibration performance						
	UVA-PRECOVID			UVA-COVID		
Model	NRMSE	ND	Pearson correlation	NRMSE	ND	Pearson correlation
HYPERGRAPH-HETERSIS	**0.1319**	**0.1159**	**0.4294**	**0.1644**	**0.1494**	**0.3304**
GRAPH-HETERSIS	0.2512	0.2427	-0.1093	0.3863	0.3380	-0.4381

Forecasting performance						
	UVA-PRECOVID			UVA-COVID		
Model	NRMSE	ND	Pearson correlation	NRMSE	ND	Pearson correlation
HYPERGRAPH-HETERSIS	**0.1262**	**0.1091**	**0.2660**	**0.1213**	**0.1061**	**-0.1793**
GRAPH-HETERSIS	0.4215	0.3983	0.1228	0.4733	0.3910	-0.5355

4 weeks. This means we first calibrate on week 1-4 to predict the MRSA cases in week 5-8, then calibrate on week 1-8 and predict the MRSA cases in week 9-12 and repeat this procedure throughout the whole period.

As shown in Table 3.4, for both the UVA-PRECOVID and UVA-COVID dataset, HYPERGRAPH-HETERSIS achieves lower NRMSE and ND in calibration, indicating that HYPERGRAPH-HETERSIS better fits the real-world MRSA cases in UVA hospital. Moreover, we notice that the Pearson correlation for HYPERGRAPH-HETERSIS is significantly higher than GRAPH-HETERSIS which shows that the HYPERGRAPH-HETERSIS estimated MRSA curves are closely correlated to real-world MRSA outbreak trends.

3.3.4 Q3: Forecasting MRSA Cases

We show the performance of 4-weeks ahead predictions in Table 3.4. As shown in the table, HYPERGRAPH-HETERSIS achieves lower NRMSE, ND and higher forecast Pearson correlation. Note that both NRMSE and ND are normalized metrics, indicating that HYPERGRAPH-HETERSIS better forecasts future MRSA cases in UVA hospital than GRAPH-HETERSIS can.

Additionally, a recent study [132] reveals that only 11.1% of MRSA colonization (i.e.,

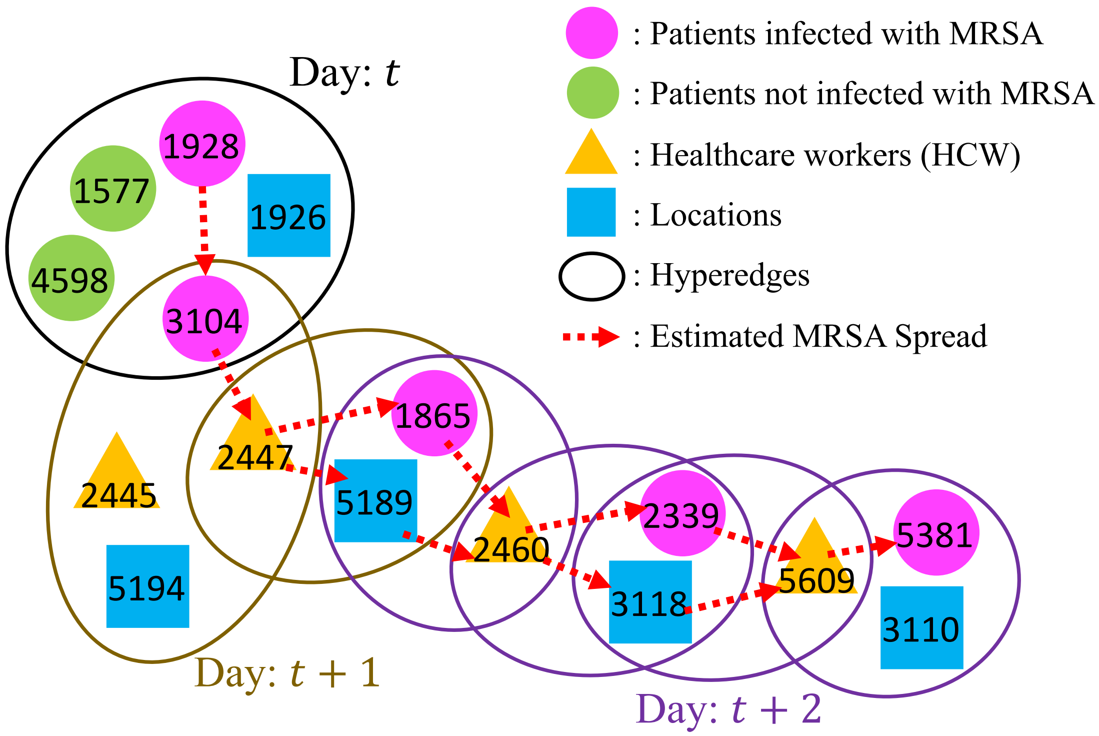


Figure 3.3: **HYPERGRAPH-HETERSIS provides a reasonable interpretation of real-world MRSA spread in the UVA hospital.** Pink and green rings, represent infected and non infected patients, yellow triangles represent healthcare workers while blue squares represent locations. Black, brown, and purple lines represent hyperedges on day t, $t + 1$, and $t + 2$, respectively. From the hypergraph, we can estimate how MRSA spread from Patient 1928 to 3104, 1865, 2339, and 5381 based on the time and connectivity of the hyperedges through the red dash arrows shown in the figure.

carry MRSA pathogens) of patients will later develop to infections and be tested. Therefore, if HYPERGRAPH-HETERSIS can forecast MRSA cases better by even 1 case, we have forecast colonized cases by almost 10 cases better. Therefore, every case counts even if the perceived improvement is marginal.

3.3.5 Q4: Interpretability

We find that HYPERGRAPH-HETERSIS is easily interpretable for MRSA spread. In Figure 3.3 we showcase a real example, with anonymized IDs, to explain the positive MRSA cases in the UVA hospital over 3 continuous days t to $t + 2$ (due to privacy concerns, we cannot release the exact dates of the example, but use t, $t + 1$, and $t + 2$ instead). Pink and green rings, represent infected and non infected patients, yellow triangles represent healthcare workers while blue squares represent locations. The circles represent the various hyperedges over different days. From Figure 3.3, we could explain the MRSA infections of 3104, 1865, 2339, and 5381 using the fact that patient 1928 tested positive for MRSA on day t and was indirectly connected to the others via the hyperedges with the estimated spread depicted by the dashed red arrows. This shows that HYPERGRAPH-HETERSIS is effective in capturing group interactions and indirect spread pathways (e.g., HCWs 2447, 2470, and 5609).

3.4 Discussion

This work proposes a new class of hypergraph-based HABMs (H^2ABM) by extending the current graph-based heterogeneous agent-based models (HABMs) to better capture the more complex dynamics of group interactions. We show that hypergraphs capture the group interactions better than graphs. We also use a widely used graph-based HABM, GRAPH-HETERSIS as an example to showcase how SIS graph-based HABM can be extended to a H^2ABM (HYPERGRAPH-HETERSIS). Our experiments show that HYPERGRAPH-HETERSIS leads to better interpretability, fitting and forecasting of MRSA cases in the UVA hospital than GRAPH-HETERSIS. Note that we choose a SIS model since it is the most suitable to describe MRSA spread. However, one could extend to other more complex compartment models (e.g., SIR, SIRD, SIRV) to hypergraph-based HABMs for other diseases. Based on our results and other literature (not just epidemics but other contagion phenomena, e.g., smoking, where smokers are known to start and stop in groups [133]), we suggest that any phenomenon that occurs in groups, be modeled as such via hypergraphs. Another interesting followup study would be to use this model to evaluate various interventions to minimize spread such as increased hand washing etc. Additionally, extending the model to account for the several asymptomatic cases, a common occurrence in practice [10], would be a useful direction.

CHAPTER 4

USING SPECTRAL CHARACTERIZATION TO IDENTIFY HEALTHCARE-ASSOCIATED INFECTION (HAI) PATIENTS FOR CLINICAL CONTACT PRECAUTION

Healthcare-associated infections (HAIs) are a major problem in hospital infection control. Although HAIs can be suppressed using contact precautions, such precautions are expensive, and we can only apply them to a small fraction of patients (i.e., a limited budget). In this work, we focus on two clinical problems arising from the limited budget: (a) choosing the best patients to be placed under precaution given a limited budget to minimize the spread (the Isolation Problem), and (b) choosing the best patients to release when limited budget requires some of the patients to be cleared from precaution (the Clearance Problem). A critical challenge in addressing them is that HAIs have multiple transmission pathways such that locations can also accumulate "load" and spread the disease. One of the most common practices when placing patients under contact precautions is the regular clearance of pathogen loads. However, standard propagation models like Independent Cascade (IC)/Susceptible-Infectious-Susceptible (SIS) cannot capture such mechanisms directly. Hence to account for this challenge, using non-linear system theory, we develop a novel spectral characterization of a recently proposed pathogen load based model, 2-MODE-SIS model, on people/location networks to capture spread dynamics of HAIs. We formulate the two clinical problems using this spectral characterization and develop effective and efficient algorithms for them. Our experiments show that our methods outperform several natural structural and clinical approaches on real-world hospital testbeds and pick meaningful solutions.

4.1 Introduction

Healthcare-associated Infections (HAIs) such as Methicillin-resistant *Staphylococcus aureus* (MRSA) and *Clostridioides difficile* (*C. diff*) are infections acquired by patients during treatment at healthcare facilities such as hospitals and long-term care homes [21, 22, 23]. While under treatment, patients can get exposed to pathogens from other sick patients or contaminated locations like rooms or medical devices. HAIs impose tremendous costs on hospitals, as they tend to extend patients' stays longer than initially planned to reach full recovery and increase the risk of mortality [36, 37, 38, 3, 39, 40]. Due to its high health burden, epidemiologists and clinicians have proposed several approaches to control HAI spread in hospitals, such as contact precautions, antimicrobial stewardship, and contact tracing [7, 8, 9, 10, 11, 12]. Among these approaches, contact precautions have been shown to be more effective in preventing further nosocomial infections [46, 47], and easier to implement [48]. Therefore, they are most commonly used in most hospitals to control HAI spread [9]. However, real-world contact precautions are expensive [1, 2, 3, 4, 5, 6] and therefore they can only be applied to a small fraction of patients (a "budget"). Further, there are several downsides to contact precautions on patient health outcomes, e.g., [6, 5]. Therefore, using contact precautions optimally is an important goal. We consider the following two important problems in HAI control with a limited budget, motivated by the above considerations.

Clinical Problem 1 (Isolation Problem, IP)**.** *Given a limited budget for precaution, how can we choose the "best" patients to put under precaution?*

Clinical Problem 2 (Clearance Problem, CP)**.** *Assume a hospital initially has a set of patients under precaution, but limited budget requires a subset of these patients to be cleared from precaution. How can we choose the "best" patients to release from precaution?*

Intuitively, IP aims to find which patients to put under precautions. In contrast to IP, the setting for CP assumes that a set of patients are already under precautions, and the

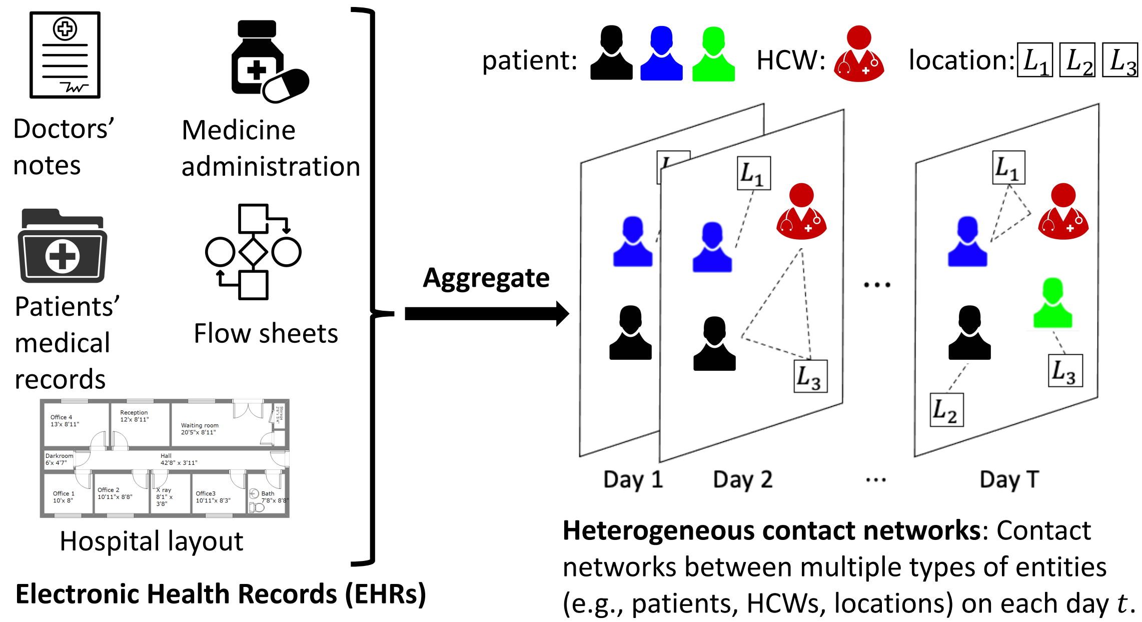
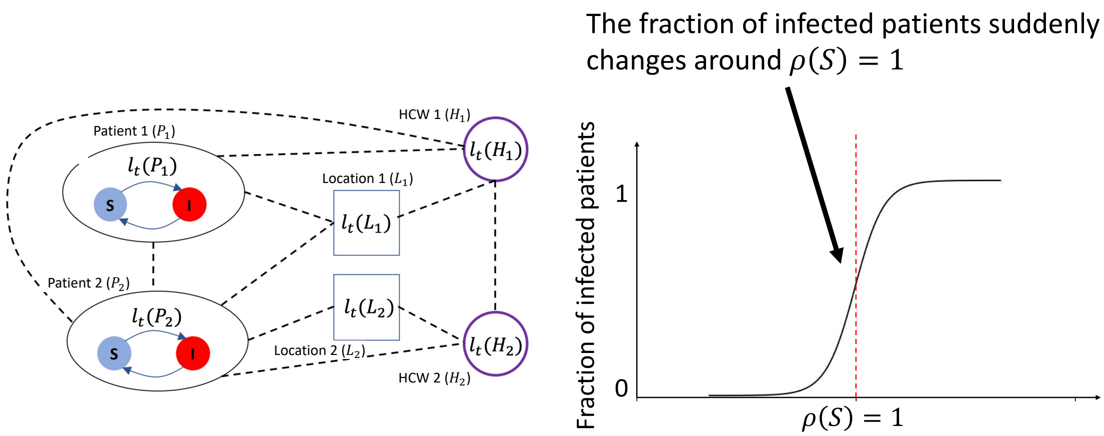


Figure 4.1: **Heterogeneous contact networks construction.** In this work, we collect the time and location of interactions between patients and HCWs from different tables in the Electronic Health Records (EHRs) dataset, such as medicine administration, admission and discharge and flow sheets, and doctors' notes. We construct daily heterogeneous contact networks from this data. For example, as shown in the figure on day T, the red healthcare worker and blue patient are in location L_1, the black patient is in location L_2, and the green patient is in location L_3.

goal is to find a subset of patients from this set to *clear* from precautions. Although there are some guidelines on removing patients from contact precautions [134, 48] from clinical perspective, this is still a novel problem in the network science research field in terms of problem formulation and optimization: It is the opposite of the well-studied influence maximization problem on social networks [73] since the objective of CP is to minimize the spread while taking off precautions from nodes. This problem is also very different from the influence containment problem [135, 136] since these problems are edge-based, while IP and CP need to remove multiple edges when only isolating/releasing one node. No good formulations exist for the clinical problems above, as we cannot easily assess how each precaution on a specific patient affects the overall epidemic.

It is important to note that healthcare workers (HCWs) and contaminated physical sur-



(a) 2-Mode-SIS Model

(b) Tipping point captured by our spectral characterization $\boldsymbol{\rho(S)}$

Figure 4.2: **2-MODE-SIS model.** (a) An example of the 2-MODE-SIS model on a heterogeneous contact network: On some day t, there are two patients (P_1, P_2), two HCWs (H_1, H_2), and two locations (L_1, L_2) in the network, and dash lines represent edges in the heterogeneous contact network (e.g., P_1 was in L_1 on day t as well as in contact with P_2 and H_1 on day t). For every node i, their pathogen load is represented as $\boldsymbol{l}_t(i)$ by 2-MODE-SIS model, and these loads can spread via contacts. For patients P_1, P_2, their infection states can transfer between S ($\boldsymbol{x}_t(i) = 0$) and I ($\boldsymbol{x}_t(i) = 1$). (b) Our spectral characterization $\rho(\boldsymbol{S})$ captures a critical "tipping point": When $\rho(\boldsymbol{S}) > 1$, the fraction of infected patients will be close to 1. When $\rho(\boldsymbol{S}) < 1$, almost none of the patients will be infected (i.e., the epidemic dies out).

faces can also spread the pathogens [24, 25, 26] in addition to the direct patient-patient contacts. However, traditional contact networks with only patient-patient contacts cannot capture such pairwise relations among diverse entities. To account for this challenge, heterogeneous contact networks are needed in modeling. Heterogeneous graphs contain multiple types of entities and can represent the diversity among nodes/pairwise relations in real-life networks. They have gathered great interest in recent years with applications in personalized recommendation [123], publication ranking [124], and drug design [125]. Here, we consider heterogeneous contact networks where each node can be one of three types: patients, HCWs, and locations. Besides, the contacts between each node are also changing with time. Hence, such heterogeneous contact networks should also be time-varying to capture the contact changes with time. We note that contact networks are not explicitly known, and need to be inferred. As shown in Figure 4.1, we infer a contact network from Electronic Health Records (EHRs) data, which includes a variety of patients' medical records and doctors' notes. Specifically, we infer interactions between HCWs and patients through medicine administration, admission and discharge and flow sheets, and doctor notes from EHRs, and aggregate them into daily heterogeneous contact networks for each day t.

Modeling on heterogeneous contact networks is different from modeling on traditional contact networks. We need to identify the role of HCWs and location nodes in spreading the pathogens and involve this in modeling. We assume that HCWs and locations can also carry some pathogen loads and spread them to patients, but HCWs do not get infected. In this section, we use a recently proposed model for HAI, the 2-MODE-SIS model proposed by Jang et al. [68], which incorporates pathogens spread through both people and locations as shown in Figure 4.2(a). Such pathogen load based models have been used in recent high-profile HAI modeling studies [137, 10, 67, 121]. Importantly, they can also model the cleaning practice widely used in contact precautions (e.g., using disinfectants to clean the ward) directly [27, 28, 29, 138, 139] compared with traditional Independent Cascade(IC) [140]/Susceptible-Infectious-Susceptible(SIS) [141] models. However, modeling with 2-MODE-SIS model is non-trivial since there are multiple transmission pathways through multiple kinds of nodes, making its dynamics harder to model.

The main idea of this work is to use spectral characterization to help formulate the two clinical problems we are interested in as network optimization problems: We give a novel spectral characterization $\rho(\boldsymbol{S})$ of the 2-MODE-SIS model on underlying time-varying people-location heterogeneous contact networks using stability analysis of discrete-time non-linear dynamical systems [142]. We first approximate the propagation dynamics of the 2-MODE-SIS model on heterogeneous networks, after which we can analyze the stability of its equilibrium point (specifically, *all-zero* point where the disease dies out) and check whether this point is stable or not. As shown in Figure 4.2(b), we show that this characterization is meaningful in capturing a “tipping point” for HAI spread dynamics. At this ”tipping point”, there will be a phase transition for the fraction of infected patients at the end: when $\rho(\boldsymbol{S}) > 1$, the fraction of infected patients will be close to 1. When $\rho(\boldsymbol{S}) < 1$, almost none of the patients will be infected (i.e., the epidemic dies out). This $\rho(\boldsymbol{S})$ is similar to the basic reproductive number R_0 [143, 144, 145, 146] for standard SIR-type models, and helps relate the structural and disease parameters to the dynamics. In the following section, we will show how we formulate the clinical problems in detail, and propose our GREEDY-SPECTRAL algorithm to address it.

4.2 Methods

4.2.1 Heterogeneous Contact Networks

$\mathcal{G}$ is a set of time-varying heterogeneous contact networks - it is sequence of graphs $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_T$ with T number of timesteps. We focus on unweighted and undirected graphs; hence each $\mathcal{G}_t$ is represented as an $N \times N$ symmetric binary adjacency matrix $\boldsymbol{A}_t$ s.t. $\boldsymbol{A}_t(i, j) = 1$ if nodes i and j are connected with an edge at time t, $\boldsymbol{A}_t(i, j) = 0$ otherwise. Note that our node-set is shared across all graphs and can represent people (patients and HCWs) as well as locations (patient rooms, nurses’ stations, etc.).

As mentioned earlier, contact networks are not explicitly available in EHR data. Instead, we infer them from various tables, such as Flowsheets, Medicine Administration, Admission Discharge, and doctor’s notes. These tables specify specific events that occurred during the patient’s stay in the hospital, e.g., admission into the hospital or ICUs, administration of different kinds of medications. One can infer the exact location of the patient, and when a HCW enters the patient room, which we use to infer co-location. The contact network is constructed using this co-location information.

Table 4.1: List of notations

Notation	Description
P	Number of patients
H	Number of Healthcare workers (HCWs)
L	Number of locations
N	Total number of nodes ($N = P + H + L$)
M	Total number of edges
T	Number of timesteps
$\boldsymbol{l}_t$	Pathogen load vector at time t
$\boldsymbol{x}_t$	Infection state vector at time t
$\boldsymbol{p}_t$	Infection probability vector at time t ($\boldsymbol{p}_t = \mathbb{E}[\boldsymbol{x}_t]$)
$\boldsymbol{A}_t$	Adjacency matrix at time t
$\boldsymbol{R}_t$	Pathogen transfer matrix at time t
$\rho(\boldsymbol{M})$	Spectral radius (i.e., the largest eigenvalue) of matrix $\boldsymbol{M}$
$\mathrm{nnz}(\boldsymbol{M})$	Number of nonzeros in matrix $\boldsymbol{M}$

4.2.2 2-MODE-SIS Model

As shown in Figure 4.2(a), we use the 2-MODE-SIS model [68] to simulate the spread of pathogens (see Algorithm 3 for simulation steps). 2-MODE-SIS is pathogen "load-based" where the patient can be either *susceptible* or *infected*, and the model keeps track of pathogen load on all people and locations using a load vector $\boldsymbol{l}_t$. Unlike classical SIS models [147], in 2-MODE-SIS model, the probability of a patient i to get infected from susceptible is formulated as a dose-response function proportional to the amount of pathogen on the patient i, or $\boldsymbol{l}_t(i)$, as shown in Algorithm 3 step 8 with β as the disease infectivity). Once infected, the patient i sheds α additional pathogen per timestep to his own load, which can later be transferred to neighbors (both people and locations) via edges; this shedding continues until the patient recovers with recovery probability δ. Specifically, in this model, each node carries some amount of pathogens that changes over time, and the exchange of pathogens among the nodes is driven by edges that imply close contact among the nodes. 2-MODE-SIS model uses τ_{ijt} to denote the ratio of pathogen being transferred (or remaining if $i = j$) from node j to node i at time t, and ϕ_i to denote the natural pathogen reduction rate on node i. By constructing a transfer matrix $\boldsymbol{R}_t$ from the transfer ratios τ_{ijt}, reduction

Table 4.2: List of 2-MODE-SIS model parameters

Notation	Description
α	Pathogen shedding rate
β	Disease infectivity
δ	Recovery probability
ϕ_i	Natural pathogen reduction rate on node i
τ_{ijt}	Transfer ratio from node j to node i at time t

rates ϕ_i, and adjacency matrix $\boldsymbol{A}_t$, the pathogen load updates can be written as a linear operation as in Algorithm 3 step 5. Here, $\boldsymbol{x}_t$ is the infection state vector for all patients at time t, where the ith element, $\boldsymbol{x}_t(i)$, corresponds to patient i. α is pathogen shedding rate for infected patients. Note that the column-sums of $\boldsymbol{R}_t$ are restricted to be less than or equal to 1, which implies that the total amount of pathogen cannot increase after transfer (i.e., $\|\boldsymbol{R}_t\boldsymbol{l}_t\|_1 \leq \|\boldsymbol{l}_t\|_1$). Susceptible patients may still be colonized with pathogen loads and can spread them to others. HCWs and locations are assumed to be non-infectable, but act as pathways of pathogen transfer.

Algorithm 3 2-MODE-SIS model

1: **Inputs:** $\Theta = \{\alpha, \beta, \delta, \{\boldsymbol{A}_t\}_{t=1}^{T}, \{\tau_{ijt}\}_{i,j,t}\}$
2: Initialize infection-states $\boldsymbol{x}_1$ and loads $\boldsymbol{l}_1$
3: Compute $\boldsymbol{R}_t(i,j) = \begin{cases} \tau_{ijt}\boldsymbol{A}_t(i,j) & \text{if } i \neq j \\ \tau_{ijt}\phi_i & \text{if } i = j \end{cases}$ for all t.
4: **for** $t = 1, \ldots, T$ **do**
5:     Update loads $\boldsymbol{l}_{t+1} = \boldsymbol{R}_t\boldsymbol{l}_t + \alpha\boldsymbol{x}_t$.
6:     **for** each patient i **do**
7:         **if** i is susceptible at time t (i.e., $\boldsymbol{x}_t(i) = 0$) **then**
8:             i gets infected (i.e. $\boldsymbol{x}_{t+1}(i) = 1$) with probability $\beta\boldsymbol{l}_t(i)$.
9:         **else**
10:             i gets susceptible (i.e. $\boldsymbol{x}_{t+1}(i) = 0$) with probability δ.
11:         **end if**
12:     **end for**
13: **end for**

4.2.3 Spectral Characterization for the 2-MODE-SIS Model

Approximation via NLDS

We develop a characterization to capture the overall dynamics of the system and derive a spectral characterization for the 2-MODE-SIS model. Our proposed approach is to approximate the propagation dynamics of the 2-MODE-SIS model using the stability analysis of discrete-time non-linear dynamical systems (NLDS) with two coupled states: the pathogen loads $\boldsymbol{l}_t$ and the expectation of infection probabilities $\boldsymbol{p}_t = \mathbb{E}[\boldsymbol{x}_t]$. Let $\boldsymbol{s}_t = \begin{bmatrix} \boldsymbol{p}_t \\ \boldsymbol{l}_t \end{bmatrix} \in \mathbb{R}^{(P+N)}$ be the vector describing the state of the system at timestep t, we can write down the state transition updates at t and use linearity of expectation to get the vector for next timestep as $\boldsymbol{s}_{t+1} = g_t(\boldsymbol{s}_t)$, where g_t is a non-linear mapping.

$$\boldsymbol{s}_{t+1} = g_t(\boldsymbol{s}_t) := \begin{bmatrix} (1-\delta)\boldsymbol{p}_t + \beta \boldsymbol{l}_t (1 - \boldsymbol{p}_t) \\ \boldsymbol{R}_1 \boldsymbol{l}_t + \alpha \boldsymbol{p}_t \end{bmatrix} \tag{4.1}$$

Stability Analysis of the NLDS

With the non-linear equations of our approximated discrete-time NLDS, we can then analyze its asymptotic behavior. The long-term behavior of a dynamical system is dictated by the stability of its equilibrium points, at which the states no longer change over time (i.e., $\boldsymbol{s}^{eq} = g(\boldsymbol{s}^{eq})$). The particular equilibrium point of interest is the *all-zero* point $\mathbf{0}$ where there is no infection as well as no pathogen remaining. Clearly, all g_t map $\mathbf{0}$ onto itself and $\mathbf{0}$ is an equilibrium point. Specifically, we write the Jacobian matrix evaluated at $\mathbf{0}$ as $\boldsymbol{S}_t$

$$\boldsymbol{S}_t = \nabla g\big|_{\boldsymbol{s}_t=\mathbf{0}} = \left.\frac{\partial \boldsymbol{s}_{t+1}}{\partial \boldsymbol{s}_t}\right|_{\boldsymbol{s}_t=\mathbf{0}} = \begin{bmatrix} (1-\delta)\boldsymbol{I}_P & \beta \boldsymbol{I}_P \quad \mathbf{0} \\ \alpha \boldsymbol{I}_P & \boldsymbol{R}_1 \\ \mathbf{0} & \end{bmatrix} \tag{4.2}$$

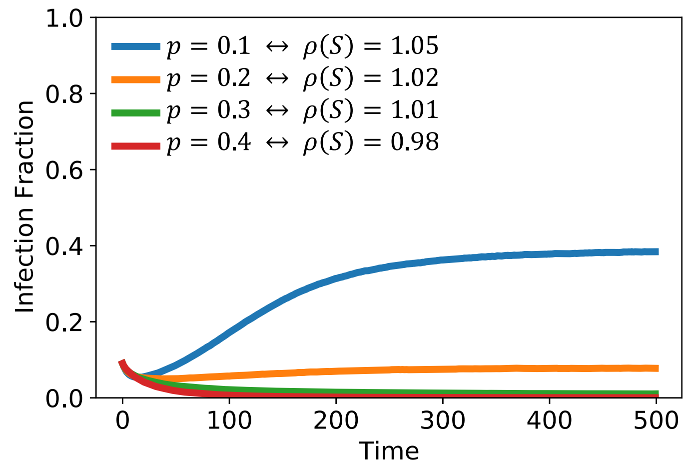


Figure 4.3: **Example of "dilution effect" on static Erdős-Rényi random graphs.** With increasing link probability p, the simulation results show that the infection fraction decreases.

Here, I_P represents the identity matrix sized $P \times P$. The following Theorem 3 on asymptotic stability in a discrete-time NLDS shows what condition this equilibrium point is "stable" [142], i.e. a small perturbation does not cause large deviations.

Theorem 3 (Spectral characterization)**.** *If the spectral radius (i.e., the largest eigenvalue) of the system matrix $\boldsymbol{S} = \prod_{t=1}^{T} \boldsymbol{S}_{T-t+1}$ is less than 1 (i.e. $\rho(\boldsymbol{S}) < 1$), then the all-zero equillibrium of the NLDS is asymptotically stable.*

Theorem 3 demonstrates that $\rho(\boldsymbol{S}) = 1$ is a critical tipping point that there will be a phase transition for the fraction of infected patients at the end: When $\rho(\boldsymbol{S}) > 1$, there is always a fraction of infected patients. When $\rho(\boldsymbol{S}) < 1$, almost no patients will be infected (i.e., the epidemic dies out). The tipping point behavior is similar to that associated with the basic reproductive number R_0. The characterization in Theorem 3 also helps explain the counter-intuitive "dilution effect" (i.e., the number of infections decreases with increasing link probability between patients) observed in many such load-based models [68, 148, 149]. In Figure 4.3. we show the simulation results on static Erdős-Rényi random graphs with increasing link probability p. The transfer ratios τ_{ijt} for $i \neq j$ and ϕ_i are fixed, such that the only difference between the curves are the number of edges in the graph and the proportion of pathogen remaining in each node after transfer (τ_{ijt} for $i = j$). As p increases, the number of edges are also increasing in this situation. However, we observe that the number of infections at the end are decreasing with increasing p. This is surprising, because this goes against our intuition that more contacts lead to more infections. This also contrasts with previous Independent Cascade/Linear Threshold/SIR style models where adding an edge leads to more infections [73]. However, our $\rho(\boldsymbol{S})$ correctly captures this dynamics. The reason is that with more edges in the graph, more pathogen is sent out and less remains, which means lower τ_{ijt} for $i = j$. Hence, the diagonal elements of $\boldsymbol{S}$ decreases, and $\rho(\boldsymbol{S})$ also decreases.

4.2.4 Applications of $\rho(S)$ for Clinical Problems

Using $\rho(S)$ to Formalize Clinical Problems

Next, we show how our spectral characterization $\rho(\boldsymbol{S})$ can help formulate the two clinical problems IP and CP. The intuition is that our spectral characterization well captures the transition dynamics of the HAI breakout, and a lower $\rho(\boldsymbol{S})$ value should correspond to fewer infections. We first formalize the IP problem as follows:

Formal Isolation Problem (FIP). *Let $\boldsymbol{S}(\Theta, P)$ denote the system matrix given input* 2-MODE-SIS *model parameters Θ (e.g., α, β, $\delta \cdots$) and patients in set P under precaution. Given all patients X and a budget constraint k, find the subset of patients P^* for precaution s.t.*

$$P^* = \arg\min_{P} \rho(\boldsymbol{S}(\Theta, P)) \textit{ subject to } P^* \subset X \textit{ and } |P^*| = k$$

.

This formulation uses $\rho(\boldsymbol{S}(\Theta, P))$ to capture how isolating patients in P will affect the epidemic. Here, we formally model bringing a patient i under precaution by zeroing out off-diagonal terms in the i-th row and column in each pathogen transfer matrix $\boldsymbol{R}_t$ (i.e., cut off its links with other nodes), and reducing the diagonal terms $\boldsymbol{R}_t(i,i)$ by a multiplicative factor $\phi_{ppe} < 1$ (i.e., reduce its remaining pathogen via cleaning). They are widely used together in hospitals for contact precautions [8, 48, 150]. Note that the current setup can be readily modified to capture other contact precaution practices while maintaining a similar problem formulation. Similarly, we can also formalize the CP problem:

Formal Clearance Problem (FCP). *Let $\boldsymbol{S}(\cdot)$ be defined as in FIP. Given an initial set Q of d candidate patients already under precaution ($|Q| = d$) and a budget constraint $k < d$, find the subset of patients P^* for clearance s.t.*

$$P^* = \arg\min_P \rho(\boldsymbol{S}(\Theta, Q \setminus P)) \textit{ subject to } P \subset Q \textit{ and } |P| = k$$

4.2.5 GREEDY-SPECTRAL Algorithm

Numerical Issue in Computing $\rho(\boldsymbol{S})$

A natural algorithm for FIP or FCP would try adding (or removing) each candidate patient P under (or from) precaution and then compute by how much $\rho(\boldsymbol{S})$ changes to incrementally build a feasible solution set while minimizing $\rho(\boldsymbol{S})$. In practice, however, we find that computing the change in $\rho(\boldsymbol{S})$ due to a single node removal runs into numerical issues: If there are much more patient-to-non-patient interactions than patient-to-patient interactions, the change in $\rho(\boldsymbol{S})$ w.r.t. P can be too small to be computed within finite precision. The issue even gets worse when the total number of nodes N and the number of timesteps T increase.

Justification. To see why, let us decompose the pathogen transfer matrix $\boldsymbol{R}_t$ in $\boldsymbol{S}_t$ into rows corresponding to patients (who can be infected) and rows corresponding to non-patients (who cannot be infected). Then the $\boldsymbol{R}_{t,pp}$ submatrix of $\boldsymbol{R}_t$ encodes the patient-to-patient interactions at time t, $\boldsymbol{R}_{t,ph}$ encodes the patient-to-non-patient interactions at time t, and so on. Note that when entries in $\boldsymbol{R}_t$ are much smaller than $(1-\delta)$ and α (as in real parameter settings), the spectrum of $\boldsymbol{S}_t$ is mostly governed by the upper-left submatrix of $\boldsymbol{S}_t$ (enclosed in red square).

$$\boldsymbol{S}_t = \begin{bmatrix} (1-\delta)\boldsymbol{I}_p & \beta\boldsymbol{I}_p & \boldsymbol{0} \\ \alpha\boldsymbol{I}_p & \boldsymbol{R}_{t,pp} & \boldsymbol{R}_{t,ph} \\ \boldsymbol{0} & \boldsymbol{R}_{t,hp} & \boldsymbol{R}_{t,hh} \end{bmatrix}$$

This can be seen by viewing $\boldsymbol{R}_{t,ph}$ and $\boldsymbol{R}_{t,hp}$ as small perturbations added to a block-diagonal matrix of which the eigenvalues equal those of the upper-left submatrix in red, and $\boldsymbol{R}_{t,hh}$. Hence if the majority of interactions are between patients and non-patients rather than among patients, most changes will occur in $\boldsymbol{R}_{t,ph}$ and $\boldsymbol{R}_{t,hp}$, but these changes due to link removal can be overshadowed and lead to a change in $\rho(\boldsymbol{S})$ too small to be captured under finite precision. We observe that this is actually the case in our real-world testbeds: Only 0.005% of contacts are among patients in UVA-PRECOVID dataset, and 0.004% in UVA-COVID dataset.

Alternative Strategy and Final Algorithm

Hence we propose an alternate strategy to solve FIP and FCP by minimizing an upper bound on $\rho(\boldsymbol{S})$ that can be computed more precisely. In particular, we define a *system-adjacency* matrix as $\boldsymbol{A} \coloneqq \prod_{t=1}^{T} \boldsymbol{A}_{T-t+1}$, where $\boldsymbol{A}_t$ is the adjacency matrix for time t, and minimize its the largest eigenvalue $\rho(\boldsymbol{A})$ as a surrogate to minimize the upper bound of $\rho(\boldsymbol{S})$. The following theorem shows that $\rho(\boldsymbol{S})$ is upper bounded by a *monotonic* function of $\rho(\boldsymbol{A})$, justifying our approach of minimizing $\rho(\boldsymbol{A})$ instead.

Theorem 4. *Let $\boldsymbol{S}$, $\boldsymbol{A}$ denote the system and system-adjacency matrices of* 2-MODE-SIS *model with T number of timesteps. Then $\rho(\boldsymbol{S}) \le f(\rho(\boldsymbol{A}))$, where*

$$
\begin{aligned}
f(x) = \frac{1}{2}\Bigg\{ & (1-\delta)^T + Cx + 1 \\
& + \ \mathrm{sqrt}\Bigg(\left[(1-\delta)^T + Cx + 1\right]^2 \\
& - 4\left[(1-\delta)^T (Cx+1) - \alpha\beta(1-\delta)^{2T-2}\right] \Bigg) \Bigg\},
\end{aligned}
$$

$C = \sum_{k=0}^{T-1} \tau_{max}^{T-k} \binom{T}{k}$, and $\tau_{max} = \max_{i \neq j, t}\{\tau_{ijt}\}$. Further, $f(x)$ is a monotonic increasing function in x.

Proof. Lemma 1 first states a general upper bound on the spectral radius of any matrix $\boldsymbol{M}$ structured in a particular way. This lemma can be proven using Schur complements and the interlacing theorem on a principal submatrix.

Lemma 1. *Given non-negative scalars a, b, c, non-negative $n \times n$ matrix $\boldsymbol{O}$, and $(n+p) \times (n+p)$ matrix $\boldsymbol{M}$:*

$$
\boldsymbol{M} = \begin{bmatrix} a\boldsymbol{I}_p & b\boldsymbol{I}_p \quad \boldsymbol{0} \\ c\boldsymbol{I}_p & \\ \boldsymbol{0} & \quad \boldsymbol{O} \end{bmatrix}
$$

where $\boldsymbol{I}_p$ denotes the $p \times p$ identity matrix with $p < n$. Then, for the spectral radius of $\boldsymbol{M}$ we have $\rho(\boldsymbol{M}) \le g(\rho(\boldsymbol{O}), a, b, c)$, where $g(x, a, b, c) = \frac{1}{2}\left(a + x + \sqrt{(a+x)^2 - 4(ax - bc)}\right)$. Further, $g(x, a, b, c)$ is monotonic increasing w.r.t. x.

Proof. Let $\boldsymbol{M}'$ denote a matrix that is almost the same as $\boldsymbol{M}$, but does not have the zeros filling up the gap:

$$
\boldsymbol{M}' = \begin{bmatrix} a\boldsymbol{I}_n & b\boldsymbol{I}_n \\ c\boldsymbol{I}_n & \boldsymbol{O} \end{bmatrix}
$$

Then, using the Schur complement leads to

$$\begin{aligned}\det(\boldsymbol{M}' - \lambda \boldsymbol{I}) &= \det\left((\boldsymbol{O} - \lambda \boldsymbol{I})(a\boldsymbol{I} - \lambda \boldsymbol{I}) - bc\boldsymbol{I}\right) \\ &= \det\left((a-\lambda)\boldsymbol{O} - (\lambda(a-\lambda) + bc)\boldsymbol{I}\right)\end{aligned}$$

This means that λ is an eigenvalue of $\boldsymbol{M}'$ if and only if $\mu = \dfrac{\lambda(a-\lambda)+bc}{a-\lambda}$ is an eigenvalue of $\boldsymbol{O}$. This is a quadratic equation of λ, of which the solution is

$$\lambda = \frac{1}{2}\left[a + \mu \pm \sqrt{(a+\mu)^2 - 4(a\mu - bc)}\right] =: v(\mu)$$

Next, to show that $\rho(\boldsymbol{M}') = v(\rho(\boldsymbol{A}))$, we show that $v(\cdot)$ is a monotonically non-decreasing function (i.e., $\dfrac{\partial v}{\partial \mu} \geq 0$)

$$\begin{aligned}\frac{\partial v}{\partial \mu} &= 1 \pm \frac{2(a+\mu) - 4a}{2\sqrt{(a+\mu)^2 - 4(a\mu - bc)}} \\ &= 1 \pm \frac{\mu - a}{\sqrt{(a+\mu)^2 - 4(a\mu - bc)}} \\ &= 1 \pm \frac{\mu - a}{\sqrt{(\mu - a)^2 + 4bc}} \\ &\geq 0 \quad \text{(since } b, c \leq 0\text{)}\end{aligned}$$

Lastly, note that $\boldsymbol{M}$ is a principle submatrix of $\boldsymbol{M}'$. Since $\boldsymbol{M}'$ is non-negative, applying interlacing theorem of principle submatrix [151] leads to $\rho(\boldsymbol{M}) \leq \rho(\boldsymbol{M}')$ and this proves the lemma. $\square$

Using Lemma 1 and the fact that $\boldsymbol{S}$ is structured similarly to $\boldsymbol{M}$, Lemma 2 shows that $\rho(\boldsymbol{S})$ is upper bounded by a monotonic function of $\rho(\boldsymbol{R})$, where $\boldsymbol{R}$ denotes an intermediate *system-transfer* matrix $\boldsymbol{R} \coloneqq \prod_{t=1}^{T} \boldsymbol{R}_{T-t+1}$.

Lemma 2. *Given $\boldsymbol{S}$, $\boldsymbol{R}$ and $g(\cdot)$ as before, then*

$$\rho(\boldsymbol{S}) \leq g\left(\rho(\boldsymbol{R}), (1-\delta)^T, \alpha(1-\delta)^{T-1}, \beta(1-\delta)^{T-1}\right).$$

Proof. Say we have $T = 2$.

$$\begin{aligned}
\boldsymbol{S}_2\boldsymbol{S}_1 &= \begin{bmatrix} (1-\delta)\boldsymbol{I}_p & \beta\boldsymbol{I}_p \quad \boldsymbol{0} \\ \alpha\boldsymbol{I}_p & \\ \boldsymbol{0} & \boldsymbol{R}_2 \end{bmatrix} \begin{bmatrix} (1-\delta)\boldsymbol{I}_p & \beta\boldsymbol{I}_p \quad \boldsymbol{0} \\ \alpha\boldsymbol{I}_p & \\ \boldsymbol{0} & \boldsymbol{R}_1 \end{bmatrix} \\
&= \begin{bmatrix} ((1-\delta)^2 + \alpha\beta)\boldsymbol{I}_p & \begin{bmatrix}\beta\boldsymbol{I}_p & \boldsymbol{0}\end{bmatrix} + \beta\boldsymbol{R}_1^{prow} \\ \begin{bmatrix}\alpha\boldsymbol{I}_p \\ \boldsymbol{0}\end{bmatrix} + \alpha\boldsymbol{R}_2^{pcol} & \boldsymbol{R}_2\boldsymbol{R}_1 + \begin{bmatrix}\alpha\beta\boldsymbol{I}_p & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0}\end{bmatrix} \end{bmatrix} \\
&\approx \begin{bmatrix} (1-\delta)^2\boldsymbol{I}_p & \beta(1-\delta)\boldsymbol{I}_p \quad \boldsymbol{0} \\ \alpha(1-\delta)\boldsymbol{I}_p & \\ \boldsymbol{0} & \boldsymbol{R}_2\boldsymbol{R}_1 \end{bmatrix}
\end{aligned}$$

For $T > 2$, repeatedly multiplying and omitting high-order terms leads to following approximation:

$$\boldsymbol{S} \approx \begin{bmatrix} (1-\delta)^T\boldsymbol{I}_p & \beta(1-\delta)^{T-1}\boldsymbol{I}_p \quad \boldsymbol{0} \\ \alpha(1-\delta)^{T-1}\boldsymbol{I}_p & \\ \boldsymbol{0} & \boldsymbol{R} \end{bmatrix}$$

This approximation has the exact structure discussed in Lemma 1. Applying the lemma proves the claim. $\square$

Note that having less non-infectable nodes leads to a tighter upper bound. Hence, starting from Lemma 2, we can obtain an upper bound on $\rho(\boldsymbol{S})$ in terms of $\rho(\boldsymbol{A})$ as such:

$$\begin{aligned}\rho(\boldsymbol{S}) &\leq g(\rho(\boldsymbol{R}), (1-\delta)^T, \alpha(1-\delta)^T, \beta(1-\delta)^T) \\ &\leq g(C\rho(\boldsymbol{A})+1, (1-\delta)^T, \alpha(1-\delta)^T, \beta(1-\delta)^T)\end{aligned}$$

The second inequality is due to monotonicity of $g(\cdot)$ with respect to the first input, and $\rho(\boldsymbol{R}) \leq C\rho(\boldsymbol{A}) + 1$ from Lemma 3. Expanding the last term leads to the desired result.

Next, Lemma 3 shows that $\rho(\boldsymbol{R})$ is also upper bounded by a monotonic function of $\rho(\boldsymbol{A})$ (which contains only zeros and ones) by viewing $\boldsymbol{R}$ as a Hadamard product of a constant matrix times $\boldsymbol{A}$, plus some diagonal terms.

Lemma 3. *Given $\boldsymbol{R}$, $\boldsymbol{A}$ as before, $\rho(\boldsymbol{R})$ is upper bounded by a monotonic function of $\rho(\boldsymbol{A})$: $\rho(\boldsymbol{R}) \leq C\rho(\boldsymbol{A}) + 1$ (C and τ_{max} as given in Theorem 4).* □

Proof. Note that the individual transfer matrices $\boldsymbol{R}_t$ are formed by a Hadamard product plus diagonal terms: $\boldsymbol{R}_t = \boldsymbol{T}_t \circ \boldsymbol{A}_t + \boldsymbol{D}_t$. Here $\boldsymbol{T}_t$ stores the transfer rates $[\boldsymbol{T}_t]_{i,j} = \tau_{ijt}$, and $\boldsymbol{D}_t$ are the additional diagonal terms. Then, the product $\boldsymbol{R}$ can be written as

$$\begin{aligned}\boldsymbol{R} = \prod_{t=1}^{T} \boldsymbol{R}_t = \prod_{t=1}^{T} \boldsymbol{T}_t \circ \boldsymbol{A}_t + \boldsymbol{D}_t &\leq \prod_{t=1}^{T} \tau_{max} \boldsymbol{A}_t + \boldsymbol{I} \\ &= \tau_{max}^T \prod_{t=1}^{T} \boldsymbol{A}_t + \tau_{max}^{T-1} \sum_{\substack{S \subset \{1,\ldots,T\} \\ |S|=T-1}} \prod_{t \in S}^{T} \boldsymbol{A}_t + \cdots + \boldsymbol{I} \\ &\leq \tau_{max}^T \boldsymbol{A} + \tau_{max}^{T-1} \binom{T}{1} \boldsymbol{A} + \cdots + \boldsymbol{I} \\ &= \sum_{k=0}^{T-1} \tau_{max}^{T-k} \binom{T}{k} \boldsymbol{A} + \boldsymbol{I}\end{aligned}$$

where the inequalities are element-wise. Then using Weyl's inequality leads to the desired upper-bound on $\rho(\boldsymbol{R})$. □

Theorem 4 then follows from Lemma 2 and Lemma 3 due to monotonicity of the upper bounds. □

Fortunately, $\rho(\boldsymbol{A})$ does not suffer from the same numerical issue as $\rho(\boldsymbol{S})$ does. However, the two surrogate problems are also NP-hard. Hence we propose a simple yet effective heuristic algorithm, GREEDY-SPECTRAL, for SIP and SCP: Starting from an empty set, we iteratively find and add a patient for isolation (or releasing) that leads to the smallest $\rho(\boldsymbol{A})$ value until the size of the set reaches the given budget. Note that in each iteration of GREEDY-SPECTRAL, we need to recalculate the change in $\rho(\boldsymbol{A})$ after isolating or releasing every node, requiring $O(kN)$ number of $\rho(\boldsymbol{A})$-computations. Each $\rho(\boldsymbol{A})$-computation can be time-consuming when done naively through computing and storing the full dense $\boldsymbol{A}$ and then computing its spectral radius, which takes $O(N^{2.37}T)$ time [152]. Instead, we leverage the sparsity of individual $\boldsymbol{A}_t$, and use power iterations to compute $\rho(\boldsymbol{A})$ without explicitly constructing $\boldsymbol{A}$. The following Lemma 4 states the computational cost of this approach.

Lemma 4. *Computing $\rho(\boldsymbol{A})$ with the Arnoldi method using implicit linear operator $\boldsymbol{A}_{op}$ requires $O(\sum_{t=1}^{T} \mathrm{nnz}(\boldsymbol{A}_t))$ time and $O(N)$ space per iteration, where $\mathrm{nnz}(\boldsymbol{A}_t)$ denotes the number of nonzeros in $\boldsymbol{A}_t$.*

Proof. Since the Arnoldi method only requires a function that computes matrix-vector products with the query matrix, we can use a linear operator $\boldsymbol{A}_{op} : \boldsymbol{x} \mapsto \boldsymbol{A}_T \cdots \boldsymbol{A}_1 \boldsymbol{x}$ that does not explicitly compute $\boldsymbol{A}$ and then apply to $\boldsymbol{x}$, but rather applies $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, up to $\boldsymbol{A}_T$ one-by-one to the input vector $\boldsymbol{x}$.

Hence, when searching for the largest-magnitude eigenvalue of $\boldsymbol{A}$, each i-th iteration of the Arnoldi method first computes the matrix-vector product $\boldsymbol{q} = \boldsymbol{A}_{op}(\boldsymbol{v}_i)$ and then normalizes the result $\boldsymbol{v}_{i+1} = \frac{\boldsymbol{q}}{\|\boldsymbol{q}\|}$. Thus, each iteration only needs to store a length-N vector, which leads to the $O(N)$ space complexity. The time complexity is dominated by the cost of computing $\boldsymbol{A}_{op}(\boldsymbol{v})$. Within $\boldsymbol{A}_{op}$, computing the sparse matrix-vector product $\boldsymbol{A}_t\boldsymbol{v}$ requires $O(\mathrm{nnz}(\boldsymbol{A}_t))$ time: each term in $\boldsymbol{A}_t$ is multiplied by an entry in $\boldsymbol{v}$ and added to the output vector only once. Therefore, aggregating the cost of T sparse matrix-vector products in $\boldsymbol{A}_{op}$ leads to $O(\sum_{t=1}^{T} \mathrm{nnz}(\boldsymbol{A}_t))$ time. $\square$

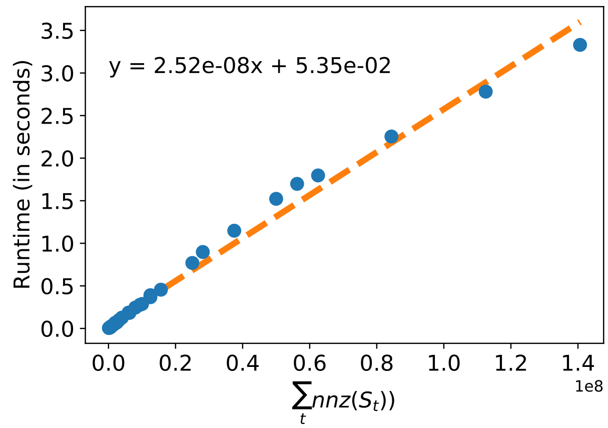


Figure 4.4: **Runtime plot of computing $\rho(\boldsymbol{A})$ with the Arnoldi method.** The actual runtimes (in blue) increase linearly to the total number of nonzeros in the $\mathbf{A}_t$ matrices.

When using the Arnoldi method, the overall computational cost of GREEDY SPECTRAL can be reduced down to $O(kn \sum_{t=1}^{T} \mathrm{nnz}(\boldsymbol{A}_t))$-time and $O(N)$-space complexity as shown in Figure 4.4. Additionally, the change in $\rho(\boldsymbol{A})$ after adding each node to the solution set can be computed in parallel for further speedup.

4.3 Results

In this section, we demonstrate the performance of our GREEDY-SPECTRAL algorithm for clinical problems using MRSA (a kind of HAI) as an example. Here, we use two time-varying heterogeneous contact networks constructed from clinical metadata (e.g., inpatient Data, doctor's Notes, and MedAdmin Data) from the Epic-based SQL database at the University of Virginia (UVA) hospital to showcase our GREEDY-SPECTRAL algorithm could achieve robust good performance. One represents 294 days before COVID-19 (May 2019 to Feb 2020), referred as the UVA-PRECOVID network, and another represents 294 days since the pandemic (May 2020 to Feb 2021), referred as the UVA-COVID network. In these two networks, MRSA rates significantly declined in the UVA-COVID period, while networks became slightly denser, though many local properties (e.g., degree distribution, local clustering) were the same [153]. We first demonstrate the performance of GREEDY-SPECTRAL algorithm and several competing baselines in suppressing the MRSA outbreak

Table 4.3: Basic statistics of real networks.

Notations	UVA-PRECOVID	UVA-COVID
T	294	294
P	8193	7600
H	4615	4526
L	595	596
N	13403	12722
M	2.16M	2.32M

for both high and relaxed precaution and cleaning scenarios (we will explain the two scenarios in later sections). We then investigate the patients picked by GREEDY-SPECTRAL using their electronic health records (EHRs) to see if they are meaningful from a clinical perspective.

4.3.1 Experiment Setup

Datasets

We evaluate the performance of GREEDY-SPECTRAL on various real datasets (see Table 4.3, the meaning of notations are listed in Table 4.1). On each dataset, we run 1000 simulation runs and report the average. For UVA datasets, we run the simulation till 294 timesteps (covering the entire given time period). This dataset contains a large suite of clinical metadata from the Epic-based SQL database at the University of Virginia (UVA) hospital. It also contains the weekly number of new tested positive MRSA cases in the hospital, and a rich set of features for each patient. From the dataset, we construct two time-varying heterogeneous contact networks, one representing 294 days before COVID-19 (May 2019 to Feb 2020), referred as the UVA-PRECOVID network, and another representing 294 days since the pandemic (May 2020 to Feb 2021), referred as the UVA-COVID network. The networks are derived from inpatient Data, doctor's Notes, and MedAdmin Data, which record the time and location of interactions between patients and HCWs. We aggregate these into daily networks such that two nodes are connected in $\mathcal{G}_t$ if they have at least one contact at day t.

Baselines

We compare GREEDY-SPECTRAL algorithm against the following popular methods:

- RANDOM: Randomly pick k patients.
- PROPDEGREE: Pick k patients with the probability of picking i proportional to the sum of the nonzero terms of i in R_t (intuitively, the degree of i) over all timesteps (i.e., $p \propto \sum_t \sum_j \mathbb{1}(R_t(j,i))$).
- SHORR: Pick k patients with the highest so-called Shorr scores [154]; Shorr score was developed to assess the clinical risk of an admitted patient acquiring MRSA, and variables used include the history of recent hospitalizations or kidney dialysis. Note that Shorr score is one of the classical clinical criteria used to pick patients for MRSA contact precaution [154], we compare with SHORR to showcase that our GREEDY-SPECTRAL is more effective than such usual practices using patient-level features.
- OPTIMALSPECTRAL: Pick the optimal k patients that minimize $\rho(\boldsymbol{A})$ by testing all possible combinations.

Evaluation metrics

We use the following metrics to quantify the performance in suppressing MRSA outbreak.

- $\eta_{\rho(\boldsymbol{A})}$: Ratio of $\rho(\boldsymbol{A})$ with $\mathcal{P}$ under precaution against $\rho(\boldsymbol{A})$ when $k = 0$ (no patient under precaution) (i.e. $\eta_{\rho(\boldsymbol{A})} = \frac{\rho(\boldsymbol{A}(\mathcal{P}))}{\rho(\boldsymbol{A}(\emptyset))}$). Lower is better.
- γ_{CASES}: Ratio of the number of new MRSA cases against $k = 0$. (e.g. γ_{CASES} for PROPDEGREE is $\frac{\text{CASES}_{\text{PROPDEGREE}}}{\text{CASES}_{k=0}}$, where $\text{CASES}_{\text{PROPDEGREE}}$ and $\text{CASES}_{k=0}$ are the number of new MRSA cases of PROPDEGREE and $k = 0$ respectively). Lower γ_{CASES} is better.

- $\gamma_{\textsc{Loads}}$: Ratio of non-patient (HCWs and location) pathogen loads against $k = 0$. (e.g. $\gamma_{\textsc{Loads}}$ for PROPDEGREE is $\frac{\textsc{Loads}_{\textsc{PropDegree}}}{\textsc{Loads}_{k=0}}$, where $\textsc{Loads}_{\textsc{PropDegree}}$ and $\textsc{Loads}_{k=0}$ are the sum of HCWs pathogen loads and location pathogen loads of PROPDEGREE and $k = 0$ respectively). Lower $\gamma_{\textsc{Cases}}$ is better.

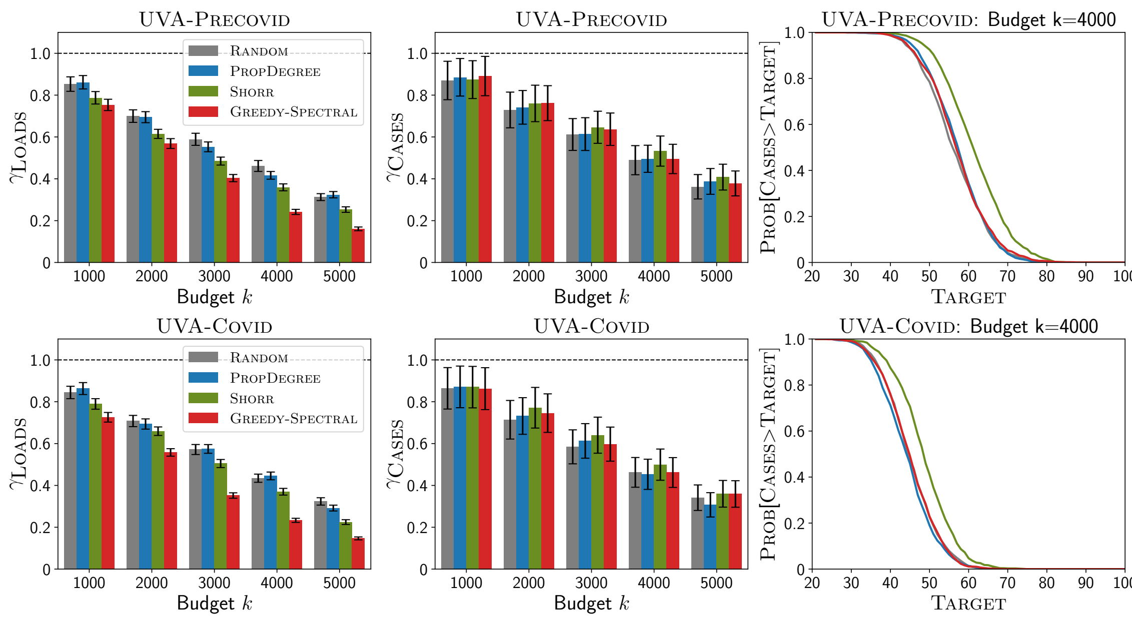
- $\textsc{Prob}[\textsc{Cases} > \textsc{Target}]$: Probability that the number of new MRSA cases is larger than TARGET. (e.g. $\textsc{Prob}[\textsc{Cases} > 500]$ is the probability that the number of new MRSA cases is larger than 500). Lower $\textsc{Prob}[\textsc{Cases} > \textsc{Target}]$ is better.

4.3.2 Performance for Clinical Problems

High Precaution and Cleaning Scenario

We first use the parameterization by calibrating 2-MODE-SIS model to the weekly number of new tested positive MRSA cases collected from EHRs data for experiments. This scenario corresponds to the normal level of precaution and cleaning practices in healthcare settings [155], as in UVA, to avoid the outbreak of HAI including MRSA.

Figure 4.5 shows how GREEDY-SPECTRAL performs in minimizing non-patient pathogen loads and the number of new MRSA cases compared with other baselines: (1) RANDOM will randomly pick k patients, (2) PROPDEGREE will pick k patients with the probability of picking patients i proportional to its degree, (3) SHORR will pick k patients with the highest so-called Shorr scores [154]. Note that SHORR was developed to assess the clinical risk of an admitted patient acquiring MRSA, better performance compared with SHORR means GREEDY-SPECTRAL is a more effective contact precaution strategy. As shown in Figure 4.5(a), GREEDY-SPECTRAL is more effective in reducing the amount of non-patient pathogen loads than other baselines. Here, $\gamma_{\textsc{Loads}}$ is defined as the ratio of non-patient (HCWs and location) pathogen loads against $k = 0$ situation. For example, when $k = 5000$, $\gamma_{\textsc{Loads}}$ for GREEDY-SPECTRAL is $\frac{\textsc{Loads}_{k=5000,\textsc{Greedy-Spectral}}}{\textsc{Loads}_{k=0}}$, where $\textsc{Loads}_{k=5000,\textsc{Greedy-Spectral}}$ and $\textsc{Loads}_{k=0}$ are the sum of HCWs pathogen loads and lo-

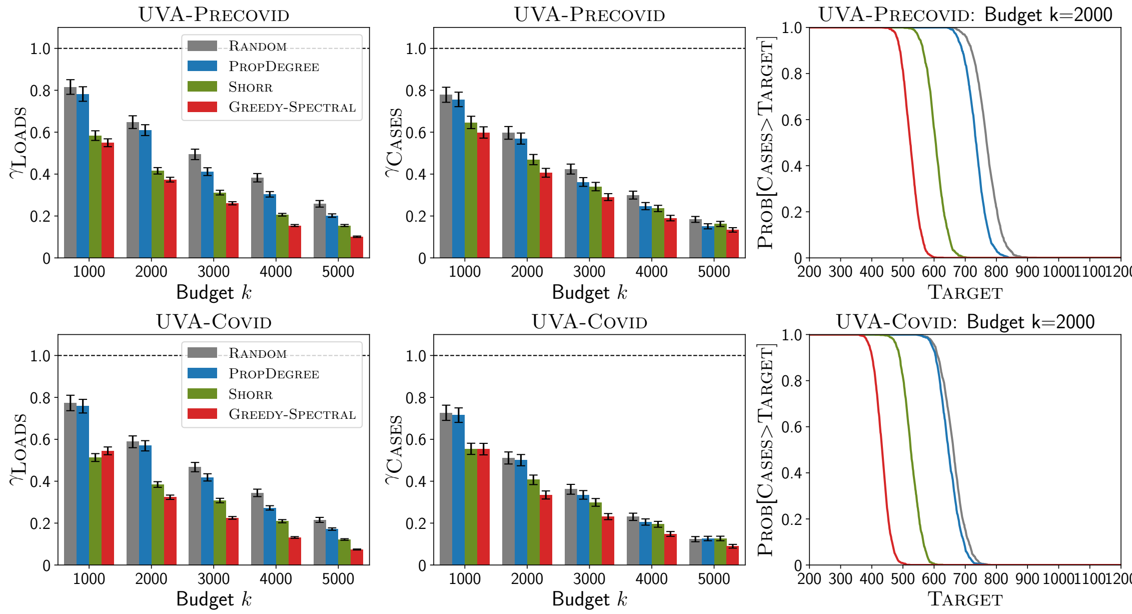


(a) γ_{LOADS} (b) γ_{CASES} (c) PROB[CASES>TARGET]

Figure 4.5: **GREEDY-SPECTRAL achieves lower non-patient pathogen loads but similar number of new MRSA cases than other baselines under high precaution and cleaning scenario for isolation problem** (IP)**.** Top is for UVA-PRECOVID dataset, down is for UVA-COVID dataset. (a) Ratio of non-patient pathogen loads γ_{LOADS} (y-axis) with varying budget k (x-axis). Lower is better. The black bars show the standard error. (b) Ratio of new MRSA cases γ_{CASES} (y-axis) with varying budget k (x-axis). Lower is better. (c) PROB[CASES $>$ TARGET] (y-axis) with varying TARGET (x-axis) for budget $k = 4000$. Lower is better.

cation pathogen loads when selecting 5000 patients for contact precaution using GREEDY-SPECTRAL, and when no patients are selected ($k = 0$) respectively. In Figure 4.5(a), when $k = 5000$, γ_{LOADS} for GREEDY-SPECTRAL is 9.2% (7.8%) lower than for SHORR on UVA-PRECOVID (UVA-COVID), this indicates that GREEDY-SPECTRAL is more effective in reducing the non-patient pathogen loads than other baselines (similar conclusion can also be drawn for other budget k). However, in Figure 4.5(b), we observe that our algorithm still gives a similar ratio of new MRSA cases γ_{CASES}, which is defined similarly as the ratio of the number of new MRSA cases against $k = 0$ situation. In Figure 4.5(c), it also gives similar probability that the number of new MRSA cases is larger than TARGET, PROB[CASES $>$ TARGET] for varying TARGET (x-axis) as other baselines.

Here, lower loads achieved by GREEDY-SPECTRAL are not leading to fewer cases,



(a) γ_{LOADS} (b) γ_{CASES} (c) PROB[CASES>TARGET]

Figure 4.6: **GREEDY-SPECTRAL achieves lower non-patient pathogen loads and number of new MRSA cases than other baselines under relaxed precaution and cleaning scenario for isolation problem** (IP). Top is for UVA-PRECOVID dataset, down is for UVA-COVID dataset. (a) Ratio of non-patient pathogen loads γ_{LOADS} (y-axis) with varying budget k (x-axis). (b) Ratio of new MRSA cases γ_{CASES} (y-axis) with varying budget k (x-axis). (c) PROB[CASES $>$ TARGET] (y-axis) with varying TARGET (x-axis) for budget $k = 2000$.

which is likely because of the isolation practices currently in use.

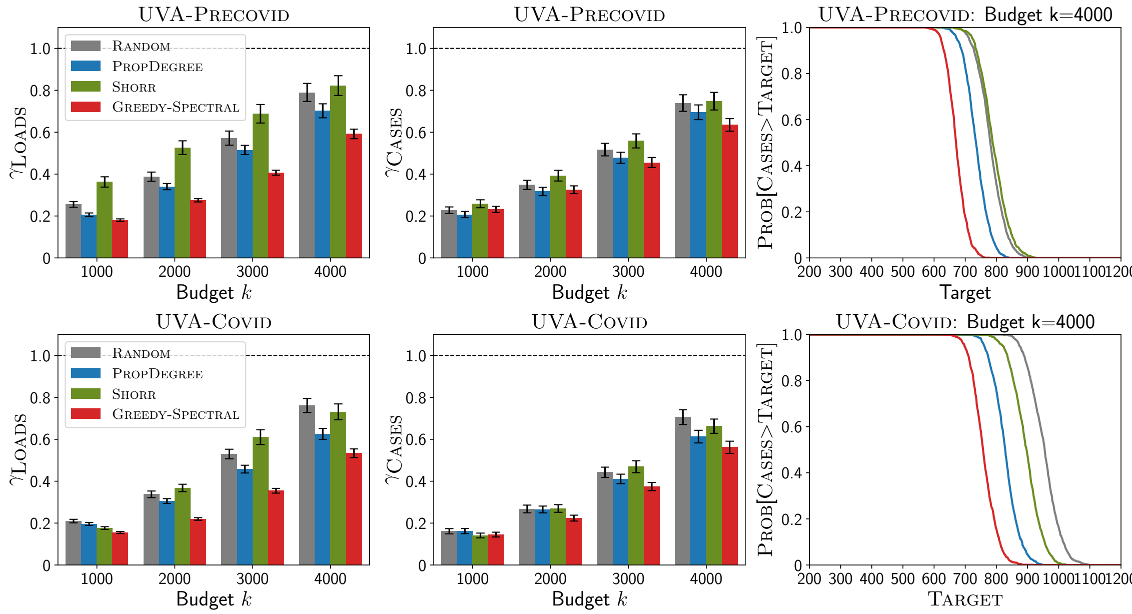
Relaxed Precaution and Cleaning Scenario

Next, we investigate how GREEDY-SPECTRAL and other baselines perform in settings where precautions and cleaning practices are relaxed. Hospitals all over the world have already faced shortages induced by the COVID-19 pandemic, including personal protective equipment, nasal swabs for testing, laboratory diagnostic capacity, and medical personnel, e.g., [156, 157]. Further, shortages in nursing staff have predated the COVID-19 pandemic, and is known to be associated with increased risk of HAI [158]. In the event of such shortages, it is hard to maintain the same level of precautions and cleaning policies, and the number of cases is likely to increase. Optimizing interventions becomes even more important in this setting. In this work, we use the pathogen shedding rate α in the 2-MODE-SIS model to capture the influence of relaxed cleaning practices, and we use α leading to different numbers of cases when $k = 0$ to capture different levels of relaxation of cleaning practices.

We first focus on isolation problem (IP) under relaxed precaution and cleaning scenario. Here, we demonstrate our GREEDY-SPECTRAL algorithm and other baselines by varying k but fixing α such that the number of cases is 1000 when $k = 0$.

Figure 4.6 shows the results with varying k. As shown in Figure 4.6, GREEDY-SPECTRAL achieves lower non-patient pathogen loads and number of new MRSA cases than other baselines. In Figure 4.6(a), it achieves lower γ_{LOADS}. For example, when $k = 2000$, it leads to 4.2% (6.1%) lower loads than SHORR, and 23.6% (24.5%) lower loads than PROPDEGREE on UVA-PRECOVID (UVA-COVID). Furthermore, our algorithm also leads to lower γ_{CASES} as shown in Figure 4.6(b). When $k = 2000$, it achieves 6.3% (7.2%) lower cases than SHORR, and 16.3% (16.6%) lower cases than PROPDEGREE on UVA-PRECOVID (UVA-COVID). This indicates that our algorithm leads to around 60-70 fewer MRSA cases than SHORR and 160-170 fewer cases than PROPDEGREE (since the number of cases is 1000 when $k = 0$). In Figure 4.6(c), we show how PROB[CASES $>$ TARGET] changes with varying TARGET. Here, GREEDY-SPECTRAL always has lowest PROB[CASES $>$ TARGET]. For example, for UVA-PRECOVID (top row) and TARGET $=$ 600, PROB[CASES $>$ TARGET] for our algorithm is 0, indicating that it can always lead to less than 600 cases. However, SHORR has 56.1% probability of having more than 600 cases, and PROPDEGREE and RANDOM have 100% probability of having more than 600 cases. This demonstrates the effectiveness of GREEDY-SPECTRAL algorithm in leading to fewer cases.

We repeat the experiments shown for isolation problem under relaxed precaution and cleaning scenario above to clearance problem. Similarly, we demonstrate our GREEDY-SPECTRAL algorithm and other baselines by varying k but fixing α such that the number



(a) γ_{LOADS} (b) γ_{CASES} (c) PROB[CASES>TARGET]

Figure 4.7: **GREEDY-SPECTRAL achieves lower non-patient pathogen loads and number of new MRSA cases than other baselines under relaxed precaution and cleaning scenario for clearance problem** (CP). Top is for UVA-PRECOVID dataset, down is for UVA-COVID dataset. (a) Ratio of non-patient pathogen loads γ_{LOADS} (y-axis) with varying releasing budget k (x-axis). (b) Ratio of new MRSA cases γ_{CASES} (y-axis) with varying releasing budget k (x-axis). (c) PROB[CASES $>$ TARGET] (y-axis) with varying TARGET (x-axis) for releasing budget $k = 4000$.

of cases is 1000 when $k = 0$.

Figure 4.7 shows the results with varying k. As shown in Figure 4.7, GREEDY-SPECTRAL achieves lower non-patient pathogen loads and number of new MRSA cases than other baselines for the clearance problem. In Figure 4.7(a), it achieves lower non-patient pathogen loads. Furthermore, it also leads to a lower number of new MRSA cases as shown in Figure 4.7(b). For example, when $k = 4000$ (i.e. releasing 4000 patients) for UVA-COVID (bottom row), our algorithm leads to 8.5% lower loads and 6.9% lower γ_{CASES} than other baselines. Besides, it also has lowest PROB[CASES $>$ TARGET] in Figure 4.7(c). For example for UVA-PRECOVID (top row), when releasing $k = 4000$ patients and TARGET $= 700$, GREEDY-SPECTRAL has only 12.3% probability of having more than 700 cases. Instead, other baselines have 100% probability of having more than 700 cases.

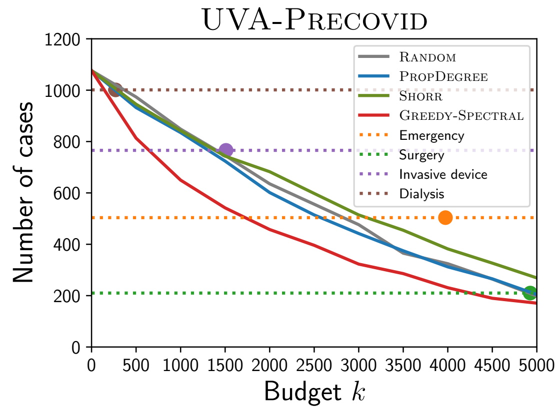


Figure 4.8: **Comparison against clinical heuristic precaution policies.** The x-axis is budget k, the y-axis is the number of cases. The horizontal dashed lines represent the simulated number of MRSA cases with the these policies, and scattered dots correspond to the number of patients isolated by these policies.

Comparison with Clinical Heuristic Precaution Policies

Many clinical heuristic precaution policies are also used in practice [159, 160]. Usually, these policies tend to pick patients with high MRSA risks. For example, patients with surgery history or invasive device use may be considered as having high risk and may be picked for precaution. We next leverage EHRs to implement these clinical heuristic policies and compare them with our GREEDY-SPECTRAL algorithm. The EHRs of patients in the UVA dataset include information like surgery history or underlying diseases (for brevity, we call them "features") for each patient. Specifically, we compare with the following four heuristic policies: isolate all emergency patients, isolate all patients with surgery history within 90 days, isolate all patients with invasive device use history within 90 days, and isolate all patients with dialysis history within 90 days. The four features used in these policies are all considered to be highly risky [159, 160, 154]. As shown in Figure 4.8 for UVA-PRECOVID, GREEDY-SPECTRAL can also suppress MRSA outbreak better than these clinical heuristic precaution policies. Here, horizontal dashed lines represent the simulated number of new cases for these four policies. Scattered dots correspond to the number of patients isolated by each policy. We can see the line for our algorithm is below

Table 4.4: Fraction of patients with different features

Features	population	PROPDEGREE	SHORR	GREEDY-SPECTRAL
Surgery (within 90 days)	0.601	0.614	0.696	0.750
Surgery (90+ days)	0.377	0.434	0.572	0.572
Invasive Device (within 90 days)	0.233	0.264	0.472	0.470
Invasive Device (90+ days)	0.171	0.180	0.356	0.382
Dialysis (within 90 days)	0.041	0.050	0.184	0.136
Dialysis (90+ days)	0.031	0.042	0.160	0.096
ICU	0.281	0.326	0.508	0.402

all scattered dots. This indicates that when isolating the same number of patients, GREEDY-SPECTRAL leads to fewer MRSA cases than these policies.

4.3.3 Case Study: Analysis on Isolated Patients

We also investigate the patients picked by our GREEDY-SPECTRAL algorithm from a clinical perspective. In Table 4.4, we use PROPDEGREE, SHORR, and GREEDY-SPECTRAL to pick 500 patients for precaution for UVA-PRECOVID, and then show the fraction of patients with highly risky features [159, 160] (note that one patient may have multiple features and can be counted multiple times). We also list the fraction of patients with these features in population for comparison. As shown in Table 4.4, our algorithm GREEDY-SPECTRAL picks meaningful patients with these features for precaution. For example, compared with PROPDEGREE and fraction in population, it picks many more patients with invasive device use history and dialysis history. Compared with SHORR, GREEDY-SPECTRAL highlights patients having surgery within 90 days more while emphasizing dialysis patients less. Additionally, GREEDY-SPECTRAL also captures the structural information of the heterogeneous contact networks well since the average degree of the patients picked by it is larger than other baselines. This explains why it performs better than SHORR in suppressing MRSA outbreaks. *Note that our algorithm uses only network information, and not patient level feature information, but it still identifies the correct patients at high risk and performs better in minimizing the MRSA spread as shown above*.

In Table 4.5, we also investigate the differences in the type of patients picked in the

Table 4.5: Fraction of patients with different features for GREEDY-SPECTRAL on UVA-PRECOVID and UVA-COVID dataset.

Notations	UVA-PRECOVID	UVA-COVID
Surgery (within 90 days)	0.750	0.752
Surgery (90+ days)	0.572	0.6
Invasive Device (within 90 days)	0.470	0.532
Invasive Device (90+ days)	0.382	0.342
Dialysis (within 90 days)	0.136	0.164
Dialysis (90+ days)	0.096	0.086
ICU	0.402	0.48

UVA-PRECOVID and UVA-COVID period. Here, we list the fraction of patients with highly risky features picked by GREEDY-SPECTRAL in both periods. As shown in Table 4.5, our algorithm is picking 13.2% more patients with invasive device use history within 90 days, 20.6% more patients with dialysis history within 90 days and 19.4% more patients in ICU in UVA-COVID than UVA-PRECOVID. For example, the increasing number of patients picked in ICU by our algorithm can be explained by the fact that ICU is getting more crowded. According to the EHRs data, there are more contacts in the UVA-COVID period than in the UVA-PRECOVID period (average degree of patients in ICU: 115.33 (UVA-PRECOVID) va. 125.24 (UVA-COVID)). Additionally, the average degree of the patients picked by our algorithm for UVA-COVID period is also larger than UVA-PRECOVID.

4.4 Discussion

We develop a spectral characterization for 2-MODE-SIS [68] for modeling the spread of HAIs, such as MRSA pathogens, which captures the long-term dynamics of pathogen spread by taking into account both the contact network structure and disease parameters. This model has been proposed as an alternative to the standard SIS models, but has not been studied rigorously before. We also study two important clinical problems arising from control of HAIs, and show how these can be formalized using our spectral characterization. We design GREEDY-SPECTRAL, an algorithm for solving these clinical problems, which greedily and iteratively finds and adds a patient to isolate (or relase) until reaching the budget.

Overall, our results demonstrate that our spectral characterization and GREEDY-SPECTRAL algorithm hold sufficient potential for minimizing the spread of MRSA. In experiments, we compare GREEDY-SPECTRAL with several baselines including SHORR, which picks patients for contact precautions using their features collected from Electronic Health Records. For the high precaution and cleaning scenario, which corresponds to the normal level of pathogens and cleaning practices in healthcare settings, our simulations suggest that the GREEDY-SPECTRAL algorithm outperforms the SHORR by achieving lower non-patient pathogen loads. For the relaxed precaution and cleaning scenario, which captures the shortage of resources (e.g., personal protective equipment, laboratory diagnostic capacity, and medical personnel in the COVID-19 pandemic [156, 157]), our simulations indicate that GREEDY-SPECTRAL algorithm not only reduces pathogen loads but also leads to fewer MRSA cases compared with SHORR. Additionally, our algorithm uses only the network information and not the patient-level feature information. However, it still identifies the correct patients at high risk and performs better in minimizing the MRSA spread as shown above.

Our GREEDY-SPECTRAL algorithm is likely to be helpful in controlling MRSA and other healthcare-associated infections. As demonstrated in our simulations, our algorithm uses only the network information rather than patient-level features, yet it successfully identifies high-risk patients and outperforms SHORR and other baselines in minimizing MRSA spread. In practice, the GREEDY-SPECTRAL algorithm can be implemented by constructing contact networks using EHR data such as inpatient data, doctors' notes, and MedAdmin data, which are as similarly available as patient-level features. Some research also suggests that wearable RFID sensors can be used to construct networks [161, 162]. With the contact networks, people can then run our GREEDY-SPECTRAL algorithm to decide the most suitable patients to put under or released from precaution (corresponding to the isolation problem and clearance probelm respectively). We expect our approach will assist epidemiologists and clinicians in identifying appropriate patients for contact precautions, ultimately aiding in the control of MRSA transmission. Our spectral characterization and GREEDY-SPECTRAL algorithm can also be extended to other diseases and epidemiological models.

One of the limitations of this work is that the 2-MODE-SIS model we use involves only two states: susceptible and infected. Real-world MRSA infections, however, manifest in various forms (e.g., bloodstream infections and pneumonia) [163], while the 2-MODE-SIS model simplifies them into a single infected state. In addition, the 2-MODE-SIS model assumes that any susceptible patient can be colonized instead of explicitly incorporating a colonization state in it itself. One potential extension of this work could extend our spectral characterization analysis to more complex models that could distinguish different kinds of MRSA infections and colonization. Our approach can also be adapted for other healthcare-associated infections, such as *C. diff*, by substituting the 2-MODE-SIS model with other models. Another limitation is that we assume that patients selected for contact precautions remain isolated throughout the entire period. However, in actual clinical practice, patients may be released from contact precautions when they recover or are discharged from the hospital. Future work could consider the possibility of isolating or releasing patients at any time step to gain more flexibility. Although contact networks and other parts of EHR data are very sensitive datasets, the privacy concerns can be mitigated by running our GREEDY-SPECTRAL algorithm on anonymous contact networks to estimate the impact of the optimal solution, and use characteristics of the solution in defining actual strategies.

CHAPTER 5

MODELING RELAXED POLICIES FOR DISCONTINUATION OF METHICILLIN RESISTANT *STAPHYLOCOCCUS AUREUS* CONTACT PRECAUTIONS

5.1 Introduction

Methicillin-resistant *Staphylococcus aureus* (MRSA) is a leading source of healthcare-associated infections (HAIs) in the United States. In hospitals, contact precautions are typically applied to patients with known MRSA status or who test positive on screening tests to control in-hospital transmission [8, 48, 150]. Nares polymerase chain reaction (PCR) tests are recommended for guiding decisions on discontinuing MRSA precautions [51], also known as releasing policies. While one to three consecutive negative test results from the nares are common thresholds for discontinuation of contact precautions, these releasing policies are often arbitrary, lacking substantial evidence and neglecting patient-specific factors. For instance, the University of Virginia (UVA) hospital employs a "three-negative polic" where contact precautions are discontinued after three consecutive negative test results. However, whether this policy is optimal in terms of balancing costs and controlling MRSA transmission is unknown [164].

If MRSA surveillance testing were always accurate, then a single-negative policy would be optimal. However, several factors, including sensitivity of the test, low bacterial load at time of collection, and sampling error, abrogates the accuracy of testing. Repeated testing improves the sensitivity and thus can extend precautions in patients that may otherwise be assumed clear; however requiring gowns and gloves and other resources to isolate MRSA patients is expensive, with an average estimated cost of more than $400 per day per patient [49]. In addition, although the extent of side-effects of contact precautions remains uncertain [5, 50], there are potential harms due to extended contact precautions [51, 37], including reductions in provider visits to patient rooms [4], which can increase risk of depression7. Thus, releasing policies must balance the potential harms from contact precautions against the potential prevention of transmission.

To-date there are few systematic analyses of the optimal length of releasing policies. Prior work on evaluating releasing policies [52, 53] have focused on individual-level risk estimation using EHR data. Here we examine the trade-off between increased transmission (which would raise treatment costs) and reduced costs from fewer precaution days at the hospital level. We use a mathematical model to simulate outcomes, since no empirical data exists.

5.2 Methods

5.2.1 2-MODE-PRECAUTION Model

In this work, we propose a new dynamical system model, referred to as the 2-MODE-PRECAUTION model, to represent MRSA transmission pathways in the UVA hospital. This model extends a prior model, referred to as the 2-MODE-SIS epidemic model [68, 30], which was proposed for capturing MRSA transmission dynamics, but does not account for contact precautions. Our model is comprised of three types of entities: patients, healthcare workers (HCWs), and locations. A key aspect of the model is the contact network, which specifies the contacts between patients and HCWs and the locations where these happen. This information is not directly available, and is inferred from EHR data, through tables such as medicine administration. Each patient in the hospital is in one of the states: S (Susceptible), C (Carriage), S_{cp} (Susceptible, under contact precaution) and C_{cp} (Carriage, under contact precaution). Patients outside the hospital can be either in S or C states. This is shown in the state transition diagram of the 2-MODE-PRECAUTION model in Figure 5.1. HCWs and locations are only associated with pathogen load (so they can be considered as in the carriage state).

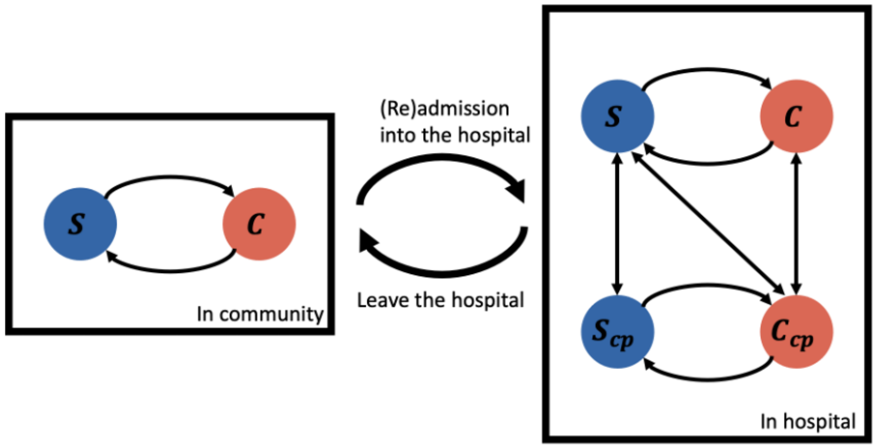


Figure 5.1: **The diagram of states for patients in the 2-MODE-PRECAUTION model.** There are six states in the 2-MODE-PRECAUTION model: four for patients in the hospital (S - Susceptible, out of contact precaution, C - Carriage, out of contact precaution, S_{cp} - Susceptible, under contact precaution, and C_{cp}- Carriage, under contact precaution) and two (S and C) for patients outside the hospital, or in the community. Each patient, HCW and location is associated with a pathogen load, which is transferred through contacts between people and locations. A patient is infected with a probability that depends on the load they have accumulated. Load on all entities decays at a steady rate.

Following the approach of previous work [68], the 2-MODE-PRECAUTION model has three parts: transfer of pathogen load between entities, patient infection (based on their pathogen load), decay in load. We assumed that a contact between entities i and j results in transfer of pathogen load between them and averages the load between both of them. The entire set of pathogen transfer events can be captured by application of a pathogen transfer matrix constructed from the network. A patient moves from a susceptible state (S or S_{cp}) to a carriage state (C or C_{cp}) with a probability that is determined by a dose-response function— this probability increases with the patient's pathogen load. Patients in the carriage state are assumed to shed at a higher rate. The shedding continues until the patient recovers.

A patient in state S or C moves to state S_{cp} or C_{cp} respectively, if they are in the hospital, and are put under contact precautions. For every patient under contact precaution (i.e., in S_{cp} and C_{cp}), contact precautions reduce the pathogen load transferred via contacts by an effectiveness factor ϕ_{edge}. Note that there is no general policy for testing in the UVA hospital. Instead, the tests are performed based on the clinician's request. Therefore, in this work, we assume that the testing rates/parameters in the precaution model are captured in the transition probabilities between different states, which are calibrated from MRSA infection data. For instance, $p_{C \to C_{cp}}$ represents the transition probability from state C to state C_{cp}, corresponding to patients who tested positive for MRSA and were placed under contact precautions. Similarly, $p_{C_{cp} \to C}$ denotes the transition probability from state C_{cp} to state C, corresponding to patients who are wrongly released due to false-negative MRSA tests. Such assumptions are reasonable since the carriage state in the 2-MODE-SIS model can represent both MRSA colonized and infected patients. There may exist undetected colonized MRSA cases due to undertesting, false negatives, and/or MRSA infections occurring after hospital discharge. We also include parameters to account for importation of cases, following existing research works [10, 165].

5.2.2 Parameter Adjustment Method

The most clinically significant parameter changes in the 2-Mode-Precaution model for estimating outcomes for different releasing polices are $p_{C_{cp} \to C}$ (the transition probability from C_{cp} to C) and $p_{S_{cp} \to S}$ (transition probability from S_{cp} to S). We denote the different policies as $p^{i-neg}_{C_{cp} \to C}$, where i is the number of tests in the policy. Other non-precaution-based infection controls and prevention measures are assumed to remain constant (e.g., terminal room disinfection, standard hand hygiene practices, antimicrobial use). Next, we will demonstrate how to utilize the MRSA PCR test records from EHR data to adjust from from $p^{3-neg}_{C_{cp} \to C}$ to estimate $p^{2-neg}_{C_{cp} \to C}$. First, let's focus on $p_{C_{cp} \to C}$. Under the 3-negative policy, only patients with three consecutive negative tests will be released. However, under the 2-negative policy, patients with the first two tests negative will also be released without a third test. We use n_0 to represent the number of records with three consecutive negative tests in EHR data (i.e., negative-negative-negative), and n_1 to represent the number of records with two negative test results followed by one positive result (i.e., negative-negative-positive). With the 2-negative policy, n_1 more patients will be released. Hence, $p_{C_{cp}\to C}$ for the 2-negative policy will be $p^{2-neg}_{C_{cp}\to C} \approx \frac{n_1+n_0}{n_0} p^{3-neg}_{C_{cp}\to C}$. Intuitively, the relaxed policy will result in a higher $p^{2-neg}_{C_{cp}\to C}$ than $p^{3-neg}_{C_{cp}\to C}$, since MRSA infectious cases are released with a higher probability due to one fewer test. To extract results of MRSA tests, we parse each patient's EHR and extract MRSA PCR nares and culture test results, constructing our dataset using the date the test was taken and the result of the test. As described above, we exclude negative culture tests, as a negative culture test does not preclude the presence of MRSA and is thus not sufficient to clear contact precautions at the UVA hospital. Next, let's focus on $p_{S_{cp}\to S}$. Under the 3-negative policy, patients must wait for three consecutive negative tests to be released. However, under the 2-negative policy, patients only need to wait for two consecutive negative tests, which indicates a shorter waiting time and hence a larger transfer probability. We use d_3 to capture the average number of days to obtain three consecutive negative tests, and d_2 to capture the average days to obtain the first two consecutive negative results after the initial positive test. By assuming that the average days follow a geometric distribution parameterized by $p_{S_{cp}\to S}$, then $p^{3-neg}_{S_{cp}\to S}$ should be proportional to $\frac{1}{d_3}$, and $p^{2-neg}_{S_{cp}\to S}$ should be proportional to $\frac{1}{d_2}$. Therefore, we have $p^{2-neg}_{S_{cp}\to S} \approx \frac{d_3}{d_2} p^{3-neg}_{S_{cp}\to S}$. Intuitively, the relaxed 2-negative policy will result in a higher $p^{2-neg}_{S_{cp}\to S}$ than $p^{3-neg}_{S_{cp}\to S}$, since patients are released out of precaution faster with a higher probability.

For the calculation of d_3, we analyze the number of instances in patient EHRs where an individual tests positive for MRSA and then receives three consecutive negative MRSA tests as their next three tests on record. We then count the number of days between the first positive test and the third negative test, excluding days when the patient was not present in the hospital (as calculated by their admit and discharge days). This value is then averaged across all instances of a positive test followed by three negative tests. Similarly, for d_2, we use the same subsets of patients that have a positive test followed by three negative tests in order to perform a consistent comparison. However, we then take the number of days

Table 5.1: Value of n_0, n_1, n_2 and d_3, d_2, d_1

Year	n_0	n_1	n_2	d_3	d_2	d_1
2015	81	12	30	155.5	154.5	153.5
2016	138	9	21	134.25	97.75	74.75
2017	121	9	15	101.75	86	21.75
2018	122	14	28	15.66	15	4.33
2019	278	41	78	169.5	140	115.625

between the positive test and the second negative test, again discounting time when a patient has left the hospital, and average this over the entire data set. Due to this methodology, we will always have $d_2 < d_3$, as hospital policy requires at least 24 hours between tests.

In summary, we can use the parameters calibrated for the current 3-negative policy in the UVA hospital to estimate parameters for the 2-negative policy with the following approach: First, we calibrate the 2-Mode-Precaution model to the number of detected cases in the UVA hospital, obtaining the parameters for the 3-negative policy. Next, we apply the parameter adjustment method to estimate the parameters for the 2-negative policy. We then run simulations with the adjusted parameters to estimate the 2-negative policy scenario. Similarly, we can also adjust the calibrated 3-negative parameters to estimate the 1-negative policy scenario. We also list the values of n_0, n_1, n_2 and d_3, d_2, d_1 in Table 5.1.

5.2.3 Costs for Different Policies

To quantitatively examine the tradeoff between the number of additional MRSA cases and the number of precaution days saved under the relaxed policies, we calculate the total cost for different policies. We use costs reported in a retrospective study in a Canadian hospital from 2005 to 2010 [49], which provides the estimated treatment (e.g., laboratory cost, infection control time, housekeeping) and precaution (e.g., private room and length of stay cost per day) cost for colonized patients and infected patients (values already adjusted to 2010 Canadian Dollars (CAD) in that study). As the study mentions, the majority (88.9%) of cases are colonized [132]. The treatment costs (which are different for these two patient groups) are derived from multiplying the number of MRSA patients by the treatment cost per patient. Finally, we convert their values to 2010 US dollars [166], and then inflation adjust to 2023 [167] for a final estimated cost of treatment of $435.89 USD in 2023 for colonized patients and $459.13 USD in 2023 for infected patients.

The cost of precautions was similarly calculated by multiplying the number of precaution days (the total number of patients under precaution each day) by the precaution cost estimates from the literature [49]. After conversion, the estimated cost of precaution is $403.58 USD in 2023 for colonized patients and $2,110.90 USD in 2023 for infected patients. Note that here we only focus on the economic costs incurred by the UVA hospital, patient costs, the downstream influence and cost to the community or other healthcare facilities are not included.

5.2.4 Dataset

We construct heterogeneous contact networks using EHR data collected at the UVA hospital. These records combine inpatient data, doctor's notes, and medication administration data, which document the location and time of interactions between patients and healthcare workers (HCWs). We then aggregate this information into daily networks, connecting two nodes (patients, HCWs, locations) in the contact networks if they have at least one contact on a given day. From January 1, 2015, to December 31, 2019, the constructed networks included 41,216 patients, 14,392 healthcare workers, and 685 locations across all departments within the hospital. The weekly number of incident MRSA cases was obtained using both MRSA positive cultures and PCR nares surveillance swabs. For removal of precautions, we only considered negative PCR results, as negative culture tests are not used to remove individuals from contact precautions at UVA. From January 1, 2015, to December 31, 2019, the UVA hospital performed 22,825 PCR tests on 15,806 patients based on clinician orders. These tests resulted in 2,481 positive and 20,344 negative outcomes, with 1,800 unique patients having at least one positive test. Although not all patients in the UVA

Table 5.2: Number of patients, HCWs, and locations across 2015-2019 in UVA dataset

	2015	2016	2017	2018	2019	Total (unique)
Patients	11,367	11,339	11,008	10,658	10,087	41,216
Healthcare workers	3,967	3,811	8,905	8,017	7,573	14,392
Locations	574	587	587	533	606	685

hospital were tested, potential undetected or colonized patients are accounted for in the carriage state of our 2-Mode-Precaution model.

5.2.5 Calibration and Setup

As described in section 5.1, the UVA hospital employs a "three-negative policy" where contact precautions are discontinued after three consecutive negative test results. Therefore, to infer the parameters for the three-negative policy, we use the Ensemble Adjustment Kalman Filter (EAKF) [127], which has been widely used for epidemiological models on various healthcare-associated infections, including MRSA, and has demonstrated good performance [165, 168, 20]. Specifically, we calibrate the number of cases transferred from C to C_{cp} (i.e., patients who are tested based on clinician orders and turned out to be positive) to the incident MRSA case numbers. Calibration was restricted to known incident hospital MRSA cases as incident cases outside the hospital are not recorded as part of UVA EHR data.

Here, we demonstrate the effectiveness of the calibration procedure from two perspectives: First, we conduct a synthetic experiment to demonstrate our ability to "recover" synthetic parameters by calibrating on the simulated curves generated using these parameters. Second, we illustrate that our simulated curves, derived from the calibrated parameters, align closely with the real-world number of detected MRSA cases.

Here, we set up 5 different "scenarios" for synthetic experiments. In each scenario, we set "ground truth" parameters and run simulations with these ground-truth parameters to generate a synthetic detected MRSA cases curves. Next, we use each synthetic MRSA

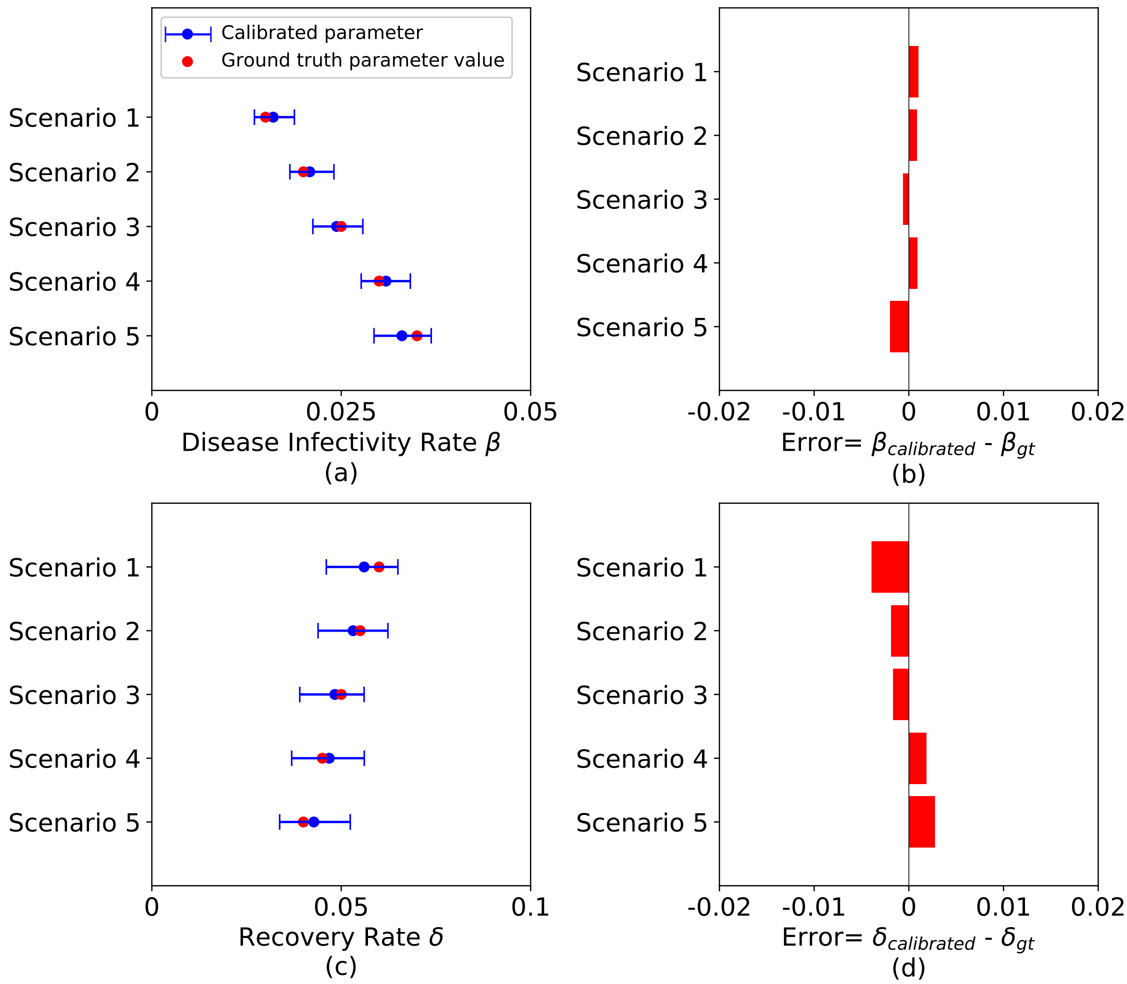


Figure 5.2: **The calibration procedure effectively "recovers" the synthetic parameters by calibrating on the simulated curve generated using these synthetic parameters.** Left: red dots represent the ground-truth values of synthetic parameters, while blue dots and error bars represent the mean value and 95% confidence interval of calibrated parameters. We can see our calibrated parameters closely align with their ground-truth parameters. Right: the error between our estimated parameters and ground-truth parameters.

case curve for calibration and obtain the calibrated parameters. If the calibrated parameters closely align with the ground truth parameters we initially set (i.e., we can "recover" these parameters through calibration), we conclude that the calibration process is effective.

We show our results in Figure 5.2. As shown in the figure, we compare the ground truth parameter values (red dots) with the calibrated parameter values (blue dots and error bars, representing mean value and 95% confidence interval) on the left. Our calibrated parameters closely align with the ground truth parameters (in fact, the ground truth parameters

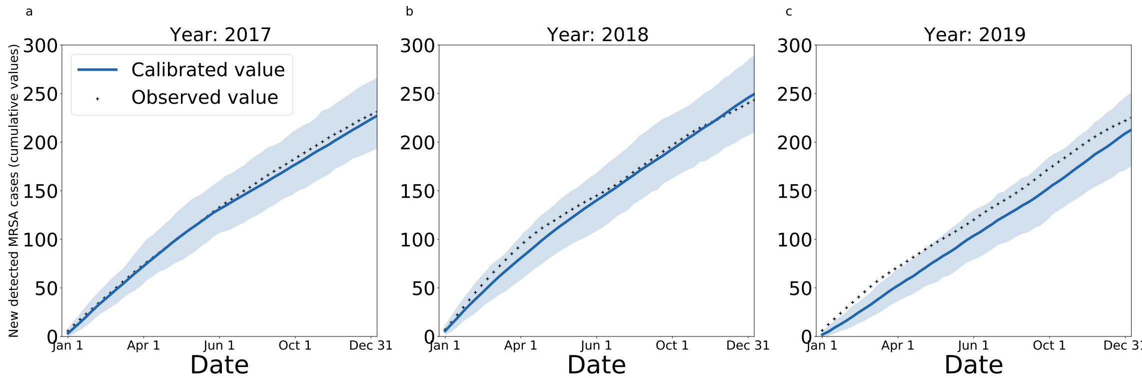


Figure 5.3: **Our calibrated parameters fit the real-world number of detected MRSA cases well.** Black dots represent the cumulative number of detected MRSA cases. The blue curves and shaded regions represent the mean value and 95% confidence interval for the cumulative number of detected MRSA cases based on the calibrated parameters. The x-axis is the date, while the y-axis is the cumulative value for the number of detected MRSA cases. Panels (a), (b), (c) correspond to 2017, 2018, 2019, respectively.

always fall within the 95% confidence interval of the calibrated parameters). We also display the error between calibrated parameters and ground-truth parameters on the right. By selecting multiple values for the ground truth parameters, we show calibration is effective over a broad range of ground truth parameters. It is worth noting that the absolute value of errors remains small. Therefore, we can conclude that our calibration procedure effectively recovers the synthetic parameters across a wide range of parameters. Next, we show that the calibration procedure works well on the number of MRSA cases from EHR data. Here, we use the cumulative values of the number of MRSA cases for calibration to obtain the calibrated parameters. We then simulate with these calibrated parameters to get the simulated curves. If the simulated curves are close to the observed values, we can conclude that the calibration procedure works well on the actual number of MRSA cases.

We show the results in Figure 5.3. Here, black dots are the cumulative number of detected MRSA cases under the 3-negative policy in UVA hospital now. Blue curves and shaded regions are our simulated curves' mean value and 95% confidence interval. Note that the blue curve closely aligns to the blue dots, and most black dots are within the confidence interval of the simulated curves (blue shadow area). Hence, the calibrated parameters

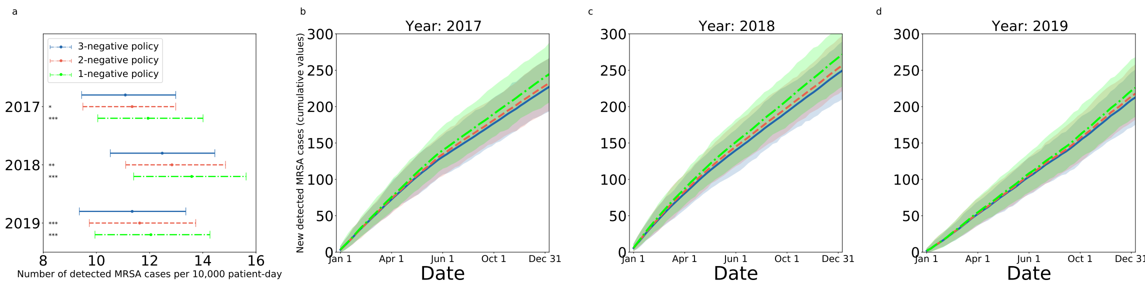


Figure 5.4: **The 2-negative and 1-negative policies lead to more new detected MRSA cases compared to the current 3-negative policy.** (a) Blue, red, and green dots and error bars indicate the mean value and 95% confidence interval for detected MRSA cases per 10,000 patient-day under the 3-negative, 2-negative, and 1-negative policies, respectively. $^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$. (b-d) The blue, red, and green curves and shaded regions represent the mean value and 95% confidence interval for the cumulative number of detected MRSA cases under the 3-negative, 2-negative and 1-negative policies, respectively. Panels (b), (c), (d) correspond to 2017, 2018, 2019, respectively.

fit the real-world number of detected MRSA cases well.

We use the parameter adjustment method to estimate the difference of outcomes between the current three-negative policy and the potential two- and one-negative policies. We ran each simulation 300 times using these adjusted parameters to estimate outcomes, and calculated the costs associated with each policy to compare their effectiveness.

5.3 Results

5.3.1 Number of Detected MRSA Cases

On average, the two-negative policy leads to 2.56% more MRSA cases (95% CI: -13.2% to 19.1%, $p < 0.001$) than the three-negative policy, while the one-negative policy results in 7.59% more MRSA cases (95% CI: -10.1% to 25.8%, $p < 0.001$) over the 2017-2019 period (see Figure 5.4). Despite the large 95% CI owing to the inherent high variance in agent-based model simulations, the statistical 2-sample t-tests reveal that both the 2-negative and 1-negative policies would lead to an increase in the number of detected MRSA cases per 10,000 patient-days, compared to the 3-negative policy.

While removing contact precautions earlier does lead to more cases, the false negative

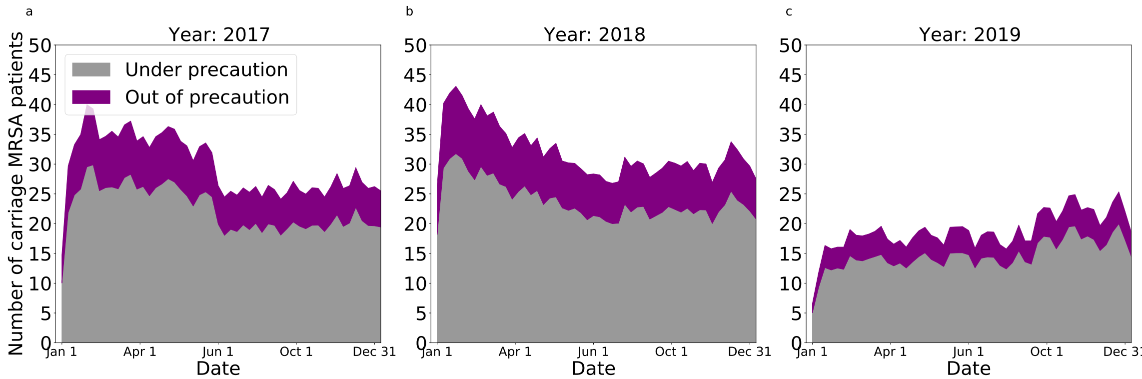


Figure 5.5: **Most MRSA cases are under contact precautions.** The shaded areas represent the weekly number of in-hospital patients in C and C_{cp} states under the 1-negative policy. The gray and purple sections correspond to cases that are under or not under contact precautions, respectively. Panels (a), (b), (c) correspond to 2017, 2018, 2019, respectively.

rate is low, and so the majority of MRSA cases still remain under contact precautions in the hospital, even under the one-negative policy (Figure 5.5). Under the one-negative policy, 75.77% (95% CI: 74.23%-77.05%) of MRSA cases remain under contact precautions.

5.3.2 Shorter Contact Precaution Durations for Two-negative and One-negative Policies

he average contact precaution durations over the 2017-2019 period for the two- and one-negative policies were 5.56 (SDs: 4.05), and 4.75 (SDs: 3.35) days, respectively. These were both shorter than the three-negative policy at 5.88 (SDs: 4.39) days (Figure 5.6). The shorter durations led to lower costs for precautions (Table 5.3). For the two-negative policy, the estimated total cost, $687,946 (95% CI: $562,522-$812,662), was marginally lower than the three-negative policy, $702,823 (95% CI: $577,277-$846,605) with $p < 0.005$ for 2-sample T-test). Meanwhile, the one-negative policy leads to a total cost of $628,452 (95% CI: $513,592-$752,148), which is significantly lower than the three-negative policy ($p < 0.001$ for 2-sample T-test).

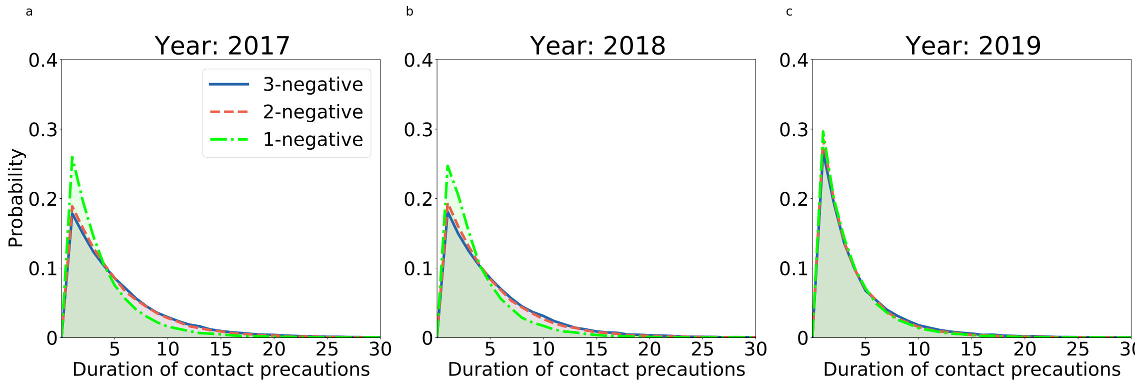


Figure 5.6: **Distribution for contact precaution durations related to MRSA.** The blue, red, and green curves represent the distribution for 3-, 2-, and 1-negative policies, respectively. The x-axis is the contact precaution durations in days, while the y-axis is the probability. Panels (a), (b), (c) correspond to 2017, 2018, 2019, respectively.

5.4 Discussion

In this work, we studied the potential cost effectiveness of changing from the current policy of needing three consecutive negative MRSA surveillance tests to remove precautions, to either a two- or one-negative clearance policy at the UVA hospital. To estimate the outcome, we developed a new 2-Mode-Precaution model and calibrated the parameters to data from the electronic health record (EHR) data. To balance the influence of rising treatment costs due to increased MRSA cases and the reduction in precaution costs resulting from fewer precaution days, we computed the total cost for the three-negative policy and the other two alternative relaxed policies. Our study indicates that a single negative MRSA nares PCR test may provide sufficient evidence to discontinue MRSA contact precautions, and that the one-negative policy may be the most cost-effective option. This research yields three key findings: (i) One-negative and one-negative policies result in relatively few (¡8%) additional MRSA cases when compared to the conventional three-negative policy; (ii) relaxed releasing policies reduce the number of precaution days; and (iii) a one-negative policy leads to significant savings by having substantially lower precaution costs.

A limitation of this work is the assumption that model parameters in the 2-Mode-Precaution model remain constant throughout the entire time period. In reality, these pa-

Table 5.3: Estimated value for the average annual cost for 3-negative, 2-negative, and 1-negative policy for 2017-2019 via simulation (in USD, adjusted to 2023)

	3-negative	2-negative	1-negative
Number of detected MRSA cases	230 (95% CI: 193-269)	236 (95% CI: 200-274)	247 (95% CI: 207-289)
Number of detected MRSA infection cases	26 (95% CI: 21-30)	26 (95% CI: 22-30)	27 (95% CI: 23-32)
Number of detected MRSA colonized cases	204 (95% CI: 172-239)	210 (95% CI: 177-243)	220 (95% CI: 184-257)
Number of precaution days	1,015 (95% CI: 831-1229)	986 (95% CI: 801-1168)	877 (95% CI: 713-1055)
Number of precaution days (for infections)	113 (95% CI: 92-136)	109 (95% CI: 89-130)	97 (95% CI: 79-117)
Number of precaution days (for colonized)	902 (95% CI: 738-1,093)	876 (95% CI: 712-1,038)	779 (95% CI: 634-937)
MRSA infection cases treatment cost	11,710 (95% CI: 9,842-13,685)	12,012 (95% CI: 10,175-13,940)	12,608 (95% CI: 10,540-14,729)
MRSA colonized cases treatment cost	89,042 (95% CI: 74,837-104,055)	91,334 (95% CI: 77,366-105,993)	95,865 (95% CI: 80,140-111,996)
MRSA infection cases precaution cost	237,864 (95% CI: 194,614-287,957)	230,962 (95% CI: 187,655-273,682)	205,433 (95% CI: 167,084-247,092)
MRSA colonized cases precaution cost	364,225 (95% CI: 297,999-440,929)	353,657 (95% CI: 287,343-419,070)	314,565 (95% CI: 255,845-378,354)
Total cost	702,823 (95% CI: 577,277-846,605)	687,946 (95% CI: 562,522-812,662)**	628,452 (95% CI: 513,592-752,148)***

rameters may be time-varying, with seasonal fluctuations potentially causing significant variations in transmission and MRSA importation rates. We also assume that the non-precaution-based infection controls are constant, which is true at the UVA hospital but may not be applicable for other hospitals. Additionally, we only utilize in-hospital MRSA test results to calibrate the 2-Mode-Precaution model due to limited data availability, we also do not account for bias in testing patients for MRSA. We also do not factor costs associated with reduced hospital bed capacity. For example, patients on MRSA isolation require placement into single rooms or cohorts, which effectively reduces bed capacity. Additionally, the EHRs employed to construct contact networks only contain interactions where both patients and healthcare providers are present. As a result, interactions exclusively between healthcare providers, such as two providers in a breakroom simultaneously, are not recorded and modeled. Moreover, although other individuals like visitors, hospital administrators, and janitorial staff may contribute to MRSA transmission, their involvement is not documented in EHRs. Future research could expand the model to accommodate these time-variant parameters and colonization states. Besides, it is still an open problem how the delays between testing and getting results will influence the trade-offs between policies.

Another limitation of this analysis is that it does not account for healthcare worker fatigue; repeatedly donning and doffing gowns and gloves may lead to reduced adherence to contact precautions over time (as demonstrated with the COVID-19 pandemic) [169], thereby potentially reducing the benefits of stricter clearance protocols. As the cost analysis was hospital-centered, we also did not factor potential downstream societal costs associated with increased MRSA-colonized patients in the community as a result of relaxed precaution discontinuation policies. However, people can easily extend from our proposed model to account for them. In this work, we focused on the spread dynamics and economic costs in the UVA hospital only, while the downstream influence of such relaxed contact precaution policy to the community or other care facilities are not included and evaluated. In particular, there is a chance that patients discharged under a one-negative clearance policy could lead to more transmission in other places, such as nursing homes. A mitigating factor in our study is that most of the additional infections are already under precautions, so this risk is reduced. However, additional work is needed to understand these effects.

Part II

Machine Learning for Accurate, Fine-grained Modeling

CHAPTER 6
ACCURATELY ESTIMATING TOTAL COVID-19 INFECTIONS USING INFORMATION THEORY

One of the most significant challenges in the early combat against COVID-19 was the difficulty in estimating the true magnitude of infections. Unreported infections drove up disease spread in numerous regions, made it very hard to accurately estimate the infectivity of the pathogen, therewith hampering our ability to react effectively. Despite the use of surveillance-based methods such as serological studies, identifying the true magnitude is still challenging today. This work proposes an information theoretic approach for accurately estimating the number of total infections. Our approach is built on top of ordinary differential equations (ODE) based models, which are commonly used in epidemiology and for estimating such infections. We show how we can help such models to better compute the number of total infections and identify the parameterization by which we need the fewest bits to describe the observed dynamics of reported infections. Our experiments show that our approach leads to not only substantially better estimates of the number of total infections but also better forecasts of infections than standard model calibration based methods. We additionally show how our learned parameterization helps in modeling more accurate what-if scenarios with non-pharmaceutical interventions. Our results support earlier findings that most COVID-19 infections were unreported and non-pharmaceutical interventions indeed helped to mitigate the spread of the outbreak. Our approach provides a general method for improving epidemic modeling which is applicable broadly.

6.1 Introduction

The COVID-19 pandemic has emerged as one of the most formidable public health challenges in recent history. By Nov 1, 2022, there were already more than 98 million reported infections and 1.07 million deaths in the United States alone. Worldwide, the reported infections summed to 636 million with at least 6.61 million deaths [170]. The devastating effects of COVID-19 extends to the economy as well. For example, in the US, the unemployment rate peaked at 15.8 percent in April 2020 [171], and US GDP contracted at a 3.5% annualized rate for 2020 [172]. Similar economic impacts have been observed worldwide.

One of the most significant challenges in the early combat against COVID-19 was estimating the number of total infections. A significant number of COVID-19 infections were unreported, due to various factors such as the lack of testing and asymptomatic infections [16, 17, 18, 19, 20]. The inability in estimating these unreported infections allowed them to drive up disease transmission in many regions. For example, phylogenetic studies revealed that COVID-19 had locally spread in Washington state before early 2020, when active community surveillance was implemented [173]. There were only 23 reported infections in five major U.S. cities by March 1, 2020, but it has been estimated that there were in fact more than 28,000 total infections by then [174]. Similar trends were observed in other countries, such as in Italy, Germany, and the UK [175]. Despite having more advanced surveillance techniques such as serological studies, estimating the total number of infections continues to be a challenge for COVID-19 response even today [176, 177].

An accurate estimation of the number of total infections is a fundamental epidemiological question and critical for pandemic planning and response. Not withstanding its importance, there is not even a commonly agreed upon metric. One proposal is the *case ascertainment rate*, which is defined as the ratio of reported symptomatic infections to the actual number of symptomatic infections [58]. Another popular proposal is the *reported rate* α_{reported}, which is defined as the ratio of reported infections to total infections [178]. This definition includes asymptomatic infections, which are known to contribute substantially to community transmission [179, 180]. In this work, we focus on this particular measure.

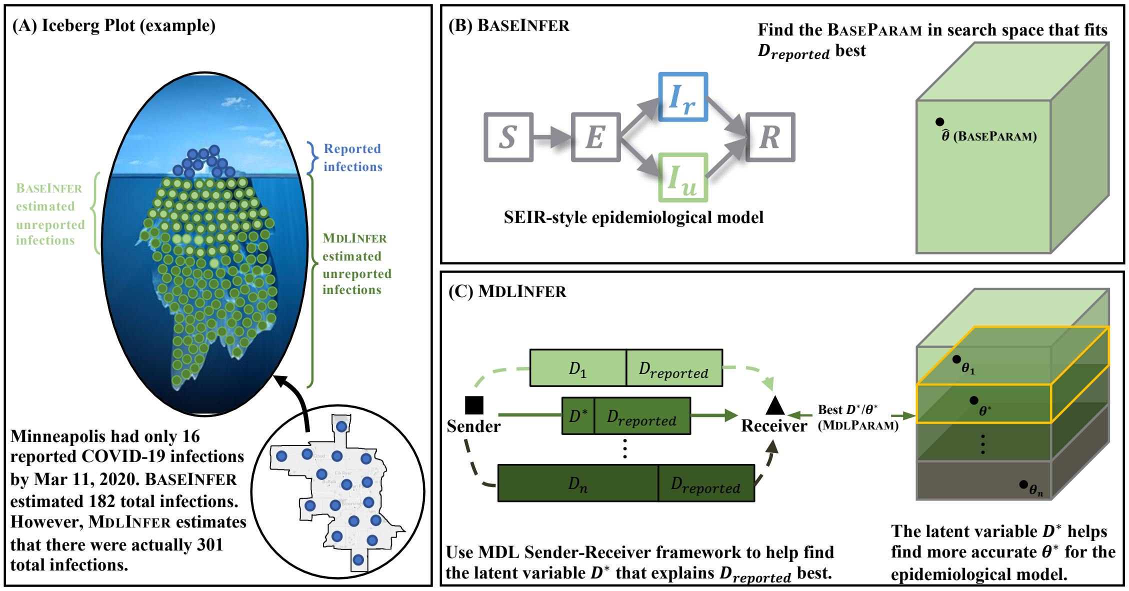


Figure 6.1: **Overview of our problem and methodology.** (**A**) We visualize the idea of reported rates using the iceberg. The visible portion above water are the reported infections, which is only a fraction of the whole iceberg representing total infections. (**B**) The usual practice is to calibrate an epidemiological model to reported data and compute the reported rate from the resultant parameterizaion of the model. Here, an SEIR-style model with explicit compartments for reported-vs-unreported infection is shown in the figure as an example. (**C**) Our new approach MDLINFER instead aims to compute a more accurate reported rate by finding a "best" parameterization *for the same epidemiological model* (i.e., SEIR-style model in this example) using a principled information theoretic formulation - two-part "sender-receiver" framework.

However, estimating the reported rate is challenging, and as a result all current methods have their limitations. One of the most effective current methods to identify the reported rate in a region is through large-scale serological studies [181, 182, 183]. These surveys use blood tests to identify the prevalence of antibodies against SARS-CoV-2 in a large population. The CDC COVID Data Tracker portal [184, 182] summarizes the results of serological studies conducted by commercial laboratories at a national level as well as at 10 specific sites. For example, the estimated reported rate was at most 0.1 in Minneapolis and South Florida as of April 2020. This means that there were at least 10 times more total infections than reported infections. While serological studies can give an accurate estimation, they are expensive and are not sustainable in the long run [185]. Furthermore, it is also challenging to obtain real-time data using such studies since there are unavoidable delays between sample collection and laboratory tests [184, 182]. Additional difficulties include sampling biases that make it necessary to use carefully designed heuristics to account for them [186]. Other methods include exploiting existing surveillance systems of related diseases like influenza, and using them to estimate symptomatic infections [187]. However, this can also be unreliable and requires ad-hoc corrections to account for the similarities between COVID-19 and influenza symptoms.

In the face of these challenges, data scientists and epidemiologists have devoted much time and effort to estimate the reported rate α_{reported} through epidemiological models O_{M}. By now, there exist carefully constructed models that capture the transmission dynamics of COVID-19 well [20, 58, 59, 60, 61, 62, 63, 64, 65, 54, 55, 56, 57]. In general, an epidemiological model O_{M} has a set of parameters Θ that we estimate from *observed data* using a so-called calibration procedure, CALIBRATE. In practice, the data we use for calibration can be the time series of the number of reported infections, which we call D_{reported}. To estimate the number of total infections, these models often explicitly include reported rate α_{reported} as one of their parameters, or include multiple parameters that jointly account for it. There are many calibration procedures commonly used in literature, such as RMSE-based [66] or Bayesian approaches [65, 64].

We call the above general methodology the basic approach to estimate the reported rate, or BASEINFER for short. It takes the epidemiological model O_{M}, a calibration procedure CALIBRATE, and observed data D_{reported} as input. The output of BASEINFER is then a baseline parameterization $\hat{\Theta}$ and, by extension, an estimated reported rate $\hat{\alpha}_{\text{reported}}$. Calibrating a parameterization is generally a complex, high-demensional problem, since $\hat{\Theta}$ consists of multiple interacting parameters. To make matters worse, there exist many possible parameterizations that show similar performance (e.g. in RMSE, likelihood) yet correspond to vastly different reported rates. BASEINFER cannot select between these competing parameterizations in a principled way: the parameterization $\hat{\Theta}$ it results in may or may not overfit the reported infections and may or may not predict future infections well. One method for selecting them is to take a Bayesian approach. That is, we choose a prior distribution, and then select the best parameterization that maximizes the posterior probability. Choosing such a prior, however, is ad-hoc and does not generalize well across different models O_{M}. As we will see in the experimental evaluation, minor differences in estimates of reported rates can indeed lead to very different forecasts of future trends and therewith intervention policy recommendations.

Instead, we propose a new information theory-based approach named MDLINFER. It takes the same input as BASEINFER, but uses a principled approach to determine the best parameterization Θ^*. It is based on the following central intuition: Suppose an oracle also gives us the time series of the number of *total infections* D in additional to the already known reported number of infections D_{reported}, and we are asked to describe D_{reported} as succinctly as possible. As we know both D and D_{reported}, it is trivial to estimate $\alpha'_{\text{reported}}$. If we know D and $\alpha'_{\text{reported}}$, it is trivial to describe D_{reported}, as it is simply $D \times \alpha'_{\text{reported}}$ plus a little bit of noise. Now to most succinctly describe D, we have to calibrate O_{M} to obtain Θ'. The only things we now have to describe are Θ', $\alpha'_{\text{reported}}$, the (small) errors that O_{M} makes in predicting D, and the (small) errors that we make predicting D_{reported} using D and $\alpha'_{\text{reported}}$. In practice, we are of course not given D, but the key idea of this work is to estimate D as a latent variable such that we can most succinctly describe (most accurately reconstruct) the dynamics of D_{reported}.

In practice, we need both a way of measuring how well a latent MODEL (i.e., D and its corresponding $\alpha'_{\text{reported}}$) describes the DATA (i.e., reported infections D_{reported}), as well as a way to find the best such MODEL. To do so, the Minimum Description Length (MDL) principle provides a statistically sound approach. MDL has been widely used for numerous optimization problems ranging from network summarization [188], causality inference [189], and failure detection in critical infrastructures [190]. MDL has also previously been used for some epidemiological problems, mainly in inferring patient-zero and associated infections in cascades over contact networks [191]. However, we are the first to propose an MDL-based approach on top of ODE-based epidemiological models, which are harder to formulate and optimize.

Specifically, we use two-part MDL (aka sender-receiver framework) consisting of hypothetical actors *S* and *R*: Sender *S* has the DATA and wants to transmit it to receiver *R* using as few bits as possible [192]. Hence, sender *S* searches for the best possible MODEL, which minimizes the overall cost of encoding and transmitting both the MODEL and the DATA given the MODEL. Following the convention in information theory, we use $L(\text{MODEL})$ to denote the number of bits required to encode the MODEL; and $L(\text{DATA}|\text{MODEL})$ to denote the number of bits required to encode the DATA, D_{reported}, given the MODEL. The overall objective of our optimization problem is to infer an optimal MODEL*, which minimizes $L(\text{MODEL}) + L(\text{DATA}|\text{MODEL})$. To put MDL to practice for our problem, we carefully design our MDL cost to minimize the discrepancy in fitting D_{reported}. This cost ensures the generalizability of our learned D^* and $\alpha^*_{\text{reported}}$ - it can avoid overfitting on D_{reported} and predict the future reported infections well. Our later experiments exactly show this. Our approach, MDLINFER, can be applied to any ODE model since two-part MDL does not assume about the nature of the DATA or the MODEL.

We compare MDLINFER and BASEINFER using two different ODE-based epidemiological models: SAPHIRE [64] and $\text{SEIR} + \text{HD}$ [65] as O_{M}. Following their literature [64, 65], we use Markov Chain Monte Carlo (MCMC) as the calibration procedure CALIBRATE for SAPHIRE and iterated filtering (IF) for $\text{SEIR} + \text{HD}$, both of with are Bayesian approaches[193]. Both these epidemiological models have previously been shown to perform well in fitting reported infections and provided insight that was beneficial for the COVID-19 response. SAPHIRE focuses on two key features of the outbreak: high covertness and high transmissibility that drove the outbreak of COVID-19 in Wuhan. $\text{SEIR} + \text{HD}$ investigates how non-pharmaceutical interventions like social distancing will be needed to maintain epidemic control. These models are broadly representative to show that MdlInfer gives consistent performance across multiple epidemiological models with different dynamics. The experiments clearly show that our proposed MDL-based approach MdlInfer performs superior to the state of the art. To illustrate, we give an example in Figure 6.1. By March 11, 2020, the Minneapolis Metro Area had only 16 COVID-19 reported infections. BaseInfer estimated 182 total infections, which are colored as light green in the iceberg. On the other hand, our MdlInfer gives an estimate of 301 total infections shown below the sea level, which is closer to the total infections estimated from serological studies [182, 184]. Additionally, MdlInfer also leads to better fits and *future* projections on reported infections. We also demonstrate that MdlInfer can aid policy making by analyzing counter-factual non-pharmaceutical interventions, while inaccurate BaseInfer estimates lead to wrong non-pharmaceutical intervention conclusions.

6.2 Methods

6.2.1 Two-part Sender-receiver Framework

In this work, we use two-part sender-receiver framework. The conceptual goal of the framework is to transmit the Data from the possession of the hypothetical sender *S* to the hypothetical receiver *R*. We assume the sender does this by first sending a Model and then sending the Data under this Model. In this MDL framework, we want to minimize the number of bits for this process. We do this by identifying the Model that encodes the Data such that the total number of bits needed to encode both the Model and the Data is minimized. Hence our cost function in the total number of bits needed is composed of two parts: (i) model cost $L(\text{Model})$: The cost in bits of encoding the Model and (ii) data cost $L(\text{Data}|\text{Model})$: The cost in bits of encoding the Data given the Model. Intuitively, the idea is that a good Model will lead to a fewer number of bits needed to encode both Model and Data. We formulate the general MDL optimization problem as follows: Given the DATA, $L(\text{MODEL})$, and $L(\text{DATA}|\text{MODEL})$, find MODEL* such that

$$\text{MODEL}^* = \arg\min_{\text{MODEL}} L(\text{MODEL}) + L(\text{DATA}|\text{MODEL}) \tag{6.1}$$

In our situation, the DATA is the reported COVID-19 infections D_{reported}: it is the only real-world data given to us. Note that total infections are not directly observed. As described in section 6.1, the MODEL is intuitively $(D, \alpha'_{\text{reported}})$. Here D refers to a candidate total infections time series, and $\alpha'_{\text{reported}}$ is the corresponding reported rate. Specifically, we calibrate O_{M} on (D, D_{reported}) using CALIBRATE to get the "candidate" parameterization $\Theta^{'}$, and then compute $\alpha'_{\text{reported}}$ from $\Theta^{'}$. Further, we choose to also add $\hat{\Theta}$ estimated by BASEINFER, making our MODEL to be $(D, \Theta^{'}, \hat{\Theta})$. There are alternative MODELs that can be considered, but we choose this MODEL $= (D, \Theta^{'}, \hat{\Theta})$. Note that as two-part MDL (and MDL in general) does not assume the nature of the DATA or the MODEL, our MDLINFER can be applied to any ODE model. Intuitively, the advantages of the MDLINFER over BASEINFER are:

1. We introduce a latent variable D to help the calibration: With D and its corresponding $\alpha'_{\text{reported}}$, we can focus on a smaller search space by fixing the reported rate and get a better fit on D_{reported}. In contrast, BASEINFER ignores the D but directly searches in the whole parameter space for $\hat{\Theta}$, which is a complex problem in high-dimensional space since there are multiple parameters to estimate simultaneously. As shown in subsection 6.3.2, we fit the observed D_{reported} better (measured by lower RMSE) in most regions.

2. We performed a principled model selection framework to search for the D. We carefully designed our MDL cost to minimize the discrepancy in fitting D_{reported}. This MDL cost ensures the generalizability of our learned D and $\alpha'_{\text{reported}}$ to avoid the overfitting of D_{reported}. Specifically, as we described in section 6.2, we attempt different Ds that correspond to different reported rates in our algorithm. This can be

viewed as a linear search on α_{reported}, which helps to avoid getting a locally optimal solution. Again, as shown in subsection 6.3.2, good generalizability leads to better forecasts of future reported infections.

Next, we give more details how to formulate our problem of estimating total infections D.

6.2.2 MDL Formulation

First, we need to introduce some notations. Given an epidemiological model O_{M} and the paramterization $\hat{\Theta}$ estimated by BASEINFER, we can compute the reported infections. However, this is only an estimate of the reported infections rather than the exact D_{reported}. This is because even though we have already calibrated O_{M} using D_{reported}, the calibration cannot be perfect, and there will be differences between these estimated reported infections and D_{reported}. Here, we term this estimated reported infections as $D_{\text{reported}}(\hat{\Theta})$. We can also estimate the total infections $D(\hat{\Theta})$ for O_{M} in the same way. Similarly, we have the $D_{\text{reported}}(\Theta^{'})$ and $D(\Theta^{'})$ for $\Theta^{'}$. As described in section 6.1, we can also calculate the reported rate $\hat{\alpha}_{\text{reported}}$ and $\alpha'_{\text{reported}}$ using $\hat{\Theta}$ and $\Theta^{'}$. With these notations, next we will formulate the space of all possible MODELs and give the equation for the cost in bits of encoding MODEL and DATA.

MODEL *Space*

We have MODEL $= (D, \Theta^{'}, \hat{\Theta})$ as described above. Hence our MODEL space will be all possible daily sequences for D and all possible parameterizations for $\Theta^{'}$ and $\hat{\Theta}$. The MDL framework will search in this space to find the MODEL*.

Model Cost

With MODEL $= (D, \Theta^{'}, \hat{\Theta})$, we conceptualize the model cost by imagining that the sender *S* will send the MODEL $= (D, \Theta^{'}, \hat{\Theta})$ to the receiver *R* in three parts: (i) first send the $\hat{\Theta}$ by encoding $\hat{\Theta}$ directly (ii) next send the $\Theta^{'}$ given $\hat{\Theta}$ by encoding $\Theta^{'} - \hat{\Theta}$ and (iii) then send

D given Θ' and $\hat{\Theta}$ by encoding $\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})$. Intuitively, both $\alpha'_{\text{reported}} \times D$ and $D_{\text{reported}}(\hat{\Theta})$ should be close to D_{reported}, and the receiver could recover the D using $\hat{\Theta}$, $\alpha'_{\text{reported}}$, and $D_{\text{reported}}(\hat{\Theta})$ as they have already been sent. We term the model cost as $L(D, \Theta', \hat{\Theta})$ with three components: $\textsc{Cost}(\hat{\Theta})$, $\textsc{Cost}(\Theta'|\hat{\Theta})$, and $\textsc{Cost}(D|\Theta', \hat{\Theta})$. Hence,

$$L(D, \Theta', \hat{\Theta}) = \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta' - \hat{\Theta}|\hat{\Theta}) + \textsc{Cost}(\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta', \hat{\Theta}) \tag{6.2}$$

For Equation 6.2, the $\textsc{Cost}(\cdot)$ function gives the total number of bits we need to spend in encoding each term. The details of the encoding method can be found in Appendix C.

Data Cost

We need to send the $\textsc{Data} = D_{\text{reported}}$ next given the $\textsc{Model}$. Given $\textsc{Model} = (D, \Theta', \hat{\Theta})$, we send $\textsc{Data}$ by encoding $\frac{D - D_{\text{reported}}}{1 - \alpha'_{\text{reported}}} - D(\Theta')$. Intuitively, $D - D_{\text{reported}}$ corresponds to the unreported infections, and $1 - \alpha'_{\text{reported}}$ is the unreported rate. Therefore, $\frac{D - D_{\text{reported}}}{1 - \alpha'_{\text{reported}}}$ should be close to the total infections D and $D(\Theta')$. The receiver could also recover the D_{reported} using D, $\alpha'_{\text{reported}}$, and $D(\Theta')$ as they have already been sent. We term data cost as $L(D_{\text{reported}}|D, \Theta', \hat{\Theta})$ and formulate it as Equation 6.3.

$$fL(D_{\text{reported}}|D, \Theta', \hat{\Theta}) = \textsc{Cost}(\frac{D - D_{\text{reported}}}{1 - \alpha'_{\text{reported}}} - D(\Theta')|D, \Theta', \hat{\Theta}) \tag{6.3}$$

Total Cost

With $L(D, \Theta', \hat{\Theta})$ as in Equation 6.2 and $L(D_{\text{reported}}|D, \Theta', \hat{\Theta})$ as in Equation 6.3 above, the total cost $L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$ is:

$$
\begin{aligned}
L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta}) &= L(D, \Theta^{'}, \hat{\Theta}) + L(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) \\
&= \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta^{'} - \hat{\Theta}|\hat{\Theta}) \\
&+ \textsc{Cost}(\alpha^{'}_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta^{'}, \hat{\Theta}) \\
&+ \textsc{Cost}(\frac{D - D_{\text{reported}}}{1 - \alpha^{'}_{\text{reported}}} - D(\Theta^{'})|D, \Theta^{'}, \hat{\Theta})
\end{aligned}
\tag{6.4}
$$

6.2.3 Problem Statement

Note that our main objective is to estimate the total infections D. With $L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta})$ formulated in Equation 6.4, we can state the problem as: Given the time sequence D_{reported}, epidemiological model O_{M}, and a calibration procedure CALIBRATE, find D^* that minimizes the MDL total cost i.e.

$$D^* = \arg\min_{D} \; L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta}) \tag{6.5}$$

6.2.4 Algorithm

Next, we will present our algorithm to solve the problem in Equation 6.5. Note that directly searching D^* naively is intractable since D^* is a daily sequence not a scalar. Instead, we propose first finding a "good enough" reported rate $\alpha^*_{\text{reported}}$ quickly with the constraint $D = \frac{D_{\text{reported}}}{\alpha^*_{\text{reported}}}$ to reduce the search space. Then with this $\alpha^*_{\text{reported}}$, we can search for the optimal D^* in Equation 6.5. Hence we propose a two-step algorithm: (i) do a linear search to find a good reported rate $\alpha^*_{\text{reported}}$ (ii) given the $\alpha^*_{\text{reported}}$ found above, use an optimization method to find the D^* that minimizes $L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta})$ with $\alpha^*_{\text{reported}}$ constraints.

Step 1: Find the $\alpha^*_{\text{reported}}$

In step 1, we do a linear search on α_{reported} to find the $\alpha^*_{\text{reported}}$. As stated before, we use $\frac{D_{\text{reported}}}{\alpha_{\text{reported}}}$ as D in $L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta})$ to help reduce the search space. Here, we formulate step 1 algorithm as Equation 6.6.

$$\alpha^*_{\text{reported}} = \arg\min_{\alpha_{\text{reported}}} L(D_{\text{reported}}, \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}, \Theta', \hat{\Theta}) \tag{6.6}$$

Algorithm 4 GETALPHA (Step 1: Find the $\alpha^*_{\text{reported}}$)

Inputs: O_{M}, CALIBRATE, D_{reported}, $\hat{\Theta}$.
The array to save the MDL cost: CostArray = [ ]
for α_{reported} in the grid search space from 0.01 to 1 with step 0.01 **do**
    $D = \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}$
    Calibrate candidate parameterization $\Theta' = \text{CALIBRATE}(O_{\text{M}}, (D, D_{\text{reported}}))$
    Save the MDL cost for α_{reported} in CostArray[α_{reported}] = $L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$
end for
Find the $\alpha^*_{\text{reported}} = \arg\min_{\alpha_{\text{reported}}}$ CostArray[α_{reported}]
Outputs: Reported rate $\alpha^*_{\text{reported}}$

Step 2: Find the D^* Given $\alpha^*_{\text{reported}}$

Algorithm 5 GETTOTALINFECTIONS (Step 2: Find the D^* given $\alpha^*_{\text{reported}}$)

Inputs: O_{M}, CALIBRATE, D_{reported}, $\alpha^*_{\text{reported}}$, $\hat{\Theta}$.
Find the $D^* = \arg\min_D L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$. (using the Nelder-Mead algorithm).
Outputs: Total infections D^*

With the $\alpha^*_{\text{reported}}$ found in step 1, we next find the D^* that minimizes the $L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$. Note that we have already found a good $\alpha^*_{\text{reported}}$, we can constrain the D^* to ensure that the sum of D^* equals to the sum of $\frac{D_{\text{reported}}}{\alpha^*_{\text{reported}}}$. We use the Nelder-Mead method [194] to solve this constrained optimization problem for D^*. Here, we formulate step 2 algorithm as Equation 6.7.

$$D^* = \arg\min_D L(D_{\text{reported}}, D, \Theta', \hat{\Theta}) \tag{6.7}$$

We give the pseudo-code for MDLINFER as follows:

Algorithm 6 MDLINFER

1: **Inputs:** Epidemiological model O_{M}, calibration procedure CALIBRATE, reported infections time sequence D_{reported}.
2: Calibrate baseline parameterization $\hat{\Theta} = \textsc{Calibrate}(O_{\mathrm{M}}, D_{\mathrm{reported}})$
3: Step 1: Find $\alpha^*_{\mathrm{reported}} = \textsc{GetAlpha}(O_{\mathrm{M}}, \textsc{Calibrate}, D_{\mathrm{reported}}, \hat{\Theta})$
4: Step 2: Find $D^* = \textsc{GetTotalInfections}(O_{\mathrm{M}}, \textsc{Calibrate}, D_{\mathrm{reported}}, \alpha^*_{\mathrm{reported}}, \hat{\Theta})$
5: **Outputs:** Total infections D^*

6.2.5 BASEINFER and MDLINFER Formulation

Here, we also give the mathematical formulations for BASEINFER and MDLINFER. As described in section 6.1, given an epidemiological model O_{M}, a typical approach is to calibrate the O_{M} to D_{reported} using the calibration procedure CALIBRATE. We call this methodology as $\textsc{BaseInfer}(O_{\mathrm{M}}, \textsc{Calibrate}, D_{\mathrm{reported}})$. As in Equation 6.8, the output of BASEINFER is the baseline parameterization (BASEPARAM) $\hat{\Theta}$.

$$\begin{aligned} \hat{\Theta} &= \textsc{BaseInfer}(O_{\mathrm{M}}, \textsc{Calibrate}, \mathrm{D}_{\mathrm{reported}}) \\ &= \textsc{Calibrate}(O_{\mathrm{M}}, \mathrm{D}_{\mathrm{reported}}) \end{aligned} \tag{6.8}$$

As for the MDLINFER, it also takes the same input (O_{M}, CALIBRATE, D_{reported}) as BASEINFER. Assume we are given the total infections D, we calibrate the O_{M} on $(D, D_{\mathrm{reported}})$ to get a "candidate" paramterization $\Theta^{'}$ in Equation 6.9.

$$\Theta^{'} = \textsc{Calibrate}(O_{\mathrm{M}}, (D, D_{\mathrm{reported}})) \tag{6.9}$$

However, we are not given the D. Hence, we use the MDL framework to find such D^* as in Equation 6.7. With such D^*, we could finally calibrate the O_{M} on $(D^*, D_{\mathrm{reported}})$ and gets another parameterization Θ^*. As in Equation 6.10, we call Θ^* as the MDL parameterization, or MDLPARAM.

$$\begin{aligned}\Theta^* &= \text{MDLINFER}(O_{\mathrm{M}}, \text{CALIBRATE}, \mathrm{D}_{\text{reported}}) \\ &= \text{CALIBRATE}(O_{\mathrm{M}}, (D^*, D_{\text{reported}}))\end{aligned} \tag{6.10}$$

where $D^* = \arg\min_D L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$. Intuitively, if $\hat{\Theta}$ estimated by BASEINFER is perfect, MDLINFER will also give the same Θ^* as $\hat{\Theta}$.

6.2.6 Epidemiological Models

Next, we describe the two epidemiological models we use in our experiments: $\mathrm{SEIR} + \mathrm{HD}$ and $\mathrm{SAPHIRE}$ model. $\mathrm{SEIR} + \mathrm{HD}$ [65] consists of 10 states: Susceptible S, exposed E, pre-symptomatic I_P, severe symptomatic I_S, mild symptomatic I_M, asymptomatic I_A, hospitalized (eventual death) H_D, hospitalized (eventual recover) H_R, recovered R, and dead D. The parameters to be calibrated are the transmission rate β_0 (the transmission rate in the absence of interventions), σ (the proportional reduction on β_0 under shelter-in-place), and E_0 (number of initial infections). The other parameters are fixed and given. They assume the importations only happen at the beginning of the pandemic (captured by E_0), and the total population N remains constant. We also extend $\mathrm{SEIR} + \mathrm{HD}$ model to infer two more parameters: α (proportion of asymptomatic infections) and α_1 (proportion of new symptomatic infections that are reported). We compute the new reported infections and unreported infections as follows:

1. $\text{New reported infections} = \alpha_1 \times (N_{I_P I_S} + N_{I_P I_M})$: Here $N_{I_P I_S} + N_{I_P I_M}$ is the number of new symptomatic infections everyday. $N_{I_P I_S}$ is the number of patients switching their state from I_P to I_S (and similarly for $N_{I_P I_M}$). We assume α_1 proportion of new symptomatic infections every day are reported.

2. $\text{New unreported infections} = (1 - \alpha_1) \times (N_{I_P I_S} + N_{I_P I_M}) + N_{E I_A}$.

$\mathrm{SAPHIRE}$ [64] consists of 7 states: Susceptible S, exposed E, pre-symptomatic P, ascertained infectious I, unascertained infectious A, hospitalized H, and recovered R. The parameters to be calibrated are the transmission rate β and reported rate r while keeping other parameters fixed as given values. We also compute the new reported infections and unreported infections as follows:

1. $\text{New reported infections} = \frac{rP}{D_p}$: Here $\frac{P}{D_p}$ is the number of new infections from presymptomatic every day. D_p is the parameter for the presymptomatic infectious period and is fixed. r is the reported rate estimated by the epidemiological model.

2. $\text{New unreported infections} = \frac{(1-r)P}{D_p}$.

6.2.7 Estimating Infections Using BASEPARAM and MDLPARAM

Here, we describe how we get the estimations in section 6.3 using BASEPARAM and MDLPARAM. Here we use the BASEPARAM from BASEINFER as the example (this can also be repeated for MDLPARAM for MDLINFER). Using the epidemiological model O_{M}, we can calculate the BASEPARAM's estimation of total infections BASEPARAM$_{\text{Tinf}}$ as the cumulative values of $D(\hat{\Theta})$ from pandemic's beginning. $D_{\text{reported}}(\hat{\Theta})$ can be directly used as the BASEPARAM's estimation of reported infections. For the cumulative reported rate BASEPARAM$_{\text{Rate}}$, we calculate it as the cumulative values of NYT-Rinf divided by $D(\hat{\Theta})$. For the symptomatic rate, $\text{SEIR} + \text{HD}$ model [65] could estimate the number of symptomatic rate BASEPARAM$_{\text{Symp}}$ by dividing the number of infections in state I_S and I_M by the population number. However, SAPHIRE model [64] does not contain states that correspond to the symptomatic cases, so we cannot estimate the symptomatic rate using this model.

6.3 Results

Next, we present our empirical findings on a large set of experiments in different geographical regions and time periods. We choose 8 regions and periods based on the severity of the outbreak and the availability of serological studies and symptomatic surveillance data. In each region, we divide the timeline into two time periods: (i) observed period, when only the number of reported infections are available, and both BASEINFER and MDLINFER are used to learn the baseline parameterization (BASEPARAM) $\hat{\Theta}$ and MDL parameterization (MDLPARAM) Θ^*, and (ii) forecast period, where we evaluate the forecasts generated by the parameterizations learned in the observed period. To handle the time-varying reported rates, we divide the observed period into multiple sub-periods and learn different reported rates for each sub-period separately.

6.3.1 Estimating Total Infections: MDLINFER Estimates Total Infections More Accurately than BASEINFER

Here, we use the point estimates of the total infections calculated from serological studies as the ground truth (black dots shown in Figure 6.2). We call it SEROSTUDY$_{\mathrm{Tinf}}$. We also plot MDLINFER's estimation of total infections, MDLPARAM$_{\mathrm{Tinf}}$, in the same figure (red curve). To compare the performance of MDLINFER and BASEINFER with SEROSTUDY$_{\mathrm{Tinf}}$, we use the cumulative value of estimated total infections. Note that values from the serological studies are not directly comparable with the total infections because of the lag between antibodies becoming detectable and infections being reported [184, 182]. In Figure 6.2, we have already accounted for this lag following CDC study guidelines [184, 182] (See section 6.2 for details). The vertical black lines shows a 95% confidence interval for SEROSTUDY$_{\mathrm{Tinf}}$. The blue curve represents total infections estimated by BASEINFER, BASEPARAM$_{\mathrm{Tinf}}$. As seen in the figure, MDLPARAM$_{\mathrm{Tinf}}$ falls within the confidence interval of the estimates given by serological studies. Significantly, in Figure 6.2**B** and Figure 6.2**F** for South Florida, BASEINFER for SAPHIRE model [64] overestimates the total infections, while for SEIR + HD model underestimates the total infections. However, MDLINFER consistently estimates the total infections correctly. This observation shows that as needed, MDLPARAM$_{\mathrm{Tinf}}$ can improve upon the BASEPARAM$_{\mathrm{Tinf}}$ in either di-

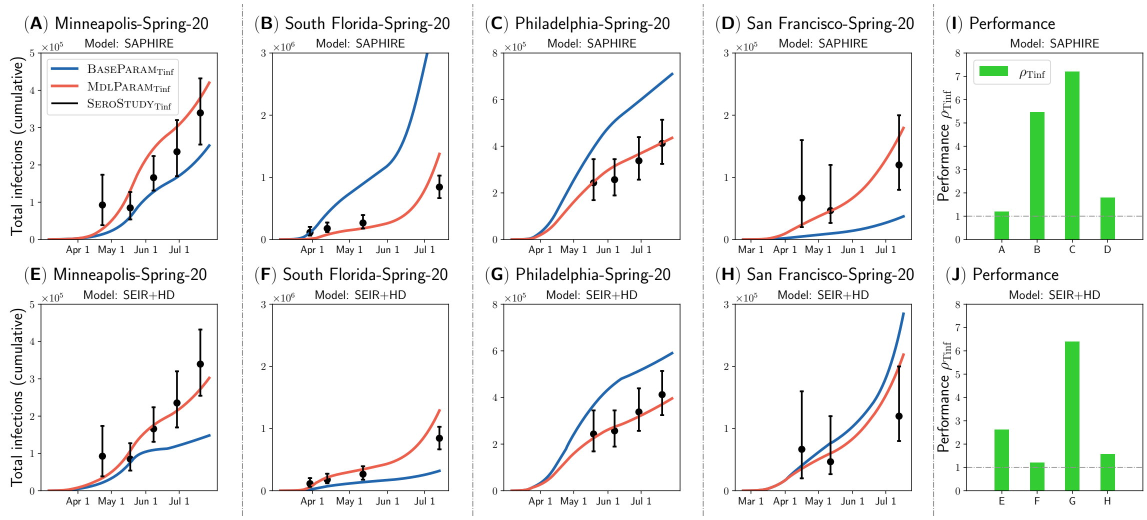


Figure 6.2: **MDLINFER (red) gives a closer estimation of total infections to serological studies (black) than BASEINFER (blue) on various geographical regions and time periods.** Note that both approaches try to fit the serological studies without being informed with them. (**A**)-(**H**) The red and blue curves represent MDLINFER's estimation of total infections, $\text{MDLPARAM}_{\text{Tinf}}$, and BASEINFER's estimation of total infections, $\text{BASEPARAM}_{\text{Tinf}}$, respectively. The black point estimates and confidence intervals represent the total infections estimated by serological studies [184, 182], $\text{SEROSTUDY}_{\text{Tinf}}$. (**A**)-(**D**) use SAPHIRE model and (**E**)-(**H**) use $\text{SEIR} + \text{HD}$ model. (**I**)-(**J**) The performance metric, ρ_{Tinf}, comparing $\text{MDLPARAM}_{\text{Tinf}}$ against $\text{BASEPARAM}_{\text{Tinf}}$ in fitting serological studies is shown for each region. (**I**) is for SAPHIRE model in (**A**)-(**D**), and (**J**) is for $\text{SEIR} + \text{HD}$ model in (**E**)-(**H**).

rection (i.e., by increasing or decreasing the total infections). Note that the $\text{MDLPARAM}_{\text{Tinf}}$ curves from both models are closer to the $\text{SEROSTUDY}_{\text{Tinf}}$ even when the $\text{BASEPARAM}_{\text{Tinf}}$ curves are different. The results of better accuracy in spite of various geographical regions and time periods show that MDLINFER is consistently able to estimate total infections more accurately.

To quantify the performance gap between the two approaches, we first compute the root mean squared error (RMSE) between $\text{SEROSTUDY}_{\text{Tinf}}$ and $\text{BASEPARAM}_{\text{Tinf}}$. We also compute the same between $\text{SEROSTUDY}_{\text{Tinf}}$ and $\text{MDLPARAM}_{\text{Tinf}}$. We then compute the ratio, ρ_{Tinf}, of the two RMSE errors as $\frac{\text{RMSE}(\text{BASEPARAM}_{\text{Tinf}}, \text{SEROSTUDY}_{\text{Tinf}})}{\text{RMSE}(\text{MDLPARAM}_{\text{Tinf}}, \text{SEROSTUDY}_{\text{Tinf}})}$. Note that the values of ρ_{Tinf} being greater than 1 implies that the $\text{MDLPARAM}_{\text{Tinf}}$ is closer to $\text{SEROSTUDY}_{\text{Tinf}}$ estimates than $\text{BASEPARAM}_{\text{Tinf}}$. In Figure 6.2**I** and Figure 6.2**J**, we plot ρ_{Tinf}. Overall,

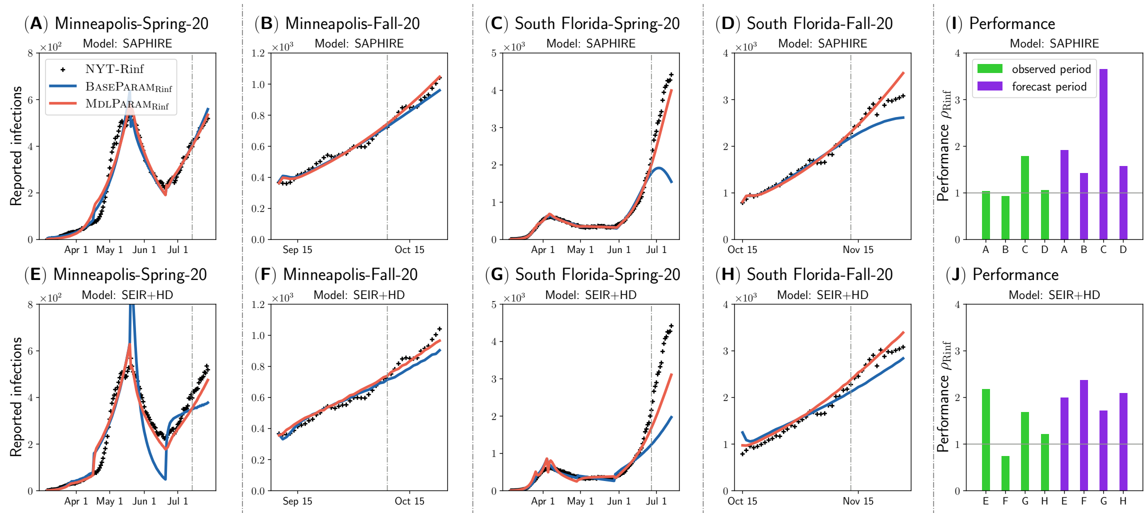


Figure 6.3: **MdlInfer (red) gives a closer estimation of reported infections (black) than BaseInfer (blue) on various geographical regions and time periods.** We use the reported infections in the observed period as inputs and try to forecast the future reported infections (forecast period). (**A**)-(**H**) The vertical grey dash line divides the observed period (left) and forecast period (right). The red and blue curves represent MdlInfer's estimation of reported infections, $\textsc{MdlParam}_{\mathrm{Rinf}}$, and BaseInfer's estimation of reported infections, $\textsc{BaseParam}_{\mathrm{Rinf}}$, respectively. The black plus symbols represent the reported infections collected by the New York Times (NYT-Rinf). (**A**)-(**D**) use $\mathrm{SAPHIRE}$ model and (**E**)-(**H**) use $\mathrm{SEIR+HD}$ model. (**I**)-(**J**) The performance metric, ρ_{Rinf}, comparing $\textsc{MdlParam}_{\mathrm{Rinf}}$ against $\textsc{BaseParam}_{\mathrm{Rinf}}$ in fitting reported infections is shown for each region. (**I**) is for $\mathrm{SAPHIRE}$ model in (**A**)-(**D**), and (**J**) is for $\mathrm{SEIR+HD}$ model in (**E**)-(**H**). Note that ρ_{Rinf} larger than 1 means that $\textsc{MdlParam}_{\mathrm{Rinf}}$ is closer to NYT-Rinf than $\textsc{BaseParam}_{\mathrm{Rinf}}$.

the ρ_{Tinf} values are greater than 1 in Figure 6.2**I** and Figure 6.2**J**, which indicates that MdlInfer performs better than BaseInfer. Note that even when the value of ρ_{Tinf} is 1.20 for Figure 6.2**A**, the improvement made by $\textsc{MdlParam}_{\mathrm{Tinf}}$ over $\textsc{BaseParam}_{\mathrm{Tinf}}$ in terms of RMSE is about 12091. Hence, one can conclude that MdlInfer is indeed superior to BaseInfer, when it comes to estimating total infections.

6.3.2 Estimating Reported Infections: MdlInfer Leads to Better Fit and Projection than BaseInfer at Different Stages of the COVID-19 Epidemic

Here, we first use the observed period to learn the parameterizations. We then *forecast* the future reported infections (i.e., forecast periods), which were *not* accessible to the model while training. The results are summarized in Figure 6.3. In Figure 6.3**A** to Figure 6.3**H**, the vertical grey dash line divides the observed and forecast period. The black plus symbols represent reported infections collected by the New York Times, NYT-Rinf. The red curve represents MDLINFER's estimation of reported infections, $\text{MDLPARAM}_{\text{Rinf}}$. Similarly, the blue curve represents BASEINFER's estimation of reported infections, $\text{BASEPARAM}_{\text{Rinf}}$. Note that the curves to the right of the vertical grey line are future predictions. As seen in Figure 6.3, $\text{MDLPARAM}_{\text{Rinf}}$ aligns more closely with NYT-Rinf than $\text{BASEPARAM}_{\text{Rinf}}$, indicating the superiority of MDLINFER in fitting and forecasting reported infections.

We define a performance metric ρ_{Rinf} as $\frac{\text{RMSE}(\text{BASEPARAM}_{\text{Rinf}},\text{NYT-Rinf})}{\text{RMSE}(\text{MDLPARAM}_{\text{Rinf}},\text{NYT-Rinf})}$ to compare $\text{MDLPARAM}_{\text{Rinf}}$ against $\text{BASEPARAM}_{\text{Rinf}}$ in a manner similar to ρ_{Tinf}. In Figure 6.3**I** and Figure 6.3**J**, we plot the ρ_{Rinf} for the observed and forecast period. In both periods, we notice that the ρ_{Rinf} is close to or greater than 1. This further shows that MDLINFER has a better or at least closer fit for reported infections than BASEINFER. Additionally, the ρ_{Rinf} for the forecast period is even greater than ρ_{Rinf} for the observed period, which shows that MDLINFER performs even better than BASEINFER while forecasting.

Note that Figure 6.3**A**, **C**, **E**, **G** correspond to the early state of the COVID-19 epidemic in spring and summer 2020, and Figure 6.3**B**, **D**, **F**, **H** correspond to fall 2020. We can see that MDLINFER performs well in estimating temporal patterns at different stages of the COVID-19 epidemic.

6.3.3 Estimating Symptomatic Rate Trends: MDLINFER Estimates the Symptomatic Rate Trends More Accurately than BASEINFER

We validate this observation using Facebook's symptomatic surveillance dataset [195]. We plot MDLINFER's and BASEINFER's estimated symptomatic rate over time and overlay the estimates and standard error from the symptomatic surveillance data in Figure 6.4. The red and blue curves are MDLINFER's and BASEINFER's estimation of symptomatic rates, $\text{MDLPARAM}_{\text{Symp}}$ and $\text{BASEPARAM}_{\text{Symp}}$ respectively. Note that SAPHIRE model does not

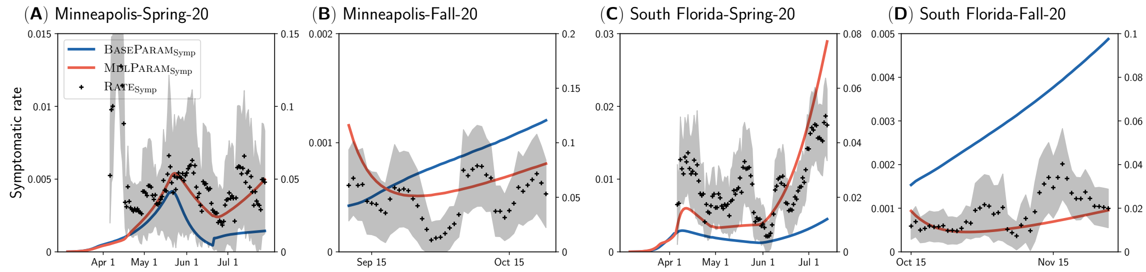


Figure 6.4: **MDLINFER (red) gives a closer estimation of the trends of symptomatic rate (black) than BASEINFER (blue) on various geographical regions and time periods.** (**A**)-(**D**) The red and blue curves represent MDLINFER's estimation of symptomatic rate, MDLPARAM$_{\text{Symp}}$, and BASEINFER's estimation of symptomatic rate, BASEPARAM$_{\text{Symp}}$, respectively. They use the y-scale on the left. The black points and the shaded regions are the point estimate with standard error for RATE$_{\text{Symp}}$ (the COVID-related symptomatic rates derived from the symptomatic surveillance dataset [195, 196]). They use the y-scale on the right. Note that we focus on trends instead of the exact numbers, hence MDLPARAM$_{\text{Symp}}$/BASEPARAM$_{\text{Symp}}$, and RATE$_{\text{Symp}}$ may scale differently.

contain states corresponding to the symptomatic infections. Therefore, we only focus on SEIR + HD model. We compare the trends of the MDLPARAM$_{\text{Symp}}$ and BASEPARAM$_{\text{Symp}}$ with the symptomatic surveillance results. We focus on trends rather than actual values because the symptomatic rate numbers could be biased [195] (see section 6.2 for a detailed discussion) and therefore cannot be compared directly with model outputs like what we have done for serological studies. As seen in Figure 6.4, MDLPARAM$_{\text{Symp}}$ captures the trends of the surveyed symptomatic rate RATE$_{\text{Symp}}$ (black plus symbols) better than BASEPARAM$_{\text{Symp}}$.

To summarize, these three sets of experiments in subsection 6.3.1, subsection 6.3.2, and subsection 6.3.3 together demonstrate that BASEINFER fail to accurately estimate the total infections including unreported ones. On the other hand, MDLINFER estimates total infections closer to those estimated by serological studies and better fits reported infections and symptomatic rate trends.

6.3.4 Evaluating the Effect of Non-pharmaceutical Interventions

We have already shown that MDLINFER is able to estimate the number of total infections accurately. In the following three observations, we show that such accurate estimations are important for evaluating the effect of non-pharmaceutical interventions.

MDLINFER *Reveals that a Large Majority of COVID-19 Infections were Unreported*

We compute the *cumulative* reported rate $\text{MDLPARAM}_{\text{Rate}}$ measured by the ratio of the cumulative value of reported infections to the total infections estimated by MDLINFER over time and plotted it for Minneapolis-Spring-20 in Figure 6.5**A**. The figure shows that the $\text{MDLPARAM}_{\text{Rate}}$ increases in early March, and then gradually decreases. This observation is explained by the community spread-driven COVID-19 outbreaks that were not reported until early March, which fits earlier studies [187].

Non-pharmaceutical Interventions on Asymptomatic and Presymptomatic Infections are Essential to Control the COVID-19 Epidemic

Our simulations show that non-pharmaceutical interventions on asymptomatic and presymptomatic infections are essential to control COVID-19. Here, we plot the simulated reported infections of MDLPARAM in Figure 6.5**B** (red curve). We then repeat the simulation of reported infections for 5 different scenarios: (i) isolate just the reported infections, (ii) isolate just the symptomatic infections, and isolate symptomatic infections in addition to (iii) 25%, (iv) 50%, and (v) 75% of both asymptomatic and presymptomatic infections. In our setup, we assume that the infectivity reduces by half when a person is isolated. As seen in Figure 6.5**B**, when only the reported infections are isolated, there is almost no change in the "future" reported infections. However, when we isolate both the reported and symptomatic infections, the reported infections decreases significantly. Even here, the reported infections are still not in decreasing trend. On the other hand, non-pharmaceutical interventions for some fraction of asymptomatic and presymptomatic infections make reported infections

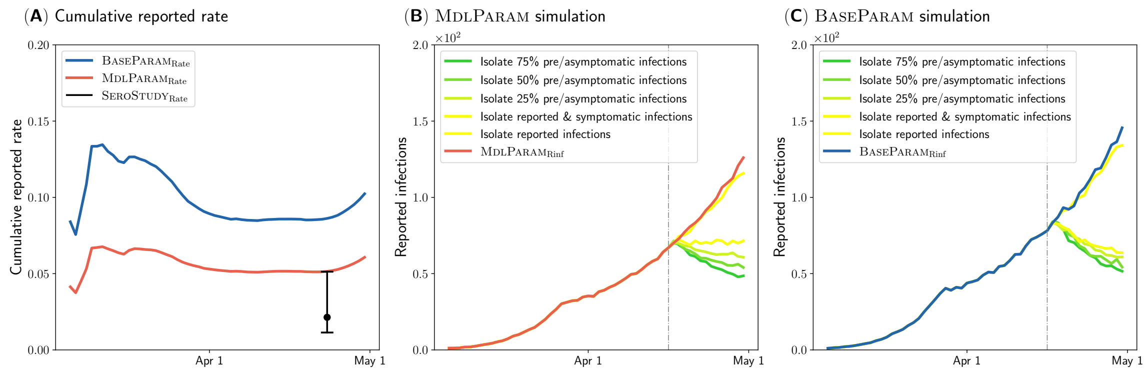


Figure 6.5: (**A**) MdlInfer estimates cumulative reported rate more accurately than BaseInfer: The blue and red curve represent the cumulative reported rate estimated by BaseInfer, BaseParam$_{\text{Rate}}$, and by MdlInfer, MdlParam$_{\text{Rate}}$, respectively. The black point estimate and its confidence interval represent the cumulative reported rate SeroStudy$_{\text{Rate}}$ estimated by serological studies [184, 182]. Note that both approaches try to fit the SeroStudy$_{\text{Rate}}$ without being informed with them. The results reveal that a large majority of COVID-19 infections were unreported. (**B**) MdlInfer reveals that non-pharmaceutical interventions (NPI) on asymptomatic and presymptomatic infections are essential to control the COVID-19 epidemic. Here, the red curve and other five curves represent the MdlInfer's estimation of reported infections for no NPI scenario and 5 different NPI scenarios described in section 6.3. The vertical grey dash line divides the observed period (left) and forecast period (right). (**C**) Inaccurate estimation by BaseInfer may lead to wrong non-pharmaceutical intervention conclusions. The blue curve and other five curves represent the BaseInfer's estimation of reported infections for no NPI scenario and the same 5 scenarios in (**B**).

decrease. Thus, we can conclude that non-pharmaceutical interventions on asymptomatic infections are essential in controlling the COVID-19 epidemic.

Accuracy of Non-pharmaceutical Intervention Simulations Relies on the Good Estimation of Parameterization

Next, we also plot the simulated reported infections generated by BaseInfer in Figure 6.5**C** (blue curve). As seen in the figure, based on BaseInfer, we can infer that only non-pharmaceutical interventions on symptomatic infections are enough to control the COVID-19 epidemic. However, this has been proven to be incorrect by prior studies and real-world observations [180]. Therefore, we can conclude that the accuracy of non-pharmaceutical intervention simulation relies on the quality of the learned parameter-

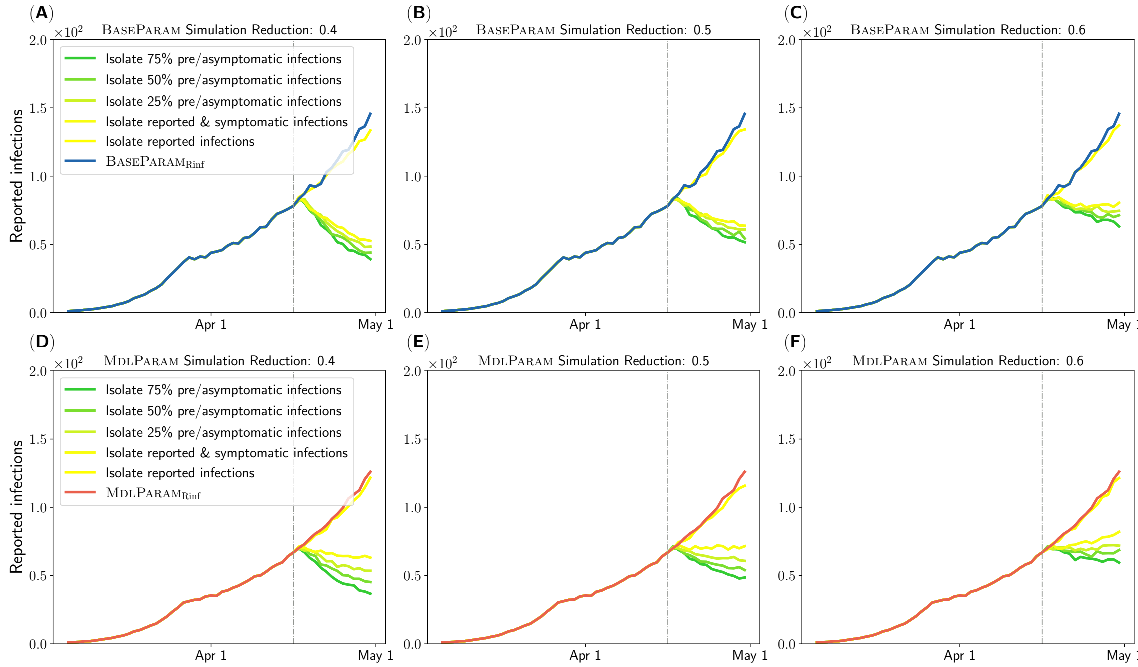


Figure 6.6: **Our non-pharmaceutical interventions simulation results are robust.** (**A**)-(**C**) The vertical grey dash line divides the observed period (left) and forecast period (right). The blue curve and other five curves represent the BASEINFER's estimation of reported infections for no NPI scenario and 5 different NPI scenarios described in section 6.3. Here, we reduce the infectivity of these isolated infections to 40% in (**A**), 50% in (**B**), and 60% in (**C**) in future period. (**D**)-(**F**) The red curve and other five curves represent the MDLINFER's estimation of reported infections for no NPI scenario and 5 different NPI scenarios described in section 6.3. Similarly, we reduce the infectivity of these isolated infections to 40% in (**D**), 50% in (**E**), and 60% in (**F**) in future period. The results are for Minneapolis-Spring-20 that we shown in Figure 6.5.

ization.

We also perform sensitivity experiments to inspect the robustness of our non-pharmaceutical interventions simulations for Minneapolis-Spring-20 in Figure 6.6. Here, we reduce the infectiousness of the isolated infections to 3 different values: 0.4, 0.5, and 0.6, and repeat simulations in each scenarios. Our results consistently show that only isolating reported or symptomatic infections is not enough to reduce the future reported infections. However, isolating both symptomatic infections and some fraction of asymptomatic and presymptomataic infections leads to reduction in reported infections in most settings.

6.4 Discussion

This study proposes MDLINFER, a data-driven model selection approach that automatically estimates the number of total infections using epidemiological models. Our approach leverages the information theoretic Minimum Description Length (MDL) principle to select total infections that "best describe" the observed outbreak. Our approach addresses several gaps in current practice including the long-term infeasibility of serological studies [182], and ad-hoc assumptions in epidemiological models [65, 20, 168, 64].

Overall, our results show that MDLINFER estimates total infections at various geographical locations and different epidemiological models more accurately than BASEINFER from both directions, i.e., it corrects both over- and under-estimates. For example, compared to BASEINFER, we correctly estimate 55719 more infections by April 1 for the $\mathrm{SEIR+HD}$ model in Figure 6.2**F**, and 87636 fewer infections for the $\mathrm{SAPHIRE}$ model Figure 6.2**B** for South Florida-Spring-20. We also show that MDLINFER leads to a better fit of the reported infections in the observed period and more accurate forecasts for the forecast period than BASEINFER. We reveal that a large majority of COVID-19 infections were unreported, where non-pharmaceutical interventions on unreported infections can help to mitigate the COVID-19 outbreak. We also show that MDLINFER estimates more accurate symptomatic rate trends than BASEINFER. Additionally, our results show consistent performance with respect to the reported infections and serological studies on both $\mathrm{SAPHIRE}$ and $\mathrm{SEIR+HD}$ model. We also show that MDLINFER identifies the ground truth parameters better than BASEINFER.

The MDLINFER framework is likely to be helpful in the surveillance of COVID-19 in the near future, and for future epidemics. Even with the U.S. returning to normalcy, surveillance of the pandemic is still essential for public health. The daily incidence of COVID-19 has decreased from early 2021 to summer 2021, according to the CDC COVID Data Tracker portal [197, 198]. However, new variants of the SARS-CoV-2 (e.g., the Delta and Omicron variants) have been spreading rapidly [199, 200, 201]. Testing for these new variants and large-scale surveillance via laboratory tests may be limited and less systematic than what was done for COVID-19 before. In such settings, using our MDLINFER framework, epidemiologists and policymakers can improve the accuracy of estimates of total infections (without large-scale serological studies), as well as forecasts of their models.

One of the limitations of our work is that the benefits of using MDLINFER depends on the suitability of the epidemiological model. If the epidemiological model is not expressive enough for the observed data, then the gains from MDLINFER may not be significant. As a future work, it may be helpful to adapt MDLINFER to measure the quality of an epidemiological model. We also note that MDLINFER is built on ODE-based epidemiological models; other kinds of epidemic models, e.g., agent-based models [202, 203, 204, 205, 206], are more suitable in some settings. It would be interesting to extend MDLINFER to incorporate such models. Finally, there is significant population or spatial heterogeneity in disease outcomes [207, 208], e.g., differences in severity rate or mortality rate, when infected with COVID-19, for different age groups [209, 210], which has not been considered in our work.

To summarize, MDLINFER is a robust data-driven method to accurately estimate total infections, which will help data scientists, epidemiologists, and policy-makers to further improve existing ODE-based epidemiological models, make accurate forecasts, and combat the ongoing COVID-19 pandemic. More generally, MDLINFER opens up a new line of research in epidemic modeling using information theory.

CHAPTER 7

END-TO-END STOCHASTIC OPTIMIZATION WITH ENERGY-BASED MODEL

Decision-focused learning (DFL) was recently proposed for stochastic optimization problems that involve unknown parameters. By integrating predictive modeling with an implicitly differentiable optimization layer, DFL has shown superior performance to the standard two-stage predict-then-optimize pipeline. However, most existing DFL methods are only applicable to convex problems or a subset of nonconvex problems that can be easily relaxed to convex ones. Further, they can be inefficient in training due to the requirement of solving and differentiating through the optimization problem in every training iteration. We propose SO-EBM, a general and efficient DFL method for stochastic optimization using energy-based models. Instead of relying on KKT conditions to induce an implicit optimization layer, SO-EBM explicitly parameterizes the original optimization problem using a differentiable optimization layer based on energy functions. To better approximate the optimization landscape, we propose a coupled training objective that uses a maximum likelihood loss to capture the optimum location and a distribution-based regularizer to capture the overall energy landscape. Finally, we propose an efficient training procedure for SO-EBM with a self-normalized importance sampler based on a Gaussian mixture proposal. We evaluate SO-EBM in three applications: power scheduling, COVID-19 resource allocation, and non-convex adversarial security game, demonstrating the effectiveness and efficiency of SO-EBM.

7.1 Introduction

Many real-life decision making tasks are stochastic optimization problems, where one needs to make decisions to minimize a cost function that involves stochastic parameters. Oftentimes, the involved stochastic parameters are *unknown* and *context-dependent*, meaning that they need to be predicted from observed features. For example, when allocating clinical resources for COVID-19, it is necessary to consider future cases in different regions, whose distributions are unknown and have to be predicted from the current state. As two other examples, hedge funds need to continuously adjust their portfolio for maximal expected return, by forecasting future return rates of different stocks; and in supply chain optimization, facility locations need to be decided to minimize long-term operational costs, by accounting for unknown and uncertain regional supplies and customer demands.

With the feasibility of training powerful deep learning predictors from large amounts of data, it is increasingly common to solve such stochastic optimization problems using a two-stage predict-*then*-optimize pipeline. In the prediction stage, one learns a predictive model for the unknown parameters using certain prediction loss (*e.g.*, negative likelihood). In the optimization stage, the predicted distributions of the parameters are used to parameterize the stochastic optimization problem, which can be then solved using off-the-shelf solvers [211, 212, 213]. This two-stage pipeline relies on an implicit and commonly-accepted assumption: *improvements in parameter prediction in terms of the prediction loss will always translate to better optimization outcomes*. However, this is not the case: machine learning models make errors and the impact of prediction errors is not uniform *w.r.t.* the optimization loss. Thus, a smaller predictive loss does not necessarily lead to a smaller decision regret.

A better approach is decision-focused learning (DFL) [214, 215, 216], which integrates prediction and optimization layers into a unified model to learn them end-to-end. Most DFL methods implement the optimization procedure as an implicitly differentiable layer and develop techniques (*e.g.*, using KKT conditions) to compute the gradients *w.r.t.* the decision variables and enable back-propagating through it. Compared to the two-stage pipeline, the prediction layer so learned is tailored for the optimization problem and better in the sense that it can yield decisions with smaller regrets. Besides DFL, there are also approaches where a policy network is trained to directly map from the input to the solution of the optimization problem using supervised or reinforcement learning [217, 218, 219]. However, their performance are often inferior to DFL as they ignore the algorithmic structure of the problem and typically require a large amount of data to rediscover the algorithmic structure.

Despite their promising results, existing DFL methods [214, 216, 215, 220, 221] suffer from two drawbacks. (1) *Generality*. To leverage the KKT condition, they are mostly applicable to only convex optimization objectives [214, 216, 215]. Though a few works approximate nonconvex objectives by quadratic functions [220, 221], their applicability is still limited to easy-to-relax nonconvex problems and can suffer from poor gradient estimates caused by the relaxation. (2) *Scalability*. Due to the reliance of using the KKT condition to compute derivatives, they require repeatedly solving the optimization program and back-propagating through it during training, which makes them unscalable. The problem is more severe when the expectation of the cost cannot be computed analytically.

We propose a new end-to-end stochastic optimization method using an energy-based model (EBM), named SO-EBM. Similar to DFL, SO-EBM models the algorithmic predict-*and*-optimize structure by stacking a differentiable optimization layer on top of a neural predictor. Different from DFL, SO-EBM eliminates the need of using KKT conditions to induce an implicit differentiable optimization layer. Instead, it leverages expressive EBMs [222] to directly model the conditional probability of the decisions and parameterizes an explicit energy-based differentiable layer as a surrogate to the original optimization problem. To better approximate the optimization landscape with the EBM surrogate, we design two complementary learning objectives in SO-EBM: 1) a local matching objective that maximizes the likelihood of the optimal decisions; and 2) a global matching objective that minimizes the distribution distance between the posterior distribution of the decision variables and the surrogate-based decision distribution.

Due to its flexibility, SO-EBM is not constrained to convex objectives, but can be applied to a wide class of stochastic optimization problems. Another key advantage of SO-EBM is its computational efficiency. As the optimization layer is parameterized by an energy-based model, SO-EBM eliminates the need of solving and differentiating through the optimization procedure at each training iteration. Rather, SO-EBM estimates the derivatives of the energy-based surrogate model using a self-normalized importance sampler. We design a Gaussian-mixture proposal distribution for the sampler, which not only reduces sampling variance, but also borrows the idea of contrastive divergence for MCMC methods to speed up the training of SO-EBM.

The main contributions of this work are: (1) We propose a new end-to-end stochastic optimization method based on energy-based model. It avoids the needs of solving and differentiating through the optimization problem during training and can be applied to a wide range of stochastic optimization problems beyond convex objectives. (2) We propose a coupled training objective to encourage the energy-based surrogate to well approximate the optimization landscape and an efficient self-normalized importance sampler based on a mixture-of-Gaussian proposal. (3) Experiments on power scheduling, COVID-19 resource allocation and adversarial network security game show that our method can achieve better or comparable performance than existing DFL methods while being much more efficient.

7.2 Methods

7.2.1 Preliminaries

Problem Formulation

Consider a stochastic optimization problem:

$$\arg\min_{a \in C} f(y, a),$$

where y denotes the parameters of the optimization problem, a denotes the decision variables within a feasible space C, and f is the cost function to be optimized. In many applications, the parameters y are *unknown* and *stochastic*, which must be inferred from some correlated features $\mathbf{x}$. We assume a dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$ drawn from the joint distribu-

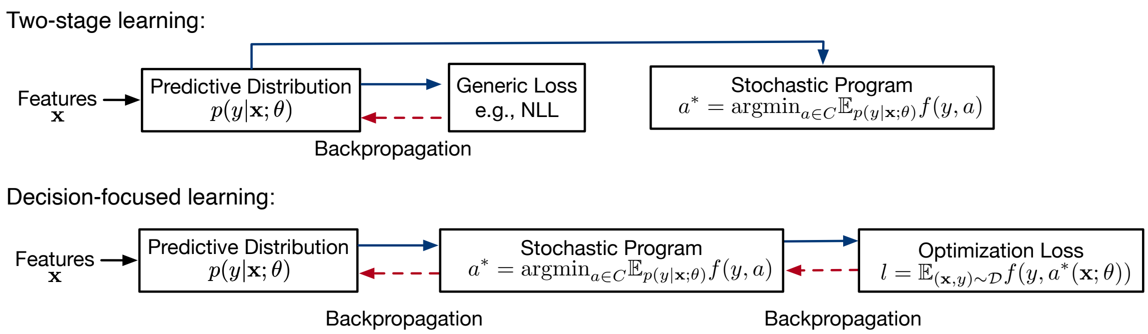


Figure 7.1: **Two-stage learning vs. decision-focused learning.** Two-stage learning back-propagates from a predictive loss to the model, ignoring the latter effect of the optimization problem. Decision-focused learning directly optimizes the task loss but needs to solve and back-propagate through the optimization problem at every training iteration.

tion of the features and problem parameters. Our task is to learn a decision-making model $\mathbb{M}$ parameterized by θ, which takes the features $\mathbf{x}$ as input and outputs the optimal decisions $a^*(\mathbf{x};\theta)$. The model should be learned such that its output optimal decisions minimize the expected decision cost under the joint distribution of $(\mathbf{x}, y)$, namely:

$$\arg\min_{\theta} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} f(y, a^*(\mathbf{x};\theta)).$$

For example, in supply chain optimization, y can be regional customer demands, a can be pre-ordered products for each region, f measures the gap between actual customer demands and pre-ordered supplies, and the problem is to decide the best a to minimize the cost f. Herein, the actual customer demands y are usually unknown and must be predicted from features $\mathbf{x}$ such as customer purchase history and regional economy indices.

Two-stage learning v.s. Decision Focused Learning (DFL).

A common practice to the above stochastic optimization problem is the two-stage predict-then-optimize framework. It first learns an probabilistic predictive model $p(y|\mathbf{x};\theta)$ and then uses existing stochastic optimization solvers to obtain the optimal action that minimizes the expected cost: $a^*(\mathbf{x};\theta) = \arg\min_{a\in C} \mathbb{E}_{y\sim p(y|\mathbf{x};\theta)} f(y,a)$. Though the two-stage approach is simple and efficient, it can suffer from suboptimal performance due to the misalignment of the prediction loss and the optimization loss. In contrast, DFL integrates prediction and optimization into an end-to-end model, thus tailoring the predictive model for the optimization task, as shown in Figure 7.1. By directly optimizing the task loss, the gradient of the model parameters can be obtained with the following chain rule:

$$\frac{\partial f(y, a^*(\mathbf{x};\theta))}{\partial \theta} = \frac{\partial f(y, a^*(\mathbf{x};\theta))}{\partial a^*(\mathbf{x};\theta)} \frac{\partial a^*(\mathbf{x};\theta)}{\partial y} \frac{\partial y}{\partial \theta}.$$

The key challenge here is to compute the Jacobian $\frac{\partial a^*(\mathbf{x};\theta)}{\partial y}$: it is needed to apply the chain rule to learn the model using gradient decent methods. This is nontrivial, because a^* is the solution of a stochastic optimization solver and not directly differentiable. To address this challenge, OptNet [214, 216] assumes quadratic optimization objectives and differentiates through the KKT conditions using the implicit function theorem. This way, OptNet can obtain the Jacobian by solving the optimization problem along with a set of linear equations in each training iteration. Several works [215, 220, 221] extend this technique to more general cases. For example, cvxpylayers [215] extends it to more general cases of convex optimization using disciplined parametrized programming (DPP) grammar.

Although existing DFL approaches can achieve better decisions compared to two-stage learning, they have several drawbacks. First, they are often constrained to convex optimization objectives as they rely on the KKT conditions to compute derivatives. Though [220, 221] propose to approximate some non-convex objectives by a quadratic function around a local minimum, the inaccurate gradients may be aggregated during the training iterations and thus lead to poor decision quality. Second, they suffer from high computational complexity because the computation of the Jacobian $\frac{\partial a^*(\mathbf{x};\theta)}{\partial y}$ requires repeatedly solving the optimization program and back-propagating through it. The problem is more severe when the expectation of the cost cannot be computed analytically. In that case, we need to use sample average approximation [223, 224, 225] and draw multiple IID samples from the predictive distribution, which makes the objective much more complicated and expensive.

7.2.2 Energy-based Model for End-to-end Stochastic Programming

Our task is to learn an optimal decision-making model $\mathbf{M}_\theta$, such that its output optimal decision $a^*(\mathbf{x};\theta)$ for input $\mathbf{x}$ minimizes the expected optimization cost: $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}} f(y, a^*(\mathbf{x};\theta))$. The core idea of our method is to directly model $\mathbf{M}_\theta$'s probability distribution over the decisions conditioned on the features, denoted as $q(a|\mathbf{x};\theta)$, using energy-based parameterization [222]:

$$q(a|\mathbf{x};\theta) = \frac{\exp(-E(\mathbf{x}, a; \theta))}{Z(\mathbf{x}, \theta)}, \quad Z(\mathbf{x};\theta) = \int \exp(-E(\mathbf{x}, a; \theta) da. \tag{7.1}$$

To parameterize the energy function $E(\mathbf{x}, a; \theta)$, a natural option is to directly use a deep neural network which takes a feature-decision pair $(\mathbf{x}, a)$ as input and output a scalar value. However, such a design ignores the algorithmic structure of the optimization problem and thus can be data-inefficient during learning. Instead, we propose to explicitly model the problem structure by using the expected task loss as the energy function:

$$E(\mathbf{x}, a; \theta) = \mathbb{E}_{p(y|\mathbf{x};\theta)} f(y, a), \tag{7.2}$$

where $p(y|\mathbf{x};\theta)$ is the predictive distribution of an uncertainty-aware neural network [226, 227].

Equation 7.2 creates an explicit energy-based differentiable layer as a surrogate to the original optimization problem. Compared with the two-stage model, it builds a direct connection between the input features $\mathbf{x}$ and decision variable a and thus is more tailored to the downstream task. Compared with pure end-to-end architectures, the task-based surrogate energy function explicitly leverages the algorithmic structure of the optimization problem and thus saves a lot of learning. This energy-based surrogate function also has a intuitive interpretation. When the decision a is in a region with smaller expected task loss, it has lower energy and higher probability; when it leads to higher expected task loss, it has higher energy and lower probability.

Since we have the ground truth for the parameters y in the training data, the feature-decision pairs $\mathcal{D}_a = \{(\mathbf{x}_i, a_i^*)\}_{i=1}^N$ can be easily constructed by solving $a_i^* = \arg\min_{a_i \in C} f(y_i, a_i)$ for each $(\mathbf{x}_i, y_i)$ using any off-the-shelf optimization solvers. Note that the construction of such feature-action training pairs needs to be done only once during preprocessing (Figure 7.3). Then, we minimize the negative log-likelihood (NLL) of all the feature-decision pairs:

$$\ell_{\text{MLE}} = -\mathbb{E}_{(\mathbf{x},a^*)\sim\mathcal{D}_a} q(a^*|\mathbf{x};\theta) = \mathbb{E}_{(\mathbf{x},a^*)\sim\mathcal{D}_a} \left(\mathbb{E}_{p(y|\mathbf{x};\theta)} f(y, a^*) + \log(Z(\mathbf{x};\theta))\right). \quad (7.3)$$

By minimizing the NLL, we are essentially minimizing the energy of the optimal actions while maximizing the energy of other points. Thus the end-to-end stochastic programming problem is translated to learning a neural network that outputs the smallest energy for the optimal actions. This new perspective avoids the need of solving and differentiating through the optimization problem at every training iteration, but meanwhile tailors the model for the downstream decision making task.

Our energy-based formulation can also be interpreted as a non-sequential maximum entropy inverse reinforcement learning (MaxEN-IRL) model [228, 229]. From the IRL perspective, the input features $\{\mathbf{x}_i\}_{i=1}^N$ can be interpreted as environment states, the feature-decision pairs $\{(\mathbf{x}_i, a_i)\}_{i=1}^N$ as expert demonstrations, and the negative expected task loss $-\mathbb{E}_{p(y|\mathbf{x};\theta)} f(y, a)$ as reward. Equation 7.3 is then equivalent to maximizing the likelihood of the expert demonstrations (optimal decisions) to recover the hidden reward function.

7.2.3 Augmenting Energy-Based Objective with Distribution Regularizer

One limitation of learning with EBM-based likelihood for the optimization problem is that we model only one data point a^* for each conditional distribution $q(a|\mathbf{x};\theta)$. Minimizing the NLL for a single data point ignores the overall matchness between the landscape of

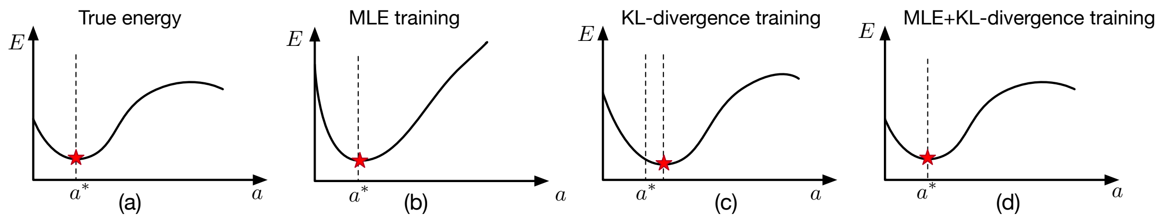


Figure 7.2: **Different training objectives of energy-based optimization.** (a) the ground-truth energy-based landscape; (b) MLE training captures the location of the optimum but ignores the overall energy shape; (c) KL-divergence training learns the overall energy shape but captures a blurry optimum location; (d) Our MLE+KL-divergence training wins the best of both worlds.

original optimization problem and that of the EBM-based probability density, which can easily cause overfitting. To learn the overall shape of the energy function better, we propose a distribution-based regularizer which augments the energy-based objective from a global training perspective.

The distribution-based regularizer is based on minimizing the distance between the model distribution and an oracle posterior distribution. Specifically, we assume that a follows a prior distribution $p(a)$. With the ground-truth label y, the posterior distribution of a is then given by $p(a|y) = \frac{p(a)\exp(-f(y,a))}{Z(y)}$ when we define the task loss as the negative log-likelihood, *i.e.*,$-\log p(y|a) = f(y,a)$. However, such an expression depends on the ground-truth label, which cannot be obtained at test time. We let the model distribution $q(a|\mathbf{x};\theta)$ mimic this oracle posterior $p(a|y)$ distribution by minimizing their KL-divergence:

$$\ell_{\text{KL}} = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\text{KL}[p(a|y)||q(a|\mathbf{x};\theta)] = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left(-\mathbb{E}_{p(a|y)}\log q(a|\mathbf{x};\theta) - \mathcal{H}(p(a|y))\right), \tag{7.4}$$

where $\mathcal{H}(\cdot)$ denotes entropy of the probability distribution. Our final training loss is a weighted combination of the NLL and the distribution based regularizer:

$$\ell_{\text{Total}} = \ell_{\text{MLE}} + \lambda\ell_{\text{KL}}, \tag{7.5}$$

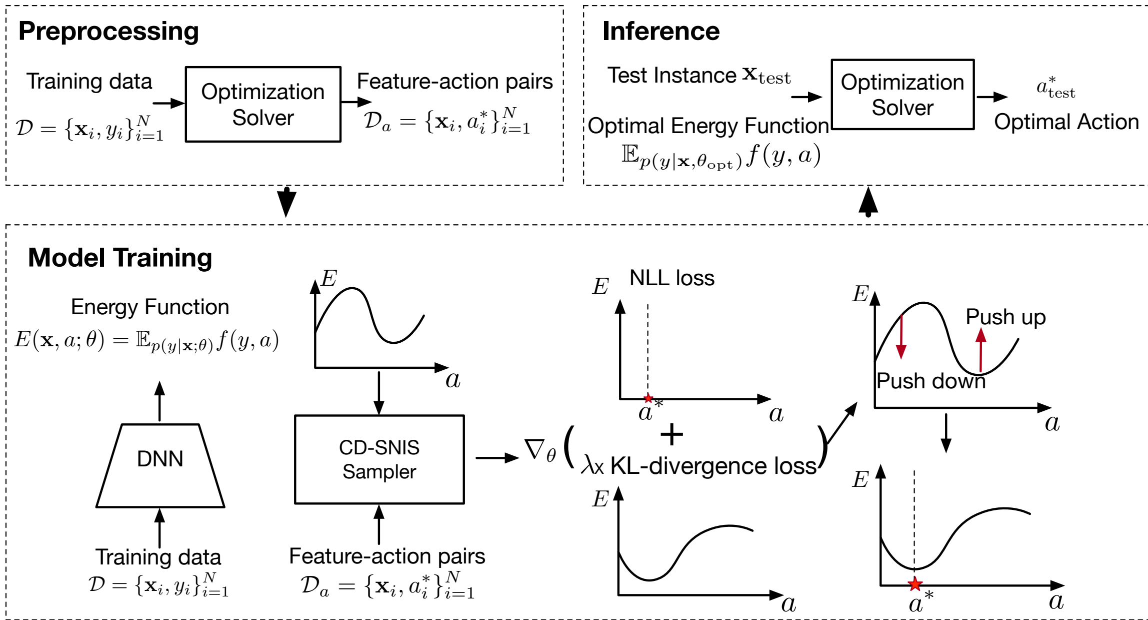


Figure 7.3: **Illustration of SO-EBM.** End-to-end stochastic optimization with energy-based model.

where λ is a hyper-parameter. In this final objective, the KL-divergence loss and the NLL loss complement each other (Figure 7.2). The NLL loss is a local training strategy while the KL-divergence loss learns the energy function from a global level. Specifically, the NLL loss helps the model capture the location of the optimal decision better but ignores the overall energy shape. The distribution-based regularizer essentially adds more anchor points besides the optimal decisions and thus helps fit the overall energy shape.

7.2.4 Training with Contrastive Divergence and Self-normalized Importance Sampling

Optimizing Equation 7.5 requires evaluating the partition function and computing the KL-divergence between two continuous distributions which typically involves intractable integrals. In this subsection, we propose to use a self-normalized importance sampler based on mixture of Gaussians to estimate the gradient of the model parameters efficiently. First, we derive the gradient of the training loss with respect to the model parameters θ as:

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{Total}}}{\partial \theta} =& \mathbb{E}_{(\mathbf{x},a^*)\sim\mathcal{D}_a}\left(\frac{\partial E(a^*,\mathbf{x};\theta)}{\partial \theta} - \mathbb{E}_{q(\tilde{a}|\mathbf{x};\theta)}\frac{\partial E(\tilde{a},\mathbf{x};\theta)}{\partial \theta}\right) \\ &+ \lambda \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left(\mathbb{E}_{p(\hat{a}|y)}\frac{\partial E(\hat{a},\mathbf{x};\theta)}{\partial \theta} - \mathbb{E}_{q(\tilde{a}|\mathbf{x};\theta)}\frac{\partial E(\tilde{a},\mathbf{x};\theta)}{\partial \theta}\right).\end{aligned} \tag{7.6}$$

As can be seen, the gradient can be estimated by sampling from the model distribution $q(a|\mathbf{x};\theta)$ and oracle distribution $p(a|y)$. Unfortunately, we cannot easily draw samples because of the unnormalized constant. Existing methods usually resort to MCMC methods (*e.g.*, Langevin dynamics) to use this gradient estimator. However, MCMC is an iterative process and can be time consuming. To improve the training efficiency, we propose to use a self-normalized importance sampler based on a Gaussian mixture proposal to estimate the gradient.

Specifically, for each $\mathbf{x}$, we first sample a set of M candidates $\{a^m\}_{m=1}^M$ from a proposal distribution $\pi(a|\mathbf{x})$, and then sample $\tilde{a}$ from the empirical distribution located at each a^m and weighted proportionally to $\exp(-E(a|\mathbf{x};\theta))/\pi(a|\mathbf{x})$. To reduce the variance of the self-normalized importance sampler, we propose to use a mixture of K Gaussians which are centered at the location of the corresponding optimal decision as the proposal distribution: $\pi(a|\mathbf{x}) = \frac{1}{K}\sum_{i=1}^K \mathcal{N}(a^*;\sigma_k)$, where K and $\{\sigma_k\}_{k=1}^K$ are hyper-parameters. This mixture of Gaussian based self-normalized importance sampler enjoys great computational efficiency. We only need to estimate the gradient of the energy-based surrogate layer by drawing samples from a simple mixture of Gaussians, instead of solving and differentiating through the optimization problem as in DFL. This significantly reduces the training time compared with DFL as shown in section 7.3. Further, locating the proposal distribution at the optimal decisions mimics the contrastive divergence method [230, 231] used in Langevin dynamics where the MCMC chain starts from the training data. This makes the proposal distribution close to the model distribution and thus has better sample efficiency. The sampling procedure from the oracle distribution $p(a|y)$ is similar with $q(a|\mathbf{x};\theta)$.

Algorithm 7 Training of SO-EBM

Require: Training data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, task loss function $f(\mathbf{x}, y, a)$, model $\mathbf{M}$ parameterized with parameter θ

Construct feature-decision pairs $\mathcal{D}_a = \{(\mathbf{x}_i, a_i^*)\}$ by solving $a^* = \arg\min_{a \in C} f(\mathbf{x}_i, y_i, a)$ for each $(\mathbf{x}_i, y_i)$ in $\mathcal{D}$
$E(\mathbf{x}, a; \theta) \triangleq -\mathbb{E}_{p(y|\mathbf{x};\theta)} f(\mathbf{x}, y, a)$
for # training iterations **do**
  Sample a mini-batch $\mathcal{B}_y$ from $\mathcal{D}$ and $\mathcal{B}_a$ from $\mathcal{D}_a$
  for For each $(\mathbf{x}_i, a_i^*)$ and $(\mathbf{x}_i, y_i)$ in $\mathcal{B}_a$ and $\mathcal{B}_y$ **do** ▷ CD-SNIS Sampler
    $\pi(a; \mathbf{x}_i) \triangleq \frac{1}{K} \sum_{k=1}^{K} \mathcal{N}(a_i^*; \sigma^k)$
    for $j = 1, \cdots, J$ **do**
      Sample $a_i^j \sim \pi(a; \mathbf{x}_i)$
      Compute $\tilde{w}(a_i^j) = \exp(-E(\mathbf{x}_i, a_i^j; \theta)) / \pi(a_i^j; \mathbf{x}_i)$
      Compute $\hat{w}(a_i^j) = \exp(-f(y_i, a_i^*)) / \pi(a_i^j; \mathbf{x}_i)$
    end for
    Compute $\tilde{Z} = \sum_{j=1}^{J} w(a_i^j)$, $\hat{Z} = \sum_{j=1}^{J} \hat{w}(a_i^j)$
    for $m = 1, \cdots, M$ **do**
      Sample $j_1 \sim \text{Categorical}(w(a_i^1)/\tilde{Z}, \cdots, w(a_i^J)/\tilde{Z})$
      Sample $j_2 \sim \text{Categorical}(w(a_i^1)/\hat{Z}, \cdots, w(a_i^J)/\hat{Z})$
      $\tilde{a}_i^m = a_i^{j_1}$, $\hat{a}_i^m = a_i^{j_2}$
    end for
  end for
  Compute $\frac{\partial \mathcal{L}_{\text{Total}}}{\partial \theta}$ via Equation 7.6 using $\mathcal{B}_y$, $\mathcal{B}_a$, $q(\tilde{a}|\mathbf{x}_i; \theta) = \frac{1}{M} \delta_{\tilde{a}_i^m}(\tilde{a})$ and $p(\hat{a}|y) = \frac{1}{M} \delta_{\hat{a}_i^m}(\hat{a})$
  Update θ using ADAM
end for

When the expected task loss has no analytical expression, we can draw multiple samples from $p(y|\mathbf{x}; \theta)$ to estimate the expectation and use reparameterization trick [232, 233, 234, 235] to make it differentiable. Finally, the model can be efficiently trained via gradient-based method, such as Adam [236]. The detailed training procedure is given in Algorithm 7.

Model inference

With the optimal model parameter θ_{opt}, we draw samples from $q(a|\mathbf{x}_{\text{test}}; \theta_{\text{opt}})$ for an unseen test instance $\mathbf{x}_{\text{test}}$. However, in the real-world applications, we usually only need the most optimal decision. This can be obtained by solving $\arg\min_{a^*_{\text{test}} \in C} \mathbb{E}_{p(y|\mathbf{x}_{\text{test}}; \theta_{\text{opt}})} f(y, a)$ with any existing black-box optimization solver such as CVXPY [213, 212] and Pyomo [237, 211] (Figure 7.3).

7.3 Results

In this section, we empirically evaluate SO-EBM. We conduct experiments in three applications: (1) Load forecasting and generator scheduling where the expected task loss has a closed-form expression; (2) Resource allocation for COVID-19 where the expected task loss has no closed-form expression; (3) Adversarial behavior learning in network security with a non-convex optimization objective. Finally, we do ablation studies to show the effect of each component in SO-EBM.

7.3.1 Load Forecasting and Generator Scheduling

In this task, a power system operator needs to decide how much electricity $a \in \mathbb{R}^{24}$ to schedule for each hour in the next 24 hours to meet the actual electricity demands. The optimization objective is a combination of an under-generation penalty, an over-generation penalty, and a mean squared loss between supplies and demands:

$$\begin{aligned} &\text{minimize}_{a\in\mathbb{R}^{24}} \sum_{i=1}^{24} \mathbf{E}_{y\sim p(y|x;\theta)}[\gamma_s[y_i - a_i]_+ + \gamma_e[a_i - y_i]_+ + \frac{1}{2}(a_i - y_i)^2] \\ &\quad\text{subject to} \quad |a_i - a_{i-1}| \leq c_r \quad \forall i, \end{aligned} \tag{7.7}$$

where $[\cdot]_+ = \max\{v, 0\}$. Usually, the penalty coefficients satisfy $\gamma_e \gg \gamma_s$ since under-generation is more serious than over-generation. The quadratic regularization term indicates the preference for generation schedules that closely match actual demands. The ramping constraint c_r restricts the change in generation between consecutive timepoints, which reflects physical limitations associated with quick changes in electricity output levels.

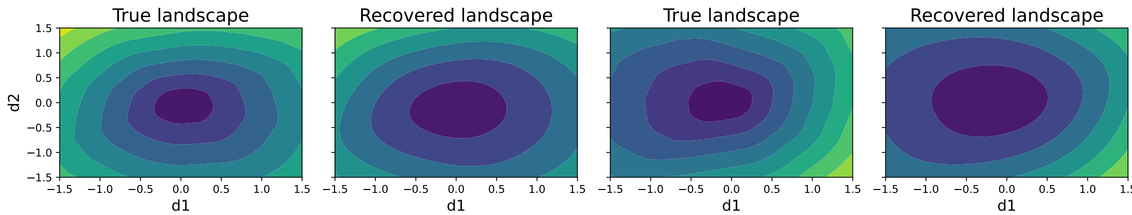


Figure 7.4: **Ground-truth and SO-EBM recovered landscapes of the energy function in the power generator scheduling task.** Darker colors represent lower energy in the heat maps. For a test sample $\mathbf{x}_{\text{test}}$, we choose the corresponding optimal action a^*_{test} as the center point and select two random direction vectors v_1 and v_2 to plot the energy landscape, *i.e.*,$E^{'}(d_1, d_2) = E(\mathbf{x}, a^* + d_1 v_1 + d_2 v_2)$.

Experiment Setup

We forecast the electricity demands $y \in \mathbb{R}^{24}$ over all 24 hours of the next day using a 2-hidden layer neural network. We assume y_i is a Gaussian with mean μ_i and variance σ_i^2; as such, the expectation in the optimization objective can be computed analytically. The input features $\mathbf{x}$ is a 150-dimensional vector including the past day's electrical load and temperature, the next day's temperature forecast, non-linear functions of the temperatures, binary indicators of weekends or holidays, and yearly sinusoidal features. Following [214], we set $\gamma_s = 0.4$, $\gamma_e = 50$ and $c = 0.4$.

We compare with the following baselines on this task: (1) A two-stage predict-then-optimize model trained with negative likelihood loss for the prediction task. (2) Decision-focused learning with the QPTH solver [214]. It uses sequential quadratic programming (SQP) to iteratively approximate the resultant convex objective as a quadratic objective, iterates until convergence, and then computes the necessary Jacobians using the quadratic approximation at the solution. (3) DFL with cvxpylayers [215] (DFL-CVX) which provides a differentiable layer with disciplined parameterized programming (DPP) grammar. Since the analytical expectation of Equation 7.7 cannot be written in DPP, we use Monte Carlo sampling to estimate the expectation for this baseline. (4) Policy-net. It direct maps from the input features to the decision variables by minimizing the task loss using supervised

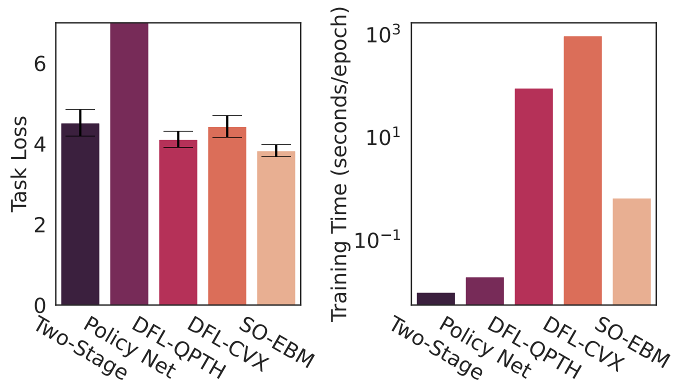


Figure 7.5: **Results on power generator scheduling.**

learning [214].

Results

Figure 7.5 shows the end task loss for all the methods. Our method SO-EBM outperforms all the baselines with a significant reduction of training time. Compared with the strongest baseline DFL-QPTH, SO-EBM improves the task loss by 7.3%. The improvement is because DFL-QPTH needs to use SQP to iteratively obtain the solutions for non-quadratic optimization problems. Differentiating through all the steps of SQP is prohibitively expensive in memory and time. To address this issue, existing works [214] differentiate through just the last step of SQP to obtain approximate gradients. The inaccurate gradients may accrue during training and thus hurt decision quality. This shows that our explicit differentiable energy-based function is an effective surrogate for the original implicit optimization layer. In term of efficiency, SO-EBM is more than $\mathbf{136}$ times faster than DFL-QPTH (0.68 second/epoch *v.s.* 93.12 second/epoch) in training. This is because we only need to draw samples from a simple mixture of Gaussians to estimate the model parameters instead of solving and differentiating through the optimization problem at every training iteration. DFL-CVX is even slower than DFL-QPTH since it needs to use sample average estimation to draw multiple samples from $p(y|\mathbf{x};\theta)$, which results in a much more complicated optimization objective. This is also likely the reason why DFL-CVX underperforms DFL-

QPTH in terms of task loss here. For Policy-Net, we have tuned its hyper-parameters extensively but still cannot achieve good performance on this task (2x-3x larger task loss than the two-stage baseline). This is not surprising: as aforementioned, Policy-Net is a pure end-to-end model that needs a large amount of data to rediscover the algorithmic structure of the optimization task. In contrast, DFL and SO-EBM model the predict-and-optimize structure in the model design, which save learning and are more data-efficient.

Figure 7.4 shows the ground-truth and SO-EBM learned landscapes of the energy function. As we can see, SO-EBM can recover the landscape of the original optimization objective effectively though with a small discrepancy. The small discrepancy is expected since the ground-truth landscape is computed by directly using the ground-truth label, while SO-EBM uses the uncertainty-aware neural network to first forecast the distribution of the label and then uses the predictive distribution to compute the expected task loss as the energy function.

7.3.2 Resource Allocation for COVID-19

As seen in the COVID-19 pandemic, an increasing number of infected patients leads to increasing demand for medical resources; which makes it challenging for policymakers and epidemiologists to plan ahead. Mechanistic epidemiological models based on Ordinary Differential Equations (ODEs) are often used to capture and forecast the dynamics of epidemics including for the COVID-19 pandemic [20, 65, 64, 33]. Such forecasts are typically then used as guidance to help plan for future resource allocation [238, 239, 128, 240]. In this task, we study the optimization problem of hospital bed preparation for COVID-19, one of the most common and important tasks epidemiologists focus on during the pandemic [241, 129]. We need to decide how many beds $a \in \mathbb{R}^7$ are needed to prepare for the next week based on the forecasted number of hospitalized patients $y \in \mathbb{R}^7$. The optimization objective is a combination of linear and quadratic costs, which accounts for over-preparations $[a - y]_+$ and under-preparations $[y - a]_+$ in the next 7 days over ODE-derived

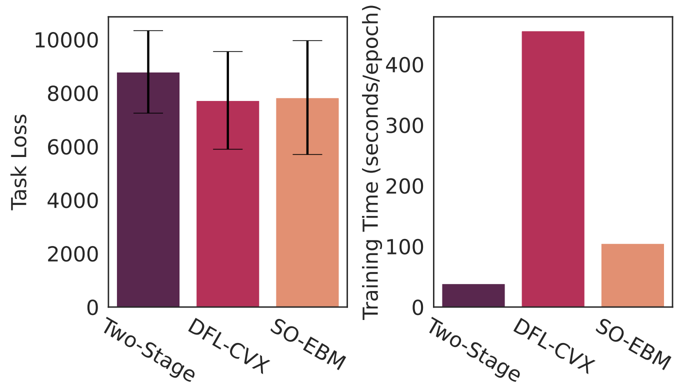


Figure 7.6: **Results on Covid-19 resource allocation.**

dynamics:

$$\text{minimize}_{a\in\mathbb{R}^7} \sum_{i=1}^{7} c_b[y_i - a_i]_+ + c_h[a_i - y_i]_+ + q_b([y_i - a_i])_+^2 + q_h([a_i - y_i])_+^2. \qquad (7.8)$$

Experiment Setup

We use the SEIR+HD ODE model proposed in [65] to capture the dynamics of the COVID-19 pandemic, which has been used in policy studies for interventions. The model is driven by a key parameter: the transmission rate β that reflects the probability of disease transmission. The forecasting model is a two-layer gated recurrent unit (GRU) [242] which takes the number of people in each state of the SEIR+HD model in the last 21 days as input features and outputs the transmission rate β for the next 7 days. We assume that β follows a Gaussian distribution. Since SEIR-HD is a complex non-linear ODE system, there is no closed expression for the distribution of the number of hospitalized patients y. Hence, we choose to sample from the forecasted distribution of β 100 times and then simulate the SEIR-HD model to obtain the empirical distribution of the number of hospitalized patients y.

Results

Figure 7.6 shows the results on the resource allocation for COVID-19 task. Both SO-EBM and DFL significantly outperforms the two-stage model in terms of the task loss, yielding $10.9\%+$ improvement. DFL-QPTH is not included here because it needs to use SQP to iteratively approximate the objective and is much slower than DFL-CVX in this task. The performance of SO-EBM and DFL are similar, however, the training time of SO-EBM is 4.3 times faster than DFL. The time improvement of SO-EBM is not as large as on the energy scheduling task because all the methods need to compute the complex ODE equations in the forward pass which is a time consuming part. Therefore, considering the overall training efficiency, we argue that SO-EBM is a better option than DFL on this task in practice.

7.3.3 Network Security Game

In this task, we study stochastic optimization in network security games [243]. Given a network $G = (V, E)$, a source node $s \in V$, and a set of target nodes $T \in V$, a *network security game* (NSG) [244, 245, 246] defines the min-max game where the attacker attempts to travel from s to any $t \in T$ while the defender places checkpoints on certain number of edges in the graph. Each of the target node has a reward $u(t)$ should the attacker reach them. The defender first chooses a mixed strategy. Having observed the defender's mixed strategy but not the sampled pure strategy, the attacker attempts to choose a path. To minimize the attacker's scores, the defender can try to predict the attacker's path decisions with node features and their past decisions, since the attacker is not perfectly rational in reality. Suppose that $\mathbf{a}$ defines the placement of checkpoints by the defender where $\mathbf{a}_e$ is the probability that edge e is covered. The attackers will perform a random walk on the graph, generating a path r, stopping at either a target or a checkpoint. The transition between two nodes is determined by the defender's strategy $\mathbf{a}$ and node-level parameters $\mathbf{y}$, where $\mathbf{y}_u$ defines the "attractiveness" of node u, representing the attacker's idiosyncratic preferences. The defender wants to choose $\mathbf{a}$ to maximize its expected utility: $\max_{\mathbf{a}} \mathbb{E}_{T\sim\mathbf{a}}\mathbb{E}_{r\sim p(r|\mathbf{a},\mathbf{y})} - g(r)$,

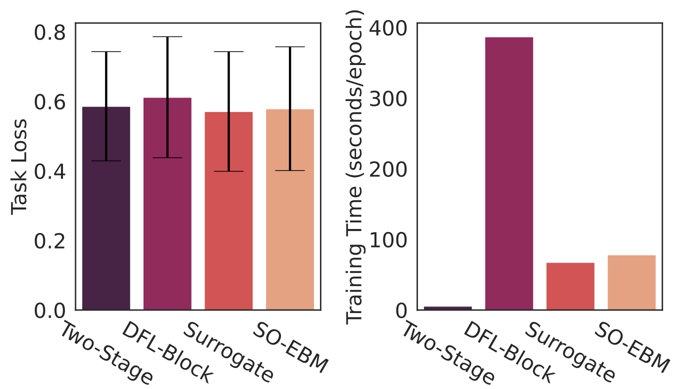


Figure 7.7: **Results on network security game.**

where T is covered edges sampled according to $\mathbf{a}$, $g(r) = u(t)$ if the attacker reaches target t with path r and 0 if the attacker is stopped at a checkpoint. The defender's objective comes with the constraint $\sum \mathbf{a} \leq k$, where $k = 3$ represents the resources available to the defender. The optimization objective is non-convex due to the irrational strategy of the attacker [221].

Experiment Setup

We follow the setup of [243].

Results

Figure 7.7 shows the results on the adversarial network security game task. Since the original DFL fails on this large scale problem, we compare a block-variable sampling approach specialized to this task (DFL-Block)[221]. Surrogate [243] performs a linear dimension reduction for DFL to speed up the training time and improve the performance by smoothing the training landscape. We did not present the results of Surrogate in the previous two tasks because the numbers of decision variables there are relatively small and it has even worse performance than the original DFL approach. As we can see, SO-EBM can achieve competitive task loss with Surrogate and outperforms DFL-block and the two-stage method. The increase of training time in our method is mainly because every evaluation of the de-

Table 7.1: Ablation study on power generator scheduling.

Method	Task loss
Two-stage	4.52 ± 0.33
So-Ebm w/o MLE	4.31 ± 0.17
So-Ebm w/o KLD	3.89 ± 0.18
So-Ebm w/ CD-LD	3.85 ± 0.13
So-Ebm	$\mathbf{3.83 \pm 0.15}$

fender's utility requires a matrix inverse; our method needs to evaluate the utility for all the samples drawn from the proposal distribution which results a larger matrix to inverse. However, this issue can be mitigated by using some advanced matrix inverse algorithm [247, 248]. DFL-Block is even worse than the two-stage model because it back-propagates through randomly sampled variables which results in inaccruate gradient estimation.

7.3.4 Ablation Study

We investigate the effectiveness of the coupled training objective and alternative EBM training algorithms via ablation studies on the load forecasting and generator scheduling task. Table 7.1 shows the results. Our findings can be summarized as follows: (1) Both the MLE and KLD training objectives work and outperform the two-stage model. This verifies the effectiveness of using the energy-based surrogate function to approximate the optimization landscape. (2) By dropping either the MLE or KLD term, we observe a performance degradation in our method. Specifically, without the MLE term, the task loss increases by 13%; without KLD term, the task loss increases by 2%. The larger degradation when simply minimizing the KLD term is possibly because accurately recover the entire optimization landscape is too difficult and it needs the MLE term to help capture the location of the optimal decision. (3) Training So-Ebm with Contrastive Divergence based Langevin Dynamics (CD-LD) [231] achieves similar performance but incurs longer training time (1.47 second/epoch *v.s.* 0.68 second/epoch). This phenomenon is likely because the optimization problem is relatively low-dimensional, thus importance sampling works

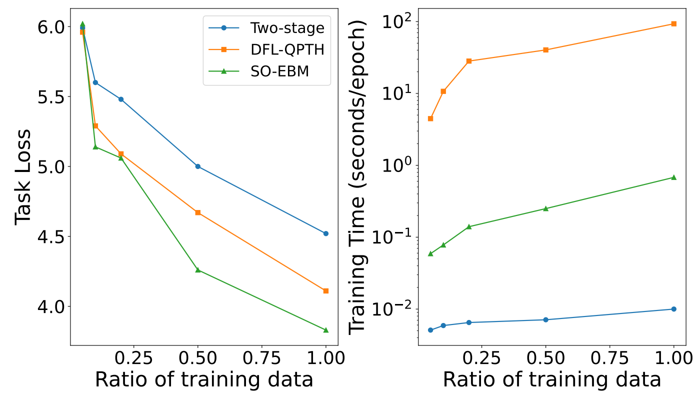


Figure 7.8: **Task losses and training time under different ratios of training data on power generator scheduling.**

well and enjoys better efficiency in this regime. However, even with CD-LD, SO-EBM is still 63 times faster than DFL-QPTH (1.47 second/epoch *v.s.* 93.12 second/epoch).

We also investigate the impact of training data size for each method. Figure 7.8 provides the task losses and training time under different ratios of training data on the load forecasting and generator scheduling task. As we can see, our method outperforms the baselines constantly except for the ratio of 0.05. When the ratio is below 0.05, all the methods cannot work properly due to the extremely low resource. The superior performance of our method is because we design the energy function as the expected task loss, which leverages the algorithmic structure inherent in the optimization problem. Hence, our method is not very data demanding. In terms of efficiency, our method reduces the training time significantly compared with DFL-QPTH across different amounts of training data.

7.4 Discussion

We focused on addressing the scalability and generality of existing decision-focused learning (DFL) for end-to-end stochastic optimization. We argue that these deficiencies stem from the reliance of implicitly differentiable optimization layers based on KKT conditions. As a remedy, we circumvent such deficiencies by replacing the implicit optimization layer with a newly parameterized energy-based surrogate function. We proposed a coupled training objective to encourage the energy-based surrogate well approximate the optimization landscape, as well as an efficient training procedure based on self-normalized importance sampling. Empirically, we demonstrated that our energy-based model is effective in a wide range of stochastic optimization problems with either convex or nonconvex objectives. It can achieve better or comparable performance than state-of-the-art DFL methods for stochastic optimization, while being several times or even orders of magnitude faster.

We discuss limitations and possible extensions of SO-EBM: (1) *Handling more complex constraints*. Our method handles the constraints implicitly through the pre-processing step and explicitly through the inference step. However, when the feasible space is extremely small, training the constrained EBM may have more benefits. To train EBMs in the constrained space, one direction is to project samples into the feasibility space. Another direction is to explore adding soft constraints to the energy function during training, *e.g.*, Augmented Lagrangian penalty and Barrier penalty. (2) *More effective training methods*. Our method is a general framework for end-to-end stochastic programming problem based on EBM. There are a number of training techniques [249] that can be plugged into our framework and we can further improve the task performance using the advanced EBM training algorithms [250, 251].

CHAPTER 8

IDENTIFYING IMPORTATION AND ASYMPTOMATIC SPREADERS OF MULTI-DRUG RESISTANT ORGANISMS IN HOSPITAL SETTINGS

Healthcare-associated infections (HAIs) due to multi-drug resistant organisms (MDROs) are a significant burden to the healthcare system. Patients are sometimes already infected at the time of admission to the hospital (referred to as "importation"), and additional patients might get infected in the hospital through transmission ("nosocomial infection"). Since many of these importation and nosocomial infection cases may present no symptoms (i.e., "asymptomatic"), rapidly identifying them is difficult since testing is limited and incurs significant delays. Although there has been a lot of work on examining the utility of both mathematical models of transmission and machine learning for identifying patients at risk of MDRO infections in recent years, these methods have limited performance and suffer from different drawbacks: Transmission modeling-based methods do not make full use of rich data contained in electronic health records (EHR), while machine learning-based methods typically lack information about mechanistic processes. In this work, we propose NeurABM, a new framework which integrates both neural networks and agent-based models (ABM) to combine the advantages of both modeling-based and machine learning-based methods. NeurABM simultaneously learns a neural network model for patient-level prediction of importation, as well as the ABM model which is used for identifying infections. Our results demonstrate that NeurABM identifies importation and nosocomial infection cases more accurately than existing methods.

8.1 Introduction

Healthcare-associated infections (HAIs), especially those caused by multi-drug resistant organisms (MDROs) pose a significant threat to patient safety and burden the healthcare system with increased costs due to longer hospital stays and more expensive therapies [36, 37, 38, 3, 39, 40]. Approximately 3% of hospitalized patients in the United States acquire an HAI during their stay, resulting in more than 35,000 deaths annually [252, 21, 22, 23]. Often, patients have already been infected but present no symptoms at admission (i.e., importation cases). For instance, the European Centre for Disease Prevention and Control (ECDC) estimates that importation cases contribute to 13% of HAI cases in Germany and 18.9% in Spain [41]. These importation cases can spread HAI-causing pathogens and lead to nosocomial infection cases, which can also be asymptomatic but further spread pathogens to additional healthy patients [42].

Despite the critical concerns associated with importation and nosocomial infection cases, identifying them rapidly and accurately still remains a challenging problem. Current methods to identify them include surveillance tests [43, 44], machine learning-based methods [85, 86], and transmission modeling-based methods [10, 68, 67, 121, 253, 254, 255]. However, they all suffer from unavoidable drawbacks. Surveillance tests such as culture or PCR tests are common methods in hospitals to identify importation and nosocomial infection cases. However, they are costly, require time to process, and are not 100% accurate [45]. Additionally, these are typically only able to be used for a subset of MDROs, and therefore not applicable for all kinds of HAIs. Machine learning and statistical techniques use patients' electronic health record (EHR) data to predict the probability of importation and nosocomial infection cases [85, 86]. However, the performance of machine learning methods has not proven to be high enough due to many reasons, including imbalance (since HAI cases form a very small fraction of the entire patient population), bias in the data (since testing is generally not done in a systematic manner), and the fact that machine learning methods do not incorporate epidemiological knowledge in their frameworks. Finally, modeling-based methods are based on detailed mechanistic models (e.g., compartmental mixing models [253, 254] and agent-based models (ABMs) [67, 10, 68]) to capture the transmission dynamics of HAIs within a healthcare facility. They are

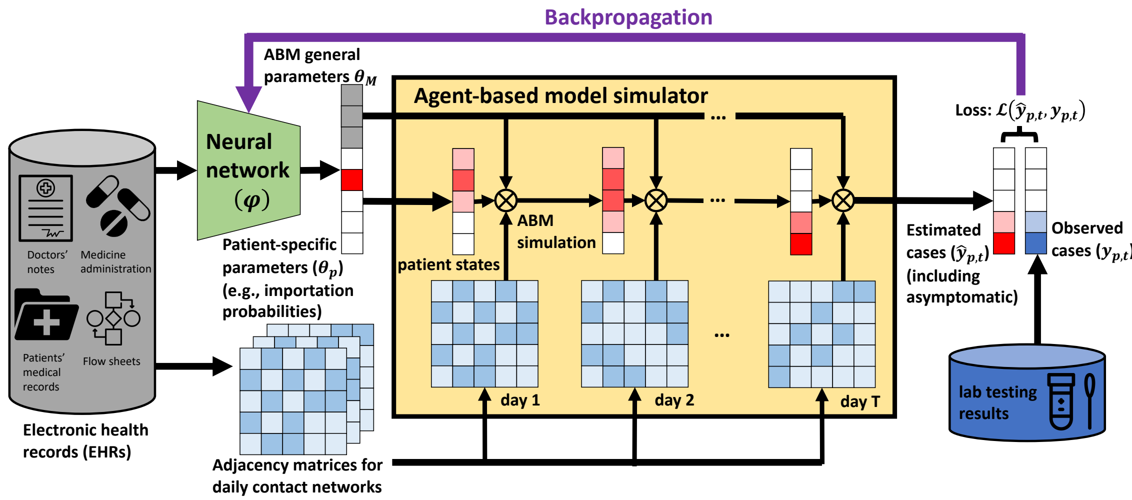


Figure 8.1: **NeurABM framework.** Our NeurABM framework involves 4 steps: (1) The neural network takes the patients' risk factor data collected from EHR as input, and outputs both the agent-based model (ABM) parameters, denoted by $\boldsymbol{\theta}_M$ (which are applicable to every patient), and patient-specific parameters, denoted by $\boldsymbol{\theta}_p$ (importation probabilities in this work, darker red means higher probability), for each patient p. (2) These parameters and the adjacency matrices for contact networks of each day collected from EHR are then fed into the ABM simulator for simulation for T days, and the output will be the probability that each patient p is in the *Carriage* state $\hat{y}_{p,t}$ for each day t (darker red means higher probability). (3) We then compare this $\hat{y}_{p,t}$ with the ground-truth observation of patients in *Carriage* state (via lab testing), denoted by $y_{p,t}$, and compute the loss $\mathcal{L}(\hat{y}_{p,t}, y_{p,t})$. (4) We backpropagate this loss to the neural network to tune the neural network parameters ϕ.

calibrated to infections in the hospital and use projections from such models for prediction. Although ABMs have used information about contact networks between patients and providers within healthcare facilities to model the infection status of an individual patient, they still cannot directly incorporate the risk factor of each patient from the EHR data, such as medications, lab results, vital signs, and device use history into modeling. As we will also show in the later results section, the inability of leveraging EHR data of patients leads to suboptimal performance.

In this work, we propose a new framework, NeurABM, to identify HAI importation and nosocomial infection cases by *coupling* a neural network and an ABM and training simultaneously. We use Methicillin-resistant *Staphylococcus aureus* (MRSA) as an example HAI in later sections. Figure 8.1 shows an overview of our framework. As shown in the figure, the neural network estimates the importation probability for each patient using EHR data, while the ABM incorporates MRSA dynamics and is used to estimate the MRSA infection probability. After training, NEURABM runs as a discrete time process; at each time step t, the following two steps are performed: (1) the neural network estimates the importation probability (i.e., identifies importation cases) for each new patient who enters the hospital, and (2) the ABM (which keeps track of the disease states of all patients in hospital till time $t-1$) runs the next step of the disease simulation to estimate all disease states (i.e., identify nosocomial infection cases, including those are asymptomatic) at time t. The parameters of NEURABM consist of two parts: those of the neural network and those of the ABM (e.g., general parameters such as transmission and recovery rates, and patient-specific parameters such as importation probabilities in this work); these parameters are learned by minimizing a loss function that considers the errors in the ABM projections and ground truth incidence data from EHR. Since the dynamics of MRSA transmission depend on the importation model (and conversely, through this kind of training process), this approach couples the neural network and ABM and is trained end-to-end, which mitigates the issues in using either of them individually. NEURABM significantly extends the work of [120], which was the first method to consider such a joint deep learning and ABM approach, by introducing approximation techniques to scale the disease model and incorporating rich patient-level EHR data.

We demonstrate the performance of NEURABM using EHR data for patients at the University of Virginia (UVA) hospital intensive care units (ICUs). Our results show that NEURABM identifies not only importation cases but also nosocomial infection cases better than other machine learning or modeling-based baselines. Note that our NEURABM is a general framework that integrates both neural networks and mechanistic models in an end-to-end way, one which can be easily extended to other ABMs or EHR data and study other clinical problems.

8.2 Methods

8.2.1 Transmission Model

In this work, we use the 2-MODE-SIS model to capture the MRSA spread dynamics in UVA ICUs [68]. 2-MODE-SIS is a pathogen load-based model that keeps track of pathogen load on all people and locations using a load vector $\boldsymbol{l}_t$. For each patient i, they can either be in the *Susceptible* (S) or *Carriage* (C) state. Specifically, the probability of transitioning from S to C is proportional to the amount of pathogen on this patient $\boldsymbol{l}_t(i)$, which can be formulated as a linear dose-response function $\beta\boldsymbol{l}_t(i)$ (β is the disease infectivity parameter). Once in the carriage state, the patient keeps shedding more pathogen loads at each step, which can later be transferred to its neighbors (including both people and locations). Such a shedding process continues until the patient recovers with a recovery probability δ.

For the pathogen load transfer, as described in the previous text, the 2-MODE-SIS model uses daily contact networks $\boldsymbol{A}_t$ to capture the exchange of pathogens among patients, HCWs, and locations. Specifically, we construct a transfer matrix $\boldsymbol{R}_t$ for each day t, where $\boldsymbol{R}_{ijt} = \tau_{ijt}\boldsymbol{A}_{ijt}$. Here, $\boldsymbol{A}_t$ is the adjacency matrix of contact networks on day t, and τ_{ijt} is the transfer ratio parameter (the ratio of pathogen being transferred (or remaining if $i = j$) from j to i on day t). Using this $\boldsymbol{R}_t$ and $\boldsymbol{l}_t(i)$, the 2-MODE-SIS model updates the pathogen loads every day as a linear operation. We also restrict the column-sums of $\boldsymbol{R}_t$ to be less than or equal to 1, which implies that the total amount of pathogen cannot increase after transfer (i.e., $|\boldsymbol{R}_t\boldsymbol{l}_t| \leq |\boldsymbol{l}_t|$). Note that susceptible patients may still carry a small amount of pathogen loads and spread them to others, and HCWs and locations are always in the *susceptible* state, which means that they can spread the MRSA pathogen loads but are non-infectable.

8.2.2 NeurABM Framework

As shown in Figure 8.1, our NeurABM is composed of two components: a neural network and an agent-based model (ABM) simulator. The neural network is used to estimate both patient-specific parameters and ABM parameters.

For the neural network component, we take the risk factors $\boldsymbol{f}$ (where f_p is for patient p) as input and then estimate both the patient-specific parameters $\boldsymbol{\theta}_p$ (which is a vector and each element θ_p is for patient p and only influenced by the patient itself's risk factors f_p, in this work it is importation probability for each patient) and ABM parameters $\boldsymbol{\theta}_M$ (i.e., the general parameters that apply to every patient in the 2-Mode-SIS model, like disease infectivity parameter β or recovery probability δ) together. The neural network is parameterized by ϕ can be represented by

$$\boldsymbol{\theta}_p, \boldsymbol{\theta}_M = NN(\boldsymbol{f}; \phi) \tag{8.1}$$

For the ABM simulator, we implement the simulation process of the 2-Mode-SIS model using matrix operations in a differentiable way. ABM simulator takes the adjacency matrices of contact networks $\boldsymbol{A}_t$ and the parameters learned by neural networks $(\boldsymbol{\theta}_p, \boldsymbol{\theta}_M)$ as input, and simulates MRSA spread in T days to estimate patient states on each day $\hat{\boldsymbol{y}}$ (where $\hat{y}_{p,t}$ is for patient p on day t). Specifically, the simulation process of the 2-Mode-SIS model can be decomposed into three substeps: (1) pathogen load transmission where the load transfers via contact edges, (2) updating the states for each patient based on their pathogen loads and recovery probability, and (3) updating the timestep from day t to day $t+1$. This process can be repeated for arbitrary steps to simulate the MRSA spread over T days. The ABM part can be represented by

$$\hat{\boldsymbol{y}} = ABM(\boldsymbol{A}; \boldsymbol{\theta}_p, \boldsymbol{\theta}_M) \tag{8.2}$$

With both the neural network and the ABM simulator, we can then integrate both components together to train simultaneously. Specifically, one training epoch comprises the following four steps.

Step 1

We feed the risk factor data $\boldsymbol{f}$ into the neural network as the input to estimate the patient-specific parameters $\boldsymbol{\theta}_p$. In this work, $\boldsymbol{\theta}_p$ is the probability of being importation cases for each patient p. It is a vector of size N, where N is the number of patients in the contact network. Meanwhile, the neural network will also give the general ABM parameters $\boldsymbol{\theta}_M$ that are applied to all patients (e.g., β, α, $\cdots$).

Step 2

We then feed $\boldsymbol{\theta}_p$, $\boldsymbol{\theta}_M$, and the contact networks $\boldsymbol{A}_t$ into the ABM simulator and simulate for T steps. The output will be the vector $\hat{\boldsymbol{y}}$ of size $N \times T$, in which $\hat{y}_{p,t}$ represents the probability of being in the state *carriage* for patient p on day t.

Step 3

We compare the estimated carriage probability $\hat{\boldsymbol{y}}$ with the corresponding ground-truth observations (i.e., known carriage patients based on lab testing) $\boldsymbol{y}$. We use the weighted binary cross entropy loss (BCE loss) $\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \sum_p \sum_t w_{pos} y_{p,t} \log(\hat{y}_{p,t}) + w_{neg}(1 - y_{p,t}) \log(1 - \hat{y}_{p,t})$ as the loss function. Here w_{pos} and w_{neg} are the weights for positive and negative observations. We set $w_{pos} : w_{neg} \propto \sum_p \sum_t \mathbb{1}[y_{p,t} = 0] : \sum_p \sum_t \mathbb{1}[y_{p,t} = 1]$, where $\mathbb{1}[\cdot]$ is the indicator function, which is 1 if the condition is true, and 0 otherwise.

Step 4

With the BCE loss $\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y})$, and the differentiable ABM simulator, we can calculate the gradient of the loss with respect to the neural network parameters ϕ via backpropagation.

This allows us to better tune the neural network and learn more reasonable parameters as the input for the ABM simulator. The pseudo code is given in Algorithm 8.

Algorithm 8 NEURABM

Inputs: Risk factors of patients $\boldsymbol{f}$, adjacency matrices of contact networks $\boldsymbol{A}$, lab testing results $\boldsymbol{y}$
for each epoch **do**
    Estimate the patient-specific parameters $\boldsymbol{\theta}_p$ and ABM parameters $\boldsymbol{\theta}_M$
    for each timestep t **do**
        Estimate the patient states on day t: $\hat{\boldsymbol{y}}_t$
    end for
    Compute the loss $\mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y})$
    Update the neural network parameters ϕ
end for
Outputs: Neural network parameters ϕ

The formal problem formulation is given as follows:

Problem Formulation

Given the risk factors of patients $\boldsymbol{f}$, adjacency matrices of contact networks $\boldsymbol{A}$, lab testing results $\boldsymbol{y}$, the neural network part as in Equation 8.1, and the ABM part as in Equation 8.2, find the best neural network parameter ϕ such that

$$\arg\min_{\phi} \mathcal{L}(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \mathcal{L}(ABM(\boldsymbol{A}; \boldsymbol{\theta}_p, \boldsymbol{\theta}_M), \boldsymbol{y}) = \mathcal{L}(ABM(\boldsymbol{A}; NN(\boldsymbol{f}; \phi)), \boldsymbol{y}) \tag{8.3}$$

With the trained NEURABM model, we can identify importation and nosocomial infection cases. For importation cases, note that the neural network part can estimate both patient-specific parameters $\boldsymbol{\theta}_p$ and ABM parameters $\boldsymbol{\theta}_M$. We can just use this $\boldsymbol{\theta}_p$ as the estimated probability of being an importation case for each patient. For nosocomial infection cases, we will use these ABM parameters $\boldsymbol{\theta}_M$ as well as the patient state vector on the last day of week $k-1$ to run simulations and estimate the patient states in week k.

8.3 Results

Here, we show the performance of our NEURABM in identifying MRSA importation and nosocomial infection cases in the ICUs of the UVA hospital in 2019. We used EHR data from the UVA hospital to construct patient contact networks (used by the ABM) and collect patient risk factors (used by the neural network). We use the 2-MODE-SIS model [68, 30] as the ABM for disease transmission in NEURABM. Ground-truth MRSA infections are identified from lab test results for each patient in the EHR. For each week k, we used the contact networks, patient risk factors, and lab test results until week $k-1$ to train the NEURABM and identify importation cases before week $k-1$. We then ran the 2-MODE-SIS model for 7 more days to infer the infection states of patients for week k, which correspond to nosocomial infections (see section 8.2). Note that only data prior to week k are used in this process—this is same as the setting considered in [10] for detection of asymptotic cases (though their ABM is different); further, they do not consider the importation problem. We also compared NEURABM with other baselines in machine learning categories (feedforward neural network [91], decision tree [256], Naive Bayes [257]), modeling categories (the 2-MODE-SIS model [68, 30], SILI-ABM model [10]), and clinical heuristic categories (length of stay [10]). Note that SILI-ABM model [10] is not designed to identify importation cases, we only compare with them for identifying nosocomial infection cases.

- 2-MODE-SIS: As an ablation study, we use only the 2-MODE-SIS model to identify importation and nosocomial infection cases to see how it performs.
- SILI-ABM: Sequential, individual-level inference (SILI) model [10] is an agent-based model that uses Bayesian approaches to infer the colonization probability of confirmed carriers for each patient.
- Feedforward neural network: Feedforward neural networks have shown good performance in different clinical estimation and prediction tasks in recent years. In this

work, we also implemented a multi-layer perceptron (MLP) feedforward neural network using Scikit-learn [91] in Python.

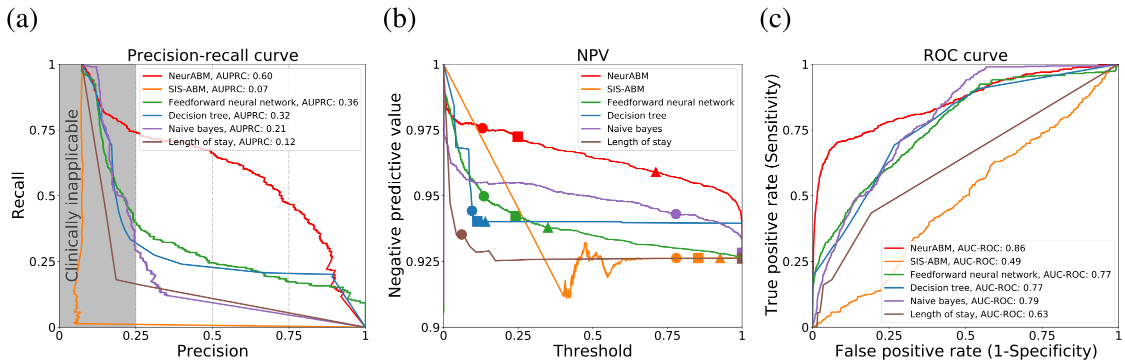
- Decision tree: Decision tree is also a widely-used machine learning algorithm in classification and prediction [257]. It represents choices in a tree-like graph, illustrating different outcomes as branches. In this work, we also use a decision tree as a baseline using Scikit-learn [91] in Python.

- Naive Bayes: Naive Bayes is also a powerful and efficient algorithm in machine learning [256]. It operates on the principle of Bayes theorem, where "naive" assumes that features are independent. Similarly, we use Scikit-learn [91] and implemented a Naive Bayes classifier.

- Length of stay: As suggested by a previous study [10], patients with longer hospital stay days are suspected of having a higher risk of infection. Therefore, we use the EHR to calculate the days a patient stays in the hospital, and use it as the clinical heuristic baseline for comparison.

8.3.1 Identifying Importation Cases

In Figure 8.2a, we show the precision-recall curves for NEURABM and other methods. Note that in clinical practice, very low precision is not very useful, since this means too many tests and treatments do not help to identify and treat MRSA cases. Therefore, we always expect high recall with not-too-low precision. Following previous work [258], we consider precision smaller than 0.25 as clinically inapplicable and focus on three important precision levels: 0.25, 0.5, and 0.75 (dashed grey lines). NEURABM always achieves the highest recall when precision is 0.25, 0.5, and 0.75, indicating that our framework is effective. Besides, the area under the precision-recall curve (AUPRC) for NEURABM is the largest (0.60) among all methods. In Figure 8.2b, we show how the negative predictive value (NPV) changes with threshold changes (when the estimated probability is higher than



(d) Identifying MRSA importation cases

Method	Precision	Recall (Sensitivity, True positive rate)	F1 score	AUPRC	False positive rate (1-Specificity)	Negative predictive value (NPV)	AUC-ROC
	0.25	0.74	0.37		0.18	0.98	
NEURABM	0.50	0.66	0.57	**0.60**	0.05	0.97	**0.86**
	0.75	0.47	0.58		0.01	0.96	
	0.25	0.01	0.02		0.01	0.93	
2-MODE-SIS	0.25	0.01	0.01	0.07	0.01	0.93	0.49
	0.75	0.01	0.01		0.01	0.93	
	0.25	0.40	0.31		0.10	0.95	
Feedforward neural network	0.50	0.24	0.33	0.36	0.02	0.94	0.77
	0.75	0.17	0.28		0.01	0.94	
	0.25	0.32	0.28		0.08	0.94	
Decision tree	0.50	0.23	0.31	0.32	0.02	0.94	0.77
	0.75	0.20	0.32		0.01	0.94	
	0.25	0.29	0.27		0.07	0.94	
Naive Bayes	0.50	0.09	0.16	0.21	0.02	0.93	0.79
	0.75	0.05	0.09		0.01	0.93	
	0.25	0.16	0.20		0.04	0.94	
Length of stay	0.50	0.11	0.18	0.12	0.03	0.93	0.63
	0.75	0.05	0.10		0.02	0.93	

Figure 8.2: **Performance in identifying importation cases.** The performance in identifying importation cases includes: (a) The precision-recall curves (PRC). The x-axis represents precision, and the y-axis represents recall. The red and other color curves represent NEURABM and other baselines. (b) The negative predictive value (NPV) with different thresholds. The x-axis is the threshold for classification, and the y-axis is the NPV value. Circles, squares, and triangles correspond to the thresholds and NPV values where precision is 0.25, 0.5, and 0.75, respectively. (c) The receiver operating characteristic (ROC) curves in identifying MRSA importation cases. The x-axis is the false positive rate, and the y-axis is the true positive rate. (d) The recall, F1 score, AUPRC, false positive rate, NPV, and AUC-ROC under different precisions (0.25, 0.5, 0.75). The best AUPRC and AUC-ROC are in bold.

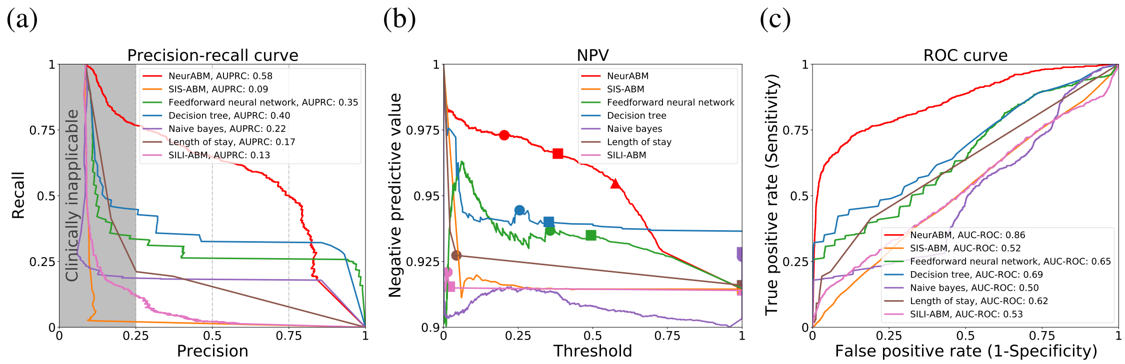
the threshold, we classify this patient as a MRSA importation case, and vice versa). The negative predictive value is the fraction of the number of true negative cases over predicted negative cases. Intuitively, a higher NPV means that when we predict a patient as negative, he or she is less likely to be a false negative patient that we fail to identify. We can see that NEURABM's NPV rate is always higher than 0.95 and other baselines, indicating that our NeurABM can identify the importation cases well with fewer missing/undetected patients. We also show the receiver operating characteristic (ROC) curve in Figure 8.2c. Here, the area under the curve (AUC-ROC) for our framework is still the largest (0.86) compared to other baselines. In the table in Figure 8.2d, we also list the recall, F1 score, and false positive rate corresponding to different precision values. We can see that NeurABM always achieves the highest recall and F1 score with a given precision, indicating the effectiveness of our framework in identifying MRSA importation cases.

8.3.2 Identifying Nosocomial Infection Cases

As shown in Figure 8.3a, the x-axis and y-axis represent precision and recall, respectively. The red curve represents the results for NeurABM. As shown in the figure, the area under the precision-recall curve for our framework is the largest (0.58) compared to other baselines. The dashed grey lines correspond to the precision of 0.25, 0.5, and 0.75. Again, NeurABM always achieves the highest recall with precision equal to 0.25, 0.5, and 0.75, indicating that our framework is effective. In Figure 8.3b, we show how the negative predictive value (NPV) changes with the threshold for classification. We can see that the NPV rate is always higher than 0.9 and other baselines, indicating that NeurABM can identify nosocomial infection cases well with fewer missing/undetected patients. We also show the ROC curve in Figure 8.3c. Here, the area under the ROC curve for our framework is the largest (0.86) compared to other baselines. In the table in Figure 8.3d, NeurABM always achieves the highest recall and F1 score with a given precision, demonstrating the effectiveness of our framework in identifying nosocomial MRSA infection cases.

8.3.3 Identification of High-risk MRSA Cases

This can help in implementing better infection control methods within the hospital. We consider the strategy of testing patients based on the ranked infected probability estimated by NeurABM and other baselines, and determine what percentage of MRSA cases can



(d) Identifying nosocomial infection cases

Method	Precision	Recall (Sensitivity, True positive rate)	F1 score	AUPRC	False positive rate (1-Specificity)	Negative predictive value (NPV)	AUC-ROC
NEURABM	0.25	0.77	0.38	**0.58**	0.22	0.97	**0.86**
	0.50	0.65	0.56		0.06	0.97	
	0.75	0.50	0.60		0.02	0.95	
2-MODE-SIS	0.25	0.02	0.04	0.09	0.02	0.91	0.52
	0.50	0.01	0.03		0.01	0.91	
	0.75	0.01	0.01		0.01	0.91	
Feedforward neural network	0.25	0.34	0.29	0.35	0.10	0.94	0.65
	0.50	0.26	0.34		0.03	0.93	
	0.75	0.26	0.39		0.01	0.93	
Decision tree	0.25	0.45	0.32	0.40	0.13	0.94	0.69
	0.50	0.33	0.39		0.03	0.94	
	0.75	0.32	0.45		0.01	0.94	
Naive Bayes	0.25	0.19	0.21	0.22	0.06	0.93	0.50
	0.50	0.18	0.27		0.02	0.93	
	0.75	0.18	0.29		0.01	0.93	
Length of stay	0.25	0.21	0.23	0.17	0.06	0.93	0.62
	0.50	0.15	0.24		0.03	0.92	
	0.75	0.08	0.14		0.01	0.92	
SILI-ABM	0.25	0.12	0.16	0.13	0.03	0.92	0.53
	0.50	0.02	0.04		0.01	0.92	
	0.75	0.01	0.02		0.01	0.91	

Figure 8.3: **Performance in identifying nosocomial infection cases.** The performance in identifying nosocomial infection cases includes: (a) The precision-recall curves. The red and other color curves represent NEURABM and other baselines. (b) The negative predictive value with different thresholds. Circles, squares, and triangles correspond to the thresholds and NPV values where precision is 0.25, 0.5, and 0.75, respectively. (c) The receiver operating characteristic curves in identifying MRSA nosocomial infection cases. (d) The recall, F1 score, AUPRC, false positive rate, NPV, and AUC-ROC under different precisions. The best AUPRC and AUC-ROC are in bold.

be identified. This is shown in Figure 8.4, where we rank all patients according to the estimated infected probability of each method from high to low, and test patients in this order. Here, we observe that NEURABM can always identify more nosocomial MRSA infection cases (y-axis) given the same test budget (x-axis), which suggests that our framework is effective and practical in identifying MRSA cases in clinical settings.

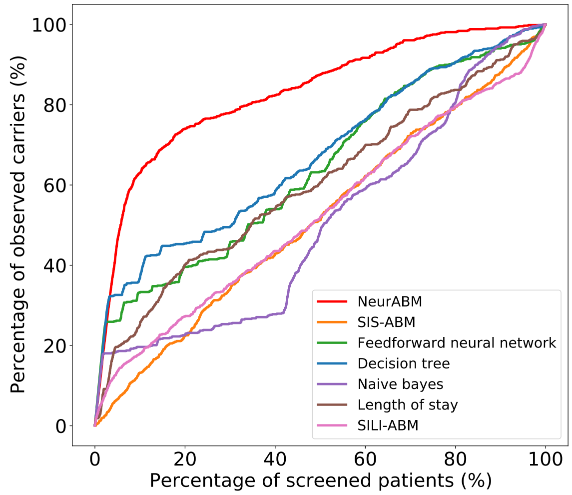


Figure 8.4: **Percentage of identified MRSA cases by screening "high-risk" patients.** For each patient in the UVA ICUs, we use each method to estimate their MRSA infection probability and rank them according to this probability from high to low. We then screen different percentages of patients (x-axis) and see how many actual MRSA cases can be identified (y-axis). As seen in the figure, NEURABM can always identify more MRSA cases than other baselines.

8.3.4 Case Study: Explanation of Neural Network

We further investigate which patient risk factors are considered as having a high risk of being importation cases by NEURABM. As shown in Figure 8.5, each dot represents a patient, and the color of the dot represents the risk factor value (red means higher, and blue means lower). Dots with a higher impact value (to the right of the control line) mean that NEURABM tends to classify this patient as having a higher probability of being an importation case (and vice versa). Here, we can see that patients who have had contact with more MRSA patients in the past 7/14 days prior to ICU admission are considered the most dangerous. Besides, we notice that patients with a device usage history, who come from or were discharged to other healthcare facilities last time, and emergency patients are more likely to be considered as importation cases.

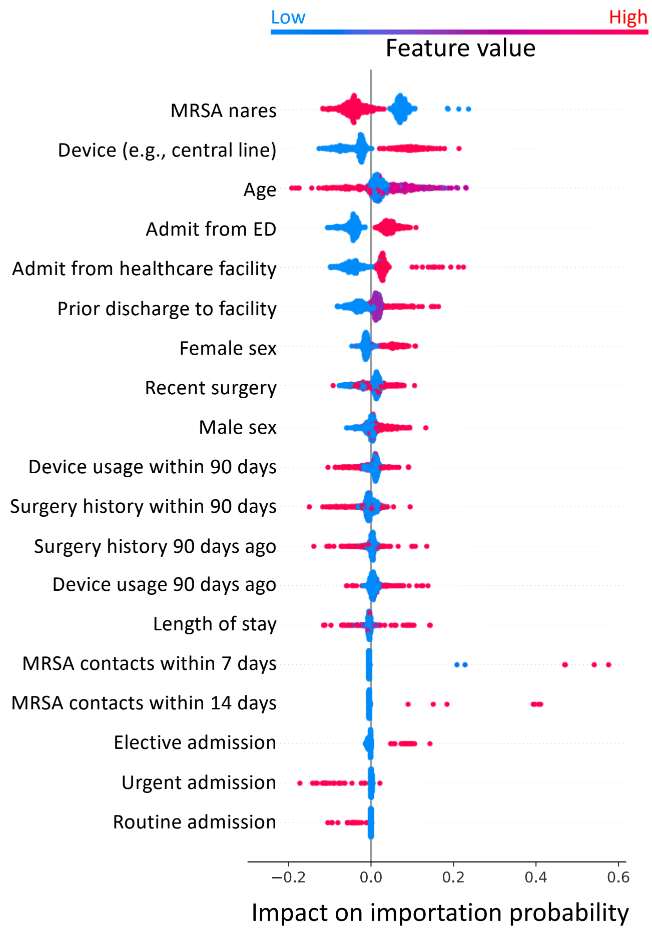


Figure 8.5: **The patient risk factors that are considered as having a high risk of being importation cases by the trained neural network.** Color of the dots represent risk factor values, with red indicating higher values. A higher impact means that NEURABM is more likely to consider the patient as an importation case.

8.4 Discussion

NEURABM is a novel framework for supporting a diverse class of HAI surveillance and control questions. This framework allows the use of diverse data sources from EHRs by deep learning methods and ABMs. While joint deep learning and ABMs have been considered before, as in [120] (which our work builds on), our approach is distinct in that the deep learning method for prediction of importations and the ABM run in lock-step, allowing the ABM to directly incorporate the predictions of importation. Additionally, the detailed contact network used in the 2-MODE-SIS model in NEURABM also plays an important role, and justifies the utility of detailed contact representations for surveillance and control of HAIs.

Our experiments show that NEURABM identifies not only MRSA importation cases but also nosocomial infection cases in UVA ICUs, with performance that can be considered clinically useful; in contrast, prior methods using EHR data haven't achieved this level of performance. Moreover, the negative predictive value (NPV) for our method is always higher than 0.95, indicating that NEURABM can identify these cases well with fewer missing/undetected patients. Our case study also reveals the risk factors that are highly related to MRSA importations, which allows clinicians to better respond to potential MRSA importation cases in their hospital.

However, our framework is not without limitations. One limitation is that the 2-MODE-SIS model may oversimplify MRSA carriage by defining just two states: *susceptible* and *carriage*. There are different forms of MRSA carriage, including various clinical infection types such as skin abscess or bloodstream infection, which may confer different levels of risk for transmission. Similarly, the 2-MODE-SIS model does not have an explicit "colonization" state. However, our framework is quite general and can be extended to use more complex ABMs that incorporate these states separately. People can also include more types of datasets (e.g., hospital layouts) as input to better learn the spatial and temporal patterns in HAI spread. We also did not take false negative or false positive MRSA tests into consideration, although the sensitivity and specificity of the nares MRSA tests are considered quite good. Another potential limitation is that MRSA surveillance has several biases, and NEURABM is not trained to specifically mitigate these biases.

Nevertheless, NEURABM opens up a new direction of research for using both rich patient risk factors from EHR data as well as agent-based models designed with epidemiological knowledge and can be used for many other questions about HAI spread in hospitals. The specific architecture of our method, which runs the deep learning method and ABM step at each time allows us to add predictors at different stages within this framework to address other kinds of questions. For instance, the framework can be adapted to forecast nosocomial infections and severe outcomes by adding a predictor for these outcomes after the ABM. The ABM can also be enriched with representations of interventions implemented in the hospital. For patient-specific parameters, we focus on importation probabilities in this work. However, many other parameters such as recovery rate can also be patient-specific and our NEURABM framework can be easily extended to it. Besides, our framework could also be extended from learning a point estimate for unknown parameters to learning the distribution of them. The performance of these variations depends on data availability, and is a promising topic for future research. It can also be adapted to other healthcare-associated infections, such as *C.diff.* The fact that NEURABM is amenable to such adaptations quite easily is an indication of its generality.

CHAPTER 9

COMBINING AGENT-BASED MODELS AND NEURAL NETWORKS TO FORECAST FUTURE CASES OF HEALTHCARE ASSOCIATED INFECTIONS

Machine learning (ML) methods are commonly used in infectious disease forecasting due to their flexibility in processing different kinds of input datasets. However, their performance is frequently suboptimal because of lacking the support of epidemiological knowledge. Therefore, researchers turn to agent-based models (ABMs), which can capture detailed agent actions and interactions within contact networks. However, ABMs rely on the assumption that the contact networks are static, limiting their accuracy for dynamic environments like hospitals. Therefore, we propose FORECASTABM, which integrates ABMs and neural networks to overcome the limitations of both of them. By using an adapter neural network, it forecasts future healthcare-associated infection cases based on current patient states without needing future contact networks, ensuring higher accuracy and practical applicability in real-world scenarios.

9.1 Introduction

Healthcare-associated infections (HAIs), such as Methicillin-resistant *Staphylococcus aureus* (MRSA), Carbapenem-Resistant Enterobacterales (CRE), and *Clostridioides difficile* (*C. diff*), are infections acquired by patients during treatment at healthcare facilities such as hospitals and long-term nursing homes [21, 22, 23]. They impose a tremendous burden on hospitals since they may lead to severe outcomes, longer hospital stays for patients, and increase the risk of mortality [36, 37, 38, 3, 39, 40]. Since many patients can be infected with HAIs without presenting any symptoms (i.e., asymptomatic) but are still able to infect other healthy patients, it is essential to accurately forecast future HAI patients to prevent the HAI spread in healthcare facilities. Specifically, motivated by the above consideration,

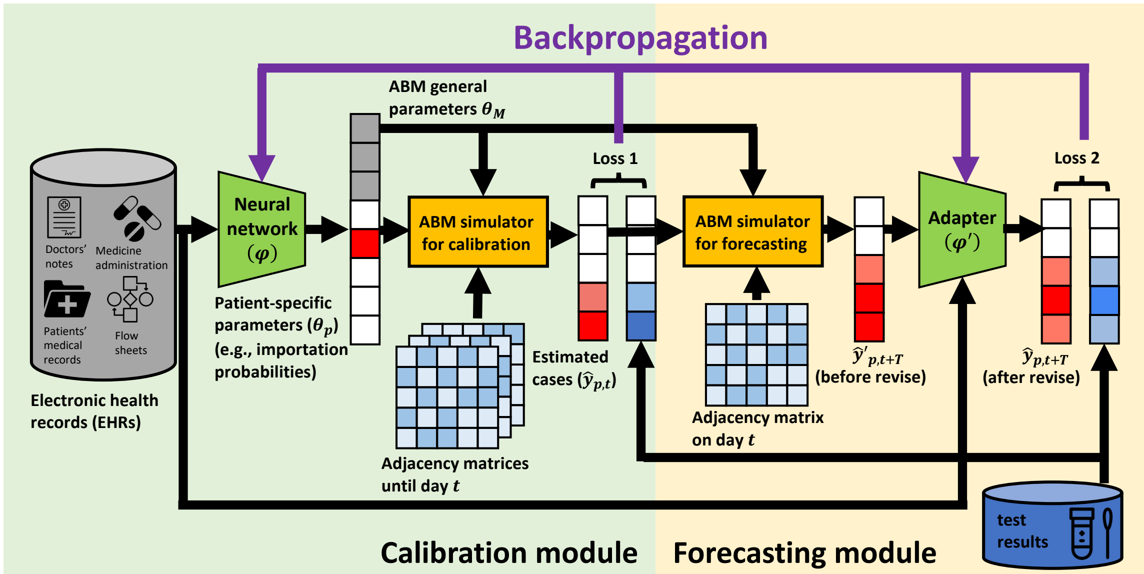


Figure 9.1: **Our FORECASTABM framework is composed by two modules:** (1) Calibration module: The neural network ϕ and the ABM simulator for calibration take the patients' risk factor data and adjacency matrices until day t as input, and outputs agent-based model (ABM) parameters $\boldsymbol{\theta}_M$ (which are applicable to every patient), patient-specific parameters $\boldsymbol{\theta}_p$ (importation probabilities in this work, darker red means higher probability) for each patient p, and the estimated patient states $\hat{y}_{p,t}$ for day t. The loss for thismodules is $\mathcal{L}(\hat{y}_{p,t}, y_{p,t})$. (2) Forecasting module: The $\hat{y}_{p,t}$ was further feed into ABM simulator for forecasting, which naively assumes that the future contact networks will repeat the day t, to get an rough forecast $\hat{y}'_{p,t+T}$. Since this $\hat{y}'_{p,t+T}$ is inaccurate, we then feed this $\hat{y}'_{p,t+T}$ into an adapter neural network parameterized by ϕ', which gives the final revised forecast $\hat{y}_{p,t+T}$ for T days ahead. The loss for this part is $\mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T})$.

we focus on the following problem in a healthcare facility setting:

Clinical Problem 3 (Forecasting Problem, FP)**.** *Given the infection states of patients today, how can we know who will be infected and who will not be infected in 7/14 days ahead?*

One natural idea to solve the forecasting problem is to use machine learning-based methods such as feedforward neural networks [91], decision trees [256], and Naive Bayes [257]. Usually, people feed the patients' risk factors (e.g., age, occupation, surgery history) and infection state for today into the model as the input and output the infection states for 7 days ahead. They then compare these estimated patient states with the ground-truth real-world observations and train the machine learning models. Although these machine learning methods have shown some potential in forecasting [259], they are less supported by epidemiological knowledge and suffer from the inability to capture the transmission events that lead to future infections [260].

To tackle this, people turn to agent-based models (ABMs), which also take the contact networks in healthcare facilities as input to capture the actions and interactions between agents and give individual-level forecasts [30]. For example, ABMs have been used to forecast COVID-19 spread at county, state, national, and international levels [261]. These ABMs often assume that contact networks are relatively constant, which allows them to make the assumption that future networks will be similar to previous ones. However, the contact networks in healthcare facilities change dynamically every day [260]. For example, patients are admitted to or discharged from hospitals every day, and they may also be transferred between different units and wards. This prevents people from naively using previous networks for the future and running ABMs for forecasts. In fact, most existing works using ABMs for identifying HAI cases are mostly retrospective [165, 10], which actually cannot solve the aforementioned forecasting problem.

Recently, a new framework integrating ABMs and neural networks, NEURABM, was proposed to leverage the advantages of mechanistic modeling and machine learning methods to provide accurate, patient-specific estimations. In its retrospective study to identify infection cases, NEURABM has shown much better performance than solely modeling or ML-based methods. However, NEURABM still relies on ABMs and contact networks to forecast future infections, making them impractical in real-world applications.

To address this, we propose FORECASTABM, which significantly extends the NEURABM framework from only retrospectively identifying cases to forecasting future cases without contact networks. As shown in Figure 9.1, we use another carefully tuned adapter network ϕ' to forecast future cases based on the NEURABM estimated patient states for today. To ensure the adapter benefits from the epidemiological knowledge encoded in NEURABM, we train both NEURABM and the adapter in an end-to-end manner to minimize training bias. Compared to traditional two-step processes or naive machine learning methods, we demonstrate that FORECASTABM achieves higher accuracy in forecasting future cases.

9.2 Methods

9.2.1 Transmission Model

Following the NEURABM work, we use the 2-MODE-SIS model to describe the MRSA spread [68]. As a pathogen load-based model, 2-MODE-SIS uses a load vector $\boldsymbol{l}_t$ to keep track of pathogen load on all people and locations. Patients can be either in the *Susceptible* (S) or *Carriage* (C) state, and the probability of transitioning from *Susceptible* to *Carriage* for a patient i is proportional to the amount of pathogen on them ($\boldsymbol{l}_t(i)$), and we use a linear dose-response function $\beta\boldsymbol{l}_t(i)$ (β is the disease infectivity parameter) to formulate the probability. Patients in the *Carriage* state will keep shedding more pathogen loads until recovery with a recovery probability δ. Specifically, in the 2-MODE-SIS model, even patients in the S state could still carry a small amount of pathogen loads and spread them to others. Similarly, HCWs and locations can also spread the pathogen loads. However, we assume they are non-infectable.

To capture how the pathogen loads transfer among entities, the 2-MODE-SIS model takes the adjacency matrix of contact networks for each day t, or $\boldsymbol{A}_t$ as input, and constructs a transfer matrix $\boldsymbol{R}_t$ for each day t, where $\boldsymbol{R}_{ijt} = \tau_{ijt}\boldsymbol{A}_{ijt}$, and τ_{ijt} is the transfer ratio parameter to capture the ratio of pathogen being transferred (or remaining if $i = j$) from j to i on day t. With the $\boldsymbol{R}_t$ and aforementioned $\boldsymbol{l}_t(i)$, the pathogen loads update from t to $t + 1$ can be described as a linear operation $\boldsymbol{l}_{t+1} = \boldsymbol{R}_t\boldsymbol{l}_t$. Specifically, the column-sums of $\boldsymbol{R}_t$ should be restricted to be less than or equal to 1 since the total amount of pathogen cannot increase after transfer (i.e., $|\boldsymbol{R}_t\boldsymbol{l}_t| \leq |\boldsymbol{l}_t|$).

9.2.2 FORECASTABM Framework

As shown in Figure 9.1, our NEURABM is composed of two modules: calibration module and forecasting module.

The calibration module takes the patients' risk factor $\boldsymbol{f}$ (where f_p is for patient p) and adjacency matrices of contact networks $\boldsymbol{A}_t$ as input, and outputs agent-based model (ABM) parameters $\boldsymbol{\theta}_M$ (which are applicable to every patient, such as disease infectivity parameter β or recovery probability δ), patient-specific parameters $\boldsymbol{\theta}_p$ (importation probabilities in this work, each element θ_p in $\boldsymbol{\theta}_p$ is only influenced by the patient p's own risk factors f_p) for each patient p, and the estimated patient states $\hat{y}_{p,t}$ for day t. It can be further decomposed into the neural network part and the ABM simulator for calibration part. As in Equation 9.1, the neural network is parameterized by ϕ and takes $\boldsymbol{f}$ as input to estimate the ABM parameters $\boldsymbol{\theta}_M$ and patient-specific parameters $\boldsymbol{\theta}_p$.

$$\boldsymbol{\theta}_p, \boldsymbol{\theta}_M = \mathrm{NN}(\boldsymbol{f}; \phi) \tag{9.1}$$

Next, the ABM simulator for calibration takes $\boldsymbol{A}_t$, $\boldsymbol{\theta}_p$, and $\boldsymbol{\theta}_M$ as input, runs simulations to estimate the patient states $\hat{y}_{p,t}$ for each patient p on day t as in Equation 9.2. Following the NEURABM framework, the simulation process involves three substeps: (1) pathogen load transmission where the load transfers via contact edges, (2) updating the states for each patient based on their pathogen loads and recovery probability, and (3) updating the timesteps.

$$\hat{y}_{p,t} = \mathrm{ABM4Calibration}(\boldsymbol{A}; \boldsymbol{\theta}_p, \boldsymbol{\theta}_M) \tag{9.2}$$

With the estimated $\hat{y}_{p,t}$, we compare it with the ground-truth observations $y_{p,t}$ collected from lab test results. Specifically, we use the weighted binary cross-entropy loss (BCE loss) in Equation 9.3 as the loss function.

$$\mathcal{L}(\hat{y}_{p,t}, y_{p,t}) = \sum_p \sum_t w_{pos} y_{p,t} \log(\hat{y}_{p,t}) + w_{neg}(1 - y_{p,t}) \log(1 - \hat{y}_{p,t}) \tag{9.3}$$

Here w_{pos} and w_{neg} are the weights for positive and negative observations. We set $w_{pos} : w_{neg} \propto \sum_p \sum_t \mathbb{1}[y_{p,t} = 0] : \sum_p \sum_t \mathbb{1}[y_{p,t} = 1]$, where $\mathbb{1}[\cdot]$ is the indicator function, which is 1 if the condition is true, and 0 otherwise.

After the calibration module, the forecasting module takes the estimated patient states $\hat{y}_{p,t}$ as the patients' risk factor $\boldsymbol{f}$ as input, and outputs the final forecast for T days ahead, or $\hat{y}_{p,t+T}$. Specifically, it can also be further decomposed into the ABM simulator 2 part and the adapter part. For the ABM simulator for forecasting part, as shown in Equation 9.4, which naively assumes that the future contact networks between day t and day $t+T$ will repeat the contact networks for day t, and runs ABM simulations to get a rough forecast $\hat{y}'_{p,t+T}$.

$$\hat{y}'_{p,t+T} = \text{ABM4Forecasting}(\hat{y}_{p,t}; \boldsymbol{A}_t, \boldsymbol{\theta}_p, \boldsymbol{\theta}_M) \tag{9.4}$$

However, as described in section 9.1, the contact networks in hospital settings are changing dynamically every day, making this $\hat{y}'_{p,t+T}$ inaccurate. Therefore, to get a better forecast, we further feed this $\hat{y}'_{p,t+T}$ into another adapter neural network parameterized by ϕ', which also takes each patient's risk factor $\boldsymbol{f}$ as input to give a revised forecast $\hat{y}_{p,t+T}$ as shown in Equation 9.5.

$$\hat{y}_{p,t+T} = \text{Adapter}(\hat{y}'_{p,t+T}, \boldsymbol{f}; \phi') \tag{9.5}$$

To train this adapter, we also added another BCE loss in Equation 9.6 and trained the whole framework with both losses in an end-to-end way. We also show the pseudocode in Algorithm 9.

$$\mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T}) = \sum_p \sum_t w_{pos} y_{p,t+T} \log(\hat{y}_{p,t+T}) + w_{neg}(1-y_{p,t+T}) \log(1-\hat{y}_{p,t+T}) \tag{9.6}$$

The optimization object is given as follows:

Algorithm 9 FORECASTABM

Inputs: Risk factors of patients $\boldsymbol{f}$, adjacency matrices of contact networks $\boldsymbol{A}$, lab testing results y
for each timestep t **do**
  for each epoch **do**
    Estimate the patient-specific parameters and ABM parameters $\boldsymbol{\theta}_p, \boldsymbol{\theta}_M = \mathrm{NN}(\boldsymbol{f}; \phi)$
    Estimate the patient states on day t: $\hat{y}_{p,t} = \mathrm{ABM4Calibration}(\boldsymbol{A}; \boldsymbol{\theta}_p, \boldsymbol{\theta}_M)$
    Roughly forecast the patient states on day $t+T$:
    $\hat{y}'_{p,t+T} = \mathrm{ABM4Forecasting}(\hat{y}_{p,t}; \boldsymbol{A}_t, \boldsymbol{\theta}_p, \boldsymbol{\theta}_M)$
    Use adapter to revise the $\hat{y}'_{p,t+T}$ to get $\hat{y}_{p,t+T} = \mathrm{Adapter}(\hat{y}'_{p,t+T}, \boldsymbol{f}; \phi')$
    Compute the loss $\mathcal{L}(\hat{y}_{p,t}, y_{p,t}) + \mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T})$
    Update the neural network parameters ϕ and adapter parameters ϕ'
  end for
end for
Outputs: Neural network parameters ϕ, and adapter parameters ϕ'

9.2.3 Optimization Object

Given the risk factors of patients $\boldsymbol{f}$, adjacency matrices of contact networks $\boldsymbol{A}$, lab testing results $\boldsymbol{y}$, find the best neural network parameter ϕ and adapter parameter ϕ' such that

$$\begin{aligned}
&\arg\min_{\phi,\phi'} \ \text{Calibration module loss} + \text{Forecsting module loss} \\
&\quad = \mathcal{L}(\hat{y}_{p,t}, y_{p,t}) + \mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T}) \\
&\quad = \mathcal{L}(\mathrm{ABM4Calibration}(\boldsymbol{A}; \mathrm{NN}(\boldsymbol{f}; \phi)), y_{p,t}) + \mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T}) \\
&\quad = \mathcal{L}(\mathrm{ABM4Calibration}(\boldsymbol{A}; \mathrm{NN}(\boldsymbol{f}; \phi)), y_{p,t}) \\
&\quad + \mathcal{L}(\mathrm{Adapter}(\mathrm{ABM4Forecasting}(\mathrm{ABM4Calibration}(\boldsymbol{A}; \mathrm{NN}(\boldsymbol{f}; \phi)), y_{p,t}); \boldsymbol{A}_t, \boldsymbol{\theta}_p, \boldsymbol{\theta}_M), \boldsymbol{f}; \phi'), y_{p,t+T})
\end{aligned} \tag{9.7}$$

9.2.4 Training the FORECASTABM

However, directly training the FORECASTABM as illustrated in Figure 9.1 is extremely hard, and the experiment results are not good. Intuitively, if we use T' days for ABM calibration (i.e., used in the calibration module) and forecast for T days ahead (i.e., for the forecasting module), the depth of the framework becomes at least $T + T'$. For example, if we utilize contact networks from the past $T' = 28$ days and aim to forecast for $T = 14$ days, there will be at least $T + T' = 42$ layers in the FORECASTABM framework, making the training process highly complex and computationally intensive. Moreover, very deep neural networks are more likely to encounter the vanishing or exploding gradient problem, where the gradients for backpropagation can either become too small or too large, which cannot lead to meaningful updates of the parameter values.

To address this problem, we adopt an iterative training approach following [262]: Specifically, we iteratively freeze the calibration module and the forecasting module during training, which allows us to simplify training FORECASTABM by breaking it down into two substeps: (1) First, we focus on training the calibration module while keeping the forecasting module fixed (i.e., freeze the ABM simulator for forecasting and the adapter ϕ', but activate the ABM simulator for calibration and neural network ϕ). This allows us to learn reasonable ABM general parameters $\boldsymbol{\theta}_M$ and patient-specific parameters $\boldsymbol{\theta}_p$, and therefore $y_{p,t}$. (2) Next, we focus on training the forecasting module while keeping the calibration module fixed (i.e., freeze the ABM simulator for calibration and neural network ϕ, but activate the ABM simulator for forecasting and the adapter ϕ'). This allows us to learn reasonable adapter parameters ϕ'. Since we already learned a good $y_{p,t}$ and $\boldsymbol{\theta}_M$ in substep 1, even by assuming future contact networks will repeat what we have for day t, we can still have a not-too-low-quality $\hat{y}'_{p,t}$ before the adapter's revision, and a high-quality $\hat{y}_{p,t}$ with the help of the adapter. We repeat the two substeps above for different t until convergence. We also show the pseudocode in Algorithm 10.

9.3 Results

We show the performance of our FORECASTABM in forecasting MRSA infection cases in the ICUs of the UVA hospital in October to December 2019. Specifically, we use the EHR data from the UVA hospital to construct the contact networks, collect patient risk factors, and identify the ground-truth MRSA cases from the lab test results. As mentioned in section 9.2, we use the 2-MODE-SIS model [68, 30] to model the MRSA spread. When forecasting for each week k, we used the contact networks, patient risk factors, and lab test results until week $k-2$ to train the FORECASTABM (i.e., we are forecasting for $T = 14$ days ahead). Note that only data prior to week $k-2$ are used in this process—this is similar

Algorithm 10 Training FORECASTABM

Inputs: Risk factors of patients $\boldsymbol{f}$, adjacency matrices of contact networks $\boldsymbol{A}$, lab testing results y
for each timestep t **do**
  for each epoch **do**
    Estimate the patient-specific parameters and ABM parameters $\boldsymbol{\theta}_p, \boldsymbol{\theta}_M = \mathrm{NN}(\boldsymbol{f}; \phi)$
    Estimate the patient states on day t: $\hat{y}_{p,t} = \mathrm{ABM4Calibration}(\boldsymbol{A}; \boldsymbol{\theta}_p, \boldsymbol{\theta}_M)$
    Compute the loss $\mathcal{L}(\hat{y}_{p,t}, y_{p,t})$
    Update the neural network parameters ϕ
  end for
  for each epoch **do**
    Roughly forecast the patient states on day $t + T$:
    $\hat{y}'_{p,t+T} = \mathrm{ABM4Forecasting}(\hat{y}_{p,t}; \boldsymbol{A}_t, \boldsymbol{\theta}_p, \boldsymbol{\theta}_M)$
    Use adapter to revise the $\hat{y}'_{p,t+T}$ to get $\hat{y}_{p,t+T} = \mathrm{Adapter}(\hat{y}'_{p,t+T}, \boldsymbol{f}; \phi')$
    Compute the loss $\mathcal{L}(\hat{y}_{p,t}, y_{p,t}) + \mathcal{L}(\hat{y}_{p,t+T}, y_{p,t+T})$
    Update the adapter parameters ϕ'
  end for
end for
Outputs: Neural network parameters ϕ, and adapter parameters ϕ'

to the setting considered in [10] for detection of asymptotic cases (though their ABM is different and they are doing retrospective studies). We also compared NEURABM with feedforward neural network [91], decision tree [256], and Naive Bayes [257]. Specifically, since most patients' infection state may not change within 14 days, we focus on the patients who change their states (i.e., newly infected or recovered) and evaluate the performance.

In Table 9.1, we list the precision, false-negative rate, F1 score, and accuracy for each method. We can see that FORECASTABM always achieves the highest precision, F1 score, accuracy, and lowest false-negative rate, indicating the effectiveness of our framework in forecasting future MRSA cases. Besides, the naive ABM-based methods that assume future contact networks will repeat what we have for today could only achieve a low accuracy of 0.05, which showcases that they cannot be helpful without our FORECASTABM design. Moreover, there are 90 patients who changed their states from October to December 2019. Our FORECASTABM could forecast 22 more patients accurately compared with the best baselines, accounting for 24% of newly infected or recovered patients. Note that in clinical practice, a low false-negative rate means that when we predict a patient as negative, he

Table 9.1: Forecasting future MRSA cases

	Precision	False-negative rate (1-recall)	F1 score	Accuracy
Naive Bayes	0.9077	0.2976	0.7919	0.6555
Decision Tree	0.8846	0.4524	0.6765	0.5111
Feedforward neural network	0.8421	0.6190	0.5246	0.3555
FORECASTABM	**0.9412**	**0.0476**	**0.9467**	**0.9000**

or she is less likely to be an actually positive patient that we fail to identify. We can see that FORECASTABM's false-negative rate is the lowest (< 0.05) compared to other baselines, indicating that our FORECASTABM can forecast future MRSA cases well with fewer missing/undetected patients.

9.4 Discussion

FORECASTABM is a novel framework that integrates both ABMs and neural networks to forecast future MRSA cases. Compared with solely ABM-based methods, our FORECASTABM bypasses the problem that ABMs rely on future contact networks in hospitals to run simulations and forecast. Meanwhile, compared with machine learning-based methods, our framework also incorporates the domain knowledge encoded in the ABMs, which better captures the spread dynamics of MRSA and provides more explainability.

Our experiments also show that FORECASTABM can better forecast future MRSA cases than other baselines in UVA ICUs. Note that most patients may not change their patient states in a short time like 7 or 14 days, so most existing methods tend to forecast patients' future infection states as what they are having now. However, our FORECASTABM overcomes this issue, and we demonstrate that we can perform better (with higher precision, F1 score, accuracy, and lower false-negative rate) on such patients. Moreover, our false-negative rate is significantly low (< 0.05), indicating that our FORECASTABM can forecast future cases well with fewer missing/undetected patients.

However, this framework is not without limitations. One limitation is that for the ABM simulator 2, we just naively assume that the future contact networks will repeat the contact networks for day t. Although in the forecasting setup we can never access the ground-truth future contact networks to run ABM simulations, a good machine learning-based algorithm to also forecast future contact networks could be helpful. Moreover, the adapter network we have now only takes the rough patient states forecast $\hat{y}'_{p,t}$ on day t and patients' risk factor data f_p as input. However, a time-series forecasting model could be more suitable. Besides, the 2-MODE-SIS model we use now just assumes that patients are either in the *susceptible* or *carriage* state. However, there are multiple types of MRSA infection (e.g., skin abscess or bloodstream infection) and colonization, and we are not distinguishing them clearly. Moreover, as described in section 9.2, this framework is extremely deep, and we use freezing techniques to help the training, while other deep neural network training techniques could also be helpful.

Nevertheless, FORECASTABM is still a general framework that opens up a new stream of work to use both ABMs and deep neural networks to forecast future MRSA cases. Since our framework is a general one that is not restricted to a specific kind of epidemiological model (e.g., 2-MODE-SIS model in this paper), people can replace it with any models they are interested in and study other forecasting problems.

CHAPTER 10

CONCLUSIONS AND FUTURE DIRECTIONS

In this chapter, I will summarize the contributions of this thesis. Specifically, the contributions can be classified into computing and public health contributions. Then, I will introduce the potential future works that follow from this thesis.

10.1 Computing Contributions

- **Network spreading models on heterogeneous hypergraphs:** Conventional heterogeneous agent-based models (HABMs) on graphs can only capture pairwise interactions between agents but fail to capture the more complex dynamics of group interactions (e.g., multiple people in the same location simultaneously), leading to suboptimal performance. However, switching from HABMs to hypergraph-based HABMs is non-trivial since the dynamics of group interactions cannot simply be considered as the linear combination of each pairwise interaction. To address this, we proposed a group-level nonlinear load discounting method and extended the current graph-based HABMs to hypergraphs. We show that such hypergraph-based HABMs can capture structural differences better than graph-based HABMs.

- **Data-driven parameter adjustment method:** One of the biggest challenges when using epidemiological models for what-if forecasting is that we can never know the new parameters under a new scenario. To tackle this, we proposed a data-driven parameter adjustment method that focuses on the most significant parameter changes under different what-if policies. By using historical clinical data, we adjust the calibrated parameters for the current policy to estimate the new values.

- **Network algorithms, spectral analysis of heterogeneous models:** We provide a novel spectral characterization of a heterogeneous model on underlying time-varying people-location heterogeneous contact networks using stability analysis of discrete-time nonlinear dynamical systems. First, we approximate the propagation dynamics of the model on heterogeneous networks, allowing us to analyze the stability of its equilibrium point (specifically, the all-zero point where the disease dies out) and determine whether this point is stable or not. We show that this characterization is meaningful in capturing a "tipping point" for HAI spread dynamics. At this tipping point, there is a phase transition for the fraction of infected patients at the end: when it is larger than this tipping point, the fraction of infected patients will be close to 1. When it is smaller than this point, almost none of the patients will be infected (i.e., the epidemic dies out).

- **Differentiable ABM for high-granularity predictions:** Current methods for HAI predictions can be classified into two categories: mechanistic transmission modeling-based methods and machine learning-based methods. However, these methods suffer from different drawbacks: transmission modeling-based methods do not fully utilize the rich data contained in electronic health records (EHR), while machine learning-based methods typically lack information about mechanistic processes. To address this, I propose integrating neural networks and epidemiological models into one joint framework. This approach leverages the flexibility of machine learning tools to learn from diverse data types for patient-specific predictions while incorporating the domain knowledge embedded in epidemiological models.

- **Decision-focused learning on non-convex problems:** Most current Decision-Focused Learning (DFL) methods are built on KKT conditions and restricted to convex problems. They also cannot account for uncertainties. Therefore, we propose a general and efficient DFL method for stochastic optimization using energy-based models, allowing us to explicitly parameterize the original problem and uncertainties using a differentiable optimization layer based on energy functions.

10.2 Public Health Contribution

- **Improved tracing and forecasting for HAIs:** With the hypergraph-based heterogeneous agent-based models (HABMs) we designed, we can better capture structural differences compared to graph-based HABMs. We analyzed the contact networks in the University of Virginia (UVA) hospital before and during the first wave of the COVID-19 outbreak in Virginia and demonstrated that hypergraphs are more effective in capturing structural differences between the two contact networks than graphs. We also showed that the hypergraph-based HABM leads to better fitting and forecasting of MRSA cases in the UVA hospital than graph-based models. Additionally, it provides good interpretability of the MRSA outbreak in the UVA hospital.

- **More efficient discharge policy from contact precaution:** We studied the potential cost-effectiveness of changing the current policy at the University of Virginia Hospital, which requires three consecutive negative MRSA surveillance tests to remove precautions, to either a two-negative clearance policy or a one-negative clearance policy. Using the proposed data-driven parameter adjustment method, we quantitatively examined the tradeoff between the number of additional MRSA cases and the number of precaution days saved under the relaxed policies. We calculated the total cost for different policies. Our study indicates that a single negative MRSA nares PCR test may provide sufficient evidence to discontinue MRSA contact precautions, and that the one-negative policy may be the most cost-effective option.

- **Theoretical explanation of the "dilution effect":** We accurately captures and helps explain counter-intuitive behaviors of pathogen load-based propagation that have been observed before, specifically the so-called "dilution effect". The dilution effect refers to the observation that when the number of edges in the network increases, the number of infections decreases. This is surprising because it goes against the intuition that more contacts lead to more infections. This also contrasts with previous Independent Cascade/Linear Threshold/SIR style models, where adding an edge leads to more infections. However, our method correctly captures this dynamic. The reason is that with more edges in the graph, more pathogen is dispersed, resulting in lower loads on patients and therefore fewer infections.

- **Better forecasting for COVID-19:** Our approach not only leads to substantially better estimates of the total number of infections but also provides better forecasts of infections compared to standard model calibration-based methods. Additionally, we demonstrate how our learned parameterization helps model more accurate what-if scenarios with non-pharmaceutical interventions. Our results support earlier findings that most COVID-19 infections were unreported and that non-pharmaceutical interventions indeed helped mitigate the spread of the outbreak. Our approach provides a general method for improving epidemic modeling that is broadly applicable.

- **More accurate identification of HAI importation cases in ICUs:** Our framework identifies not only MRSA importation cases but also nosocomial infection cases in UVA ICUs, with performance that can be considered clinically useful. In contrast, prior methods using EHR data have not achieved this level of performance. Moreover, the negative predictive value (NPV) for our method is always higher than 0.95, indicating that we can identify these cases well with fewer missed/undetected patients. Our case study also reveals the risk factors that are highly related to MRSA importations, allowing clinicians to better respond to potential MRSA importation cases in their hospital.

10.3 Future Directions

10.3.1 Epidemiological Modeling and Decision-making

Clinical decision-making needs to be specific at the patient level. However, many current modeling methods struggle to capture the heterogeneity of patients and distinguish them well. For example, although many HAI models can estimate a general recovery rate for all patients, they cannot predict who will exactly recover in the next several days. Therefore, a more effective model with higher-granularity will give patient-specific forecasts and better support clinical decisions.

Another dimension of my future work involves integrating Large Language Models (LLMs) for real-time processing of clinical data, such as doctors' notes. Currently, the manual processing of these clinical data delays modeling and response. Automating this process using LLMs can significantly reduce response times, enabling quicker and more effective interventions during outbreaks.

Looking forward, the privacy concerns prevent hospitals and clinicians from sharing their data. Instead, generative models could mimic real-world data and generate new synthetic datasets without leaking sensitive patient information. Moreover, the size of the real-world data from one hospital may also be too small, such that large models may easily overfit. However, we could generate arbitrarily large synthetic datasets using generative models for training and testing.

10.3.2 Modeling and Decision-making using Machine Learning

In recent years, large datasets covering large populations (e.g., in the COVID-19 pandemic, Facebook surveyed millions of people every day across the US to see if they show COVID-related symptoms like cough or fever) and large machine learning frameworks make patient-specific predictions at the country level realistic. However, the scalability of the current ABMs [32] is still restricted by the simulation process. Therefore, one interesting direction is to extend the current framework to large datasets at the country level, covering hundreds of millions of people. This allows us to better detect and respond to community infectious disease outbreaks, leading to better surveillance and control.

Looking forward, a critical challenge in the clinical application of AI tools is their lack of explainability. Many AI frameworks, despite their high performance, operate as "black boxes", which is not conducive to clinical practices that demand transparency for safety assurance. Therefore, an interesting direction will be enhancing the explainability and traceability of AI tools designed for epidemiological modeling, thereby fostering greater acceptance and application by clinicians.

10.3.3 Translating to Implementation in Real-world Hospitals

In chapter 8, we proposed the NEURABM framework that integrates both deep neural networks and agent-based models together, opening up a new stream of works to utilize the rich clinical datasets and domain knowledge encoded in the ABMs simultaneously to capture the HAI spread and guide clinical decisions at the patient level. Despite the promising results as we have shown in chapter 8, we believe the potential of the NEURABM framework is far from fully explored. From the data perspective, the NEURABM framework provides the flexibility of digesting any type and format of dataset as input. However, this also leads to many data quality (is the data good enough or useful) and learnability (even if the data is of high quality, can our framework benefit from it) issues, which are open to solve. Moreover, since our NEURABM framework couples both deep neural networks and agent-based models, potential improvements on the framework can also start from either of them. Besides, when evaluating the performance of a proposed methodology, clinical trials are definitely the most convincing way to showcase the results. However, clinical trials can only be implemented when we already have some level of confidence. An interesting direction to explore is how we can build up this confidence before conducting the trials.

Appendices

APPENDIX A
SUPPLEMENTARY MATERIAL FOR CHAPTER 3: H^{2}ABM: HETEROGENEOUS AGENT-BASED MODEL ON HYPERGRAPHS TO CAPTURE GROUP INTERACTIONS

A.1 Hypergraph Motifs

The information presented here is a restatement of the work from [131]. For additional details, such as implementation, applications and evaluation of this method, we ask the reader to explore that work further. We include the following information solely to serve as additional background information to understand the implications of the conclusions drawn from subsection 3.3.2.

As stated earlier, each motif represents all of the connectivity patterns possible for the overlap of any 3 hyperedges h_1, h_2, h_3. As seen in Figure A.1, there are seven subsets possible for these vertices, $h_1 \setminus h_2 \setminus h_3$, $h_2 \setminus h_1 \setminus h_3$ and $h_3 \setminus h_2 \setminus h_1$ in green, $h_1 \cap h_2 \setminus h_3$, $h_2 \cap h_3 \setminus h_1$, $h_3 \cap h_1 \setminus h_2$ in blue and $h_1 \cap h_2 \cap h_3$ in red. In total, there are 2^7 possible combinations for the emptiness and nonemptiness of these three sets. However, once you remove the duplicate and symmetric combinations you arrive at a total of 26 possible motifs.

Hypergraph motifs are a useful tool for fine grained structural analysis of hospital traces because of the following properties.

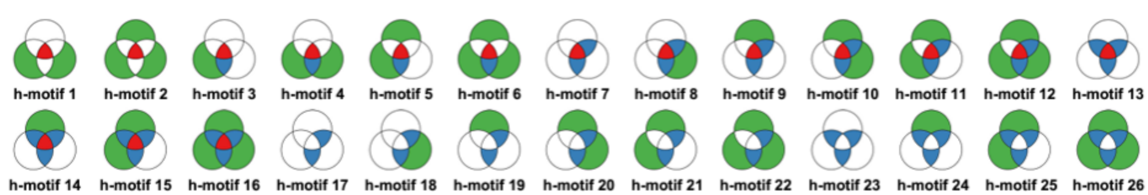


Figure A.1: **The 26 Hypergraph Motifs.** Note that this is Figure 3 in [131].

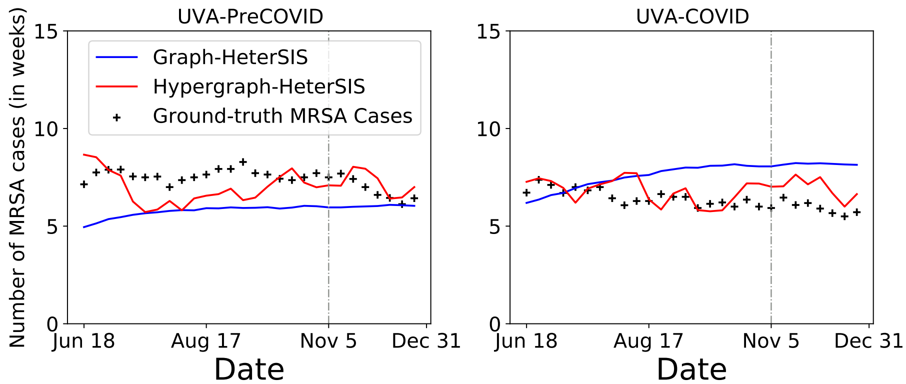


Figure A.2: **An example to showcase that HYPERGRAPH-HETERSIS gives a closer estimation for the number of MRSA cases than GRAPH-HETERSIS.** Here, we use the number of MRSA cases before the grey vertical dash line as the training data to calibrate each model, and forecast the future MRSA cases (testing period). The red and blue curves represent HYPERGRAPH-HETERSIS and GRAPH-HETERSIS model estimated MRSA cases. The black + symbols represent the ground-truth number of MRSA cases.

- *Exhaustive* - The 26 motifs encompass all of the possible connection patterns between three hyperedges.
- *Unique* - Each connectivity pattern is linked to just one motif.
- *Size Invariant* - These motifs are independent of the number of vertices in each hyperedge.

A.2 Example Calibration Results

To intuitively showcase that HYPERGRAPH-HETERSIS gives a better calibration and forecast for the number of MRSA cases than GRAPH-HETERSIS, we use a example that calibrate between Jun 18 to Nov 5, and forecast between Nov 5 to Dec 31 in Figure A.2. Here, the grey vertical dash line divides the calibration period (left) and the forecasting period (right). The black + symbols represent the ground-truth number of MRSA cases, and the red and blue curves represent HYPERGRAPH-HETERSIS and GRAPH-HETERSIS estimated MRSA cases. As seen in the figure, the red lines align more closely with the black plus symbols than the blue lines during the calibration period, which indicates that

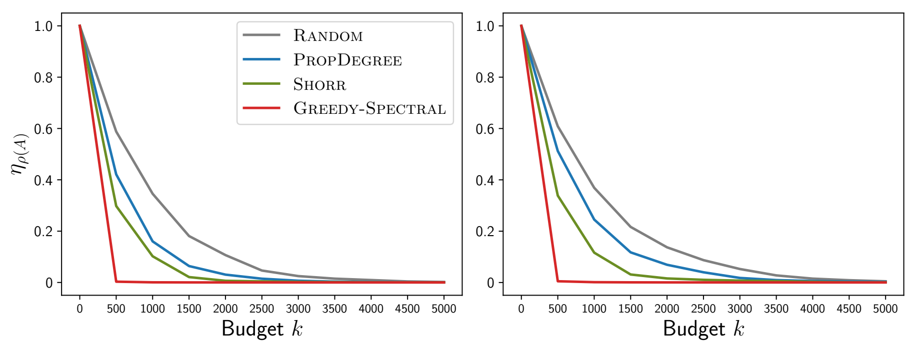
HYPERGRAPH-HETERSIS model gives a better calibration and forecasting to the number of MRSA cases than GRAPH-HETERSIS model.

APPENDIX B

SUPPLEMENTARY MATERIAL FOR CHAPTER 4: USING SPECTRAL CHARACTERIZATION TO IDENTIFY HEALTHCARE-ASSOCIATED INFECTION (HAI) PATIENTS FOR CLINICAL CONTACT PRECAUTION

B.1 Additional Results

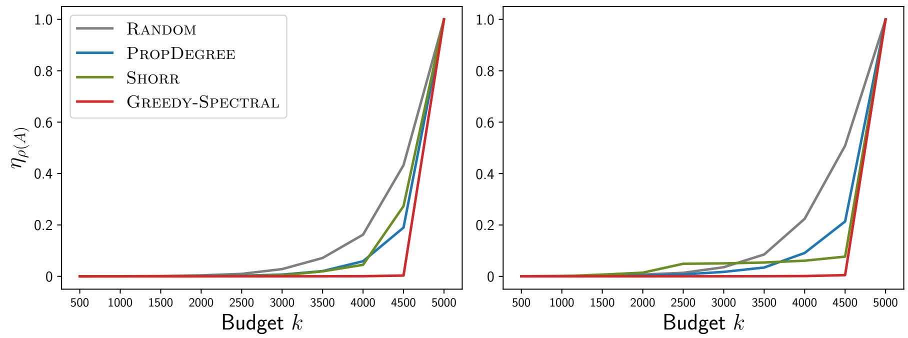
First, we show how GREEDY-SPECTRAL performs in minimizing $\rho(\boldsymbol{A})$ for Isolation problem. As seen in Figure B.1, our algorithm achieves lower $\eta_{\rho(\boldsymbol{A})}$ than RANDOM, PROPDEGREE, and SHORR on both UVA-PRECOVID and UVA-COVID dataset, as isolation budget k increases. This demonstrates the effectiveness of our algorithm in minimizing $\rho(\boldsymbol{A})$.



(a) UVA-PRECOVID (b) UVA-COVID

Figure B.1: **Change in $\eta_{\rho(\boldsymbol{A})}$ (y-axis) with increasing isolation budget k (x-axis).** (a) is for UVA-PRECOVID and (b) is for UVA-COVID. Lower is better.

Next, we show how GREEDY-SPECTRAL performs in minimizing $\rho(\boldsymbol{A})$ for Clearance problem. As seen in Figure B.2, our algorithm achieves lower $\eta_{\rho(\boldsymbol{A})}$ than RANDOM, PROPDEGREE, and SHORR on both UVA-PRECOVID and UVA-COVID dataset, as clearance budget k increases. This demonstrates the effectiveness of our algorithm in minimizing $\rho(\boldsymbol{A})$.



(a) UVA-PRECOVID (b) UVA-COVID

Figure B.2: **Change in $\eta_{\rho(\boldsymbol{A})}$ (y-axis) with increasing clearance budget k (x-axis).** (a) is for UVA-PRECOVID and (b) is for UVA-COVID. Lower is better.

B.2 Parameter Setup

As mentioned in chapter 4, for high precaution and cleaning scenario, we calibrate on the weekly number of new tested positive MRSA cases in UVA hospital to get the parameters for 2-MODE-SIS model. For calibration, we use the Approximate Bayesian Computation-Rejection scheme implemented in the ABCPY package [263]. The first samples a parameter set Θ_i from a user-defined prior, runs multiple simulations with Θ_i, and then stores Θ_i if the average simulation results are close to actual observations. The final list of Θ_i's then approximates the posterior distribution given actual data. When evaluating which Θ_i to store, we use cumulative weekly counts of new MRSA infections and measure the RMSE error between simulated and actual values. Note that in actual data, only patients who were negative in a previous test but were positive in a later test count as *new* MRSA infections.

Here, we show the effectiveness of the calibration from two perspectives: On the one hand, we use some synthetic parameters to simulate some synthetic MRSA case curves. We then calibrate on these synthetic curves to show that our could ”recover” synthetic parameters via calibration. On the other hand, we also calibrate on the real-world MRSA case curves, and show the simulated curves using our calibrated parameters align closely with the real-world curves.

In Figure B.3, we set up 5 different ”scenarios” for synthetic experiments. In each

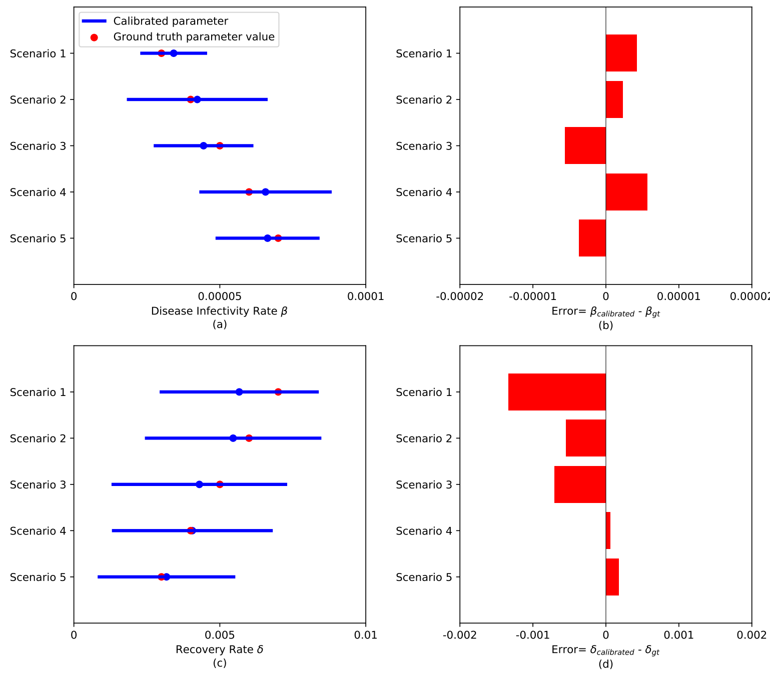


Figure B.3: **The calibration effectively ”recovers” the synthetic parameters by calibrating on the simulated curve using synthetic parameters.** Left: the red dots are the ground-truth values of the synthetic parameters, and the blue dots (as well as error bars) are the average value and standard error of the calibrated parameters.

scenario, we set some “ground truth” parameters and run simulations with these ground-truth parameters to generate synthetic MRSA cases curve. Next, we use these curves for calibration to get the calibrated parameters. If the calibrated parameters closely align with the ground truth parameters we initially set (i.e., we can ”recover” these parameters through calibration), we can conclude that the calibration process is effective.

APPENDIX C
SUPPLEMENTARY MATERIAL FOR CHAPTER 6: ACCURATELY ESTIMATING TOTAL COVID-19 INFECTIONS USING INFORMATION THEORY

C.1 Cost Derivation

Here, we show the cost for each component in Equation 6.4 and give our encoding method explicitly:

1. $\textsc{Cost}(\hat{\Theta})$: We represent $\hat{\Theta}$ as a vector of real numbers. We describe our encoding later below.

2. $\textsc{Cost}(\Theta' - \hat{\Theta}|\hat{\Theta})$: We will encode the difference of two vectors as a vector of real numbers.

3. $\textsc{Cost}(\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta', \hat{\Theta})$: Here, we encode the difference between the two time sequences: $\alpha'_{\text{reported}} \times D$ given $D_{\text{reported}}(\hat{\Theta})$.

4. $\textsc{Cost}(\frac{D-D_{\text{reported}}}{1-\alpha'_{\text{reported}}} - D(\Theta')|D, \Theta', \hat{\Theta})$: Again, we encode it as a difference between the two time sequences: $\frac{D_{\text{unreported}}}{1-\alpha'_{\text{reported}}}$ given $D(\Theta')$.

Next, we describe the encoding cost of real numbers, vectors, and the difference between two time sequences.

C.1.1 Encoding Integers

To encode a positive integer n^+, we encode both the binary representation of integer n^+ as well as the length of the representation $\log_2 n^+$. Following [264], the cost in bits of encoding a single integer n is as follows:

$$\textsc{Cost}(n^+) = \log_2 c_0 + \log^*(n^+). \tag{C.1}$$

where $c_0 \approx 2.865$ and $\log^*(n^+) = \log_2 n^+ + \log_2 \log_2 n^+ + \cdots$ as described in [264]. There are infinite terms in $\log^*(n^+)$ function since after we encode a number, we always need to encode its length as another number, which could be repeated for infinite times. Additionally, if we want to transmit an integer that can be either positive or negative, we can add another sign bit and therefore the cost in bits for integers will be

$$\textsc{Cost}(n) = \textsc{Cost}(|n|) + 1. \tag{C.2}$$

C.1.2 Encoding Real Numbers

Note that most real numbers (e.g. π or e) need infinite number of bits to encode. Hence, we introduce a precision threshold δ. With threshold δ, we approximate a positive real number x^+ with x_δ which satisfies $|x^+ - x_\delta| < \delta$, and we encode x_δ instead. To encode x_δ, we encode both the integer part $\lfloor x^+ \rfloor$ as well as the fractional part $x_\delta - \lfloor x^+ \rfloor$. Hence the cost in bits of encoding a positive real number x^+ is as follows:

$$\textsc{Cost}(x^+) = \textsc{Cost}(\lfloor x^+ \rfloor) + \log_2 \frac{1}{\delta} \tag{C.3}$$

where $\lfloor x^+ \rfloor$ is the floor of x^+ and therefore is a integer, whose encoding cost is $\textsc{Cost}(\lfloor x^+ \rfloor) = \log_2 c_0 + \log^*(\lfloor x^+ \rfloor)$. Additionally, if we want to transmit a real number that can be either positive or negative, we can add another sign bit and therefore the cost in bits for real numbers will be

$$\textsc{Cost}(x) = \textsc{Cost}(|x|) + 1 \tag{C.4}$$

C.1.3 Encoding Vectors

To encode a vector $\Theta = [\Theta[1], \Theta[2], \cdots, \Theta[n]]$, we encode every components one by one as real numbers. Hence the cost in bits of encoding a vector Θ is as follows:

$$\textsc{Cost}(\Theta) = \textsc{Cost}(\Theta[1]) + \textsc{Cost}(\Theta[2]) + \cdots + \textsc{Cost}(\Theta[n]) \tag{C.5}$$

C.1.4 Encoding the Difference between Two Time Sequences

To encode the difference $A - B = [A_{t_1} - B_{t_1}, A_{t_2} - B_{t_2}, \cdots, A_{t_n} - B_{t_n}]$ between two time sequences $A = [A_{t_1}, A_{t_2}, \cdots, A_{t_n}]$ and $B = [B_{t_1}, B_{t_2}, \cdots, B_{t_n}]$, we encode every components one by one as real numbers. Hence the cost in bits of encoding the difference is as follows:

$$\textsc{Cost}(A - B) = \textsc{Cost}(A_{t_1} - B_{t_1}) + \textsc{Cost}(A_{t_2} - B_{t_2}) + \cdots + \textsc{Cost}(A_{t_n} - B_{t_n}) \tag{C.6}$$

APPENDIX D

SUPPLEMENTARY MATERIAL FOR CHAPTER 8: IDENTIFYING IMPORTATION AND ASYMPTOMATIC SPREADERS OF MULTI-DRUG RESISTANT ORGANISMS IN HOSPITAL SETTINGS

D.1 Patient Risk Factor Data

This dataset consists of risk factors for patients in intensive care units (ICUs). We extracted 19 different risk factors from the EHR for each patient, which are all available prior to MRSA testing. All patients received an MRSA test within $(t-3, t+3)$ days of being admitted into one of the ICUs, and patients who tested positive for MRSA within this range are considered as importation cases. From July 1, 2019, to December 31, 2019, 2470 patients were in the UVA ICUs, with 157 identified as importation cases. The following provides a definition for each risk factor and the method of their derivation:

- **MRSA nares**: Binary variable indicating whether the ICU admission to be administered is a nares PCR test or a clinical culture test. Negative clinical culture tests should not be used to indicate the absence of MRSA.

- **Device (e.g., central line)**: Binary variable that indicates whether a patient had a catheter, tube, or blood line (defined as "device") administered to them in the UVA hospital system.

- **Age**: Integer value indicating the age of the patient in years.

- **Admit from ED**: Binary variable indicating that the type of admission to the hospital was defined as an emergency.

- **Admit from healthcare facility**: A variable in $-1, 0, 1$ indicating where the patient was admitted from. If the patient was admitted from a skilled nursing facility or a long-term acute care hospital, it is 1; if admitted from somewhere besides these facilities, it is -1. It is 0 if the admission location is unknown.

- **Prior discharge to facility**: A variable in $-1, 0, 1$ based on a history of a patient's prior discharges from the UVA hospital system. It is 1 if the patient had previously been discharged to a skilled nursing facility or a long-term acute care hospital and -1 if they had been previously discharged but not to one of these facilities. It is 0 if there is no prior discharge data for the patient.
- **Female sex**: Binary variable indicating that the gender of the patient is female.
- **Recent surgery**: Binary variable indicating whether the patient has had a surgery performed at any time in the UVA hospital system before the date of the ICU admission.
- **Male sex**: Binary variable indicating that the gender of the patient is male.
- **Device usage within 90 days**: Binary variable indicating whether a device has been administered to the patient within the 90 days prior to the ICU admission.
- **Surgery history within 90 days**: Binary variable indicating whether the patient has had a surgery performed in the UVA hospital system within the last 90 days of the ICU admission.
- **Surgery history 90 days ago**: Binary variable indicating if the patient has had surgery in the UVA hospital system more than 90 days prior to the ICU admission.
- **Device usage 90 days ago**: Binary variable indicating whether a device has been administered to the patient more than 90 days prior to the ICU admission.
- **Length of stay**: Integer value defining how long a patient's current stay in the hospital has been before the ICU admission.

- **MRSA contacts within 7 days**: Integer value defining the number of contacts with an individual whose known MRSA status was positive over the 7 days prior to the test administration, as defined by the daily contact networks.
- **MRSA contacts within 14 days**: Integer value defining the number of contacts with an individual whose known MRSA status was positive over the 14 days prior to the test administration, as defined by the daily contact networks.
- **Elective admission**: Binary variable indicating that the type of admission to the hospital was defined as elective.
- **Urgent admission**: Binary variable indicating that the type of admission to the hospital was defined as urgent.
- **Routine admission**: Binary variable indicating that the type of admission to the hospital was defined as routine.

REFERENCES

[1] B. N. Doebbeling and R. P. Wenzel, "The direct costs of universal precautions in a teaching hospital," *Jama*, vol. 264, no. 16, pp. 2083–2087, 1990.

[2] E. M. Martin *et al.*, "Noninfectious hospital adverse events decline after elimination of contact precautions for mrsa and vre," *infection control & hospital epidemiology*, vol. 39, no. 7, pp. 788–796, 2018.

[3] L. Valiquette, C. N. A. Chakra, and K. B. Laupland, "Financial impact of health care-associated infections: When money talks," *Canadian Journal of Infectious Diseases and Medical Microbiology*, vol. 25, no. 2, pp. 71–74, 2014.

[4] H. L. Evans *et al.*, "Contact isolation in surgical patients: A barrier to care?" *Surgery*, vol. 134, no. 2, pp. 180–188, 2003.

[5] H. Day *et al.*, "Do contact precautions cause depression? a two-year study at a tertiary care medical centre," *Journal of Hospital Infection*, vol. 79, no. 2, pp. 103–107, 2011.

[6] P. Mehrotra *et al.*, "Effects of contact precautions on patient perception of care and satisfaction: A prospective cohort study," *Infection Control & Hospital Epidemiology*, vol. 34, no. 10, pp. 1087–1093, 2013.

[7] A. M. Sjoberg *et al.*, "One health interprofessional stewardship to combat antimicrobial resistance," *Nature Medicine*, vol. 29, no. 3, pp. 512–513, 2023.

[8] C. Marshall, M. Richards, and E. McBryde, "Do active surveillance and contact precautions reduce mrsa acquisition? a prospective interrupted time series," *PloS one*, vol. 8, no. 3, e58112, 2013.

[9] M. Haque *et al.*, "Strategies to prevent healthcare-associated infections: A narrative overview," *Risk management and healthcare policy*, pp. 1765–1780, 2020.

[10] S. Pei, F. Liljeros, and J. Shaman, "Identifying asymptomatic spreaders of antimicrobial-resistant pathogens in hospital settings," *Proceedings of the National Academy of Sciences*, vol. 118, no. 37, e2111190118, 2021.

[11] S. Karanika, S. Paudel, C. Grigoras, A. Kalbasi, and E. Mylonakis, "Systematic review and meta-analysis of clinical and economic outcomes from the implementation of hospital-based antimicrobial stewardship programs," *Antimicrobial agents and chemotherapy*, vol. 60, no. 8, pp. 4840–4852, 2016.

[12] V. Mouajou, K. Adams, G. DeLisle, and C. Quach, "Hand hygiene compliance in the prevention of hospital-acquired infections: A systematic review," *Journal of Hospital Infection*, vol. 119, pp. 33–48, 2022.

[13] P. De Nardo *et al.*, "Multi-criteria decision analysis to prioritize hospital admission of patients affected by covid-19 in low-resource settings with hospital-bed shortage," *International Journal of Infectious Diseases*, vol. 98, pp. 494–500, 2020.

[14] E. G. Mendoza, E. Rojas, L. L. Tesar, and J. Zhang, "A macroeconomic model of healthcare saturation, inequality and the output–pandemia trade-off," *IMF Economic Review*, vol. 71, no. 1, p. 243, 2023.

[15] H. C. Cardenas, R. T. Carson, M. Hanemann, J. J. Louviere, and D. Whittington, "Who should get the scarce icu bed? the us public's view on triage in the time of covid-19," *Emergency Medicine Journal*, vol. 39, no. 2, pp. 94–99, 2022.

[16] Y. Bai *et al.*, "Presumed asymptomatic carrier transmission of covid-19," *JAMA*, vol. 323, no. 14, pp. 1406–1407, 2020.

[17] J. B. Aguilar, J. S. Faust, L. M. Westafer, and J. B. Gutierrez, "Investigating the impact of asymptomatic carriers on covid-19 transmission," *MedRxiv*, 2020.

[18] S. Stockmaier, N. Stroeymeyt, E. C. Shattuck, D. M. Hawley, L. A. Meyers, and D. I. Bolnick, "Infectious diseases and social distancing in nature," *Science*, vol. 371, no. 6533, 2021.

[19] J. Shaman, "An estimation of undetected covid cases in france," *Nature*, vol. 590, pp. 38–39, 7844 2020.

[20] R. Li *et al.*, "Substantial undocumented infection facilitates the rapid dissemination of novel coronavirus (sars-cov-2)," *Science*, vol. 368, no. 6490, pp. 489–493, 2020.

[21] A. J. Kallen *et al.*, "Health care–associated invasive mrsa infections, 2005-2008," *Jama*, vol. 304, no. 6, pp. 641–647, 2010.

[22] J. Fernandez-Gracia, J.-P. Onnela, M. L. Barnett, V. M. Eguiluz, and N. A. Christakis, "Influence of a patient transfer network of us inpatient facilities on the incidence of nosocomial infections," *Scientific reports*, vol. 7, no. 1, p. 2930, 2017.

[23] L. E. Rocha, V. Singh, M. Esch, T. Lenaerts, F. Liljeros, and A. Thorson, "Dynamic contact networks of patients and mrsa spread in hospitals," *Scientific reports*, vol. 10, no. 1, pp. 1–10, 2020.

[24] A. Bhalla *et al.*, "Acquisition of nosocomial pathogens on hands after contact with environmental surfaces near hospitalized patients," *Infection Control & Hospital Epidemiology*, vol. 25, no. 2, pp. 164–167, 2004.

[25] W. C. Albrich and S. Harbarth, "Health-care workers: Source, vector, or victim of mrsa?" *The Lancet infectious diseases*, vol. 8, no. 5, pp. 289–301, 2008.

[26] G. M. Snyder *et al.*, "Detection of methicillin-resistant staphylococcus aureus and vancomycin-resistant enterococci on the gowns and gloves of healthcare workers," *Infection Control & Hospital Epidemiology*, vol. 29, no. 7, pp. 583–589, 2008.

[27] N. Plipat, I. H. Spicknall, J. S. Koopman, and J. N. Eisenberg, "The dynamics of methicillin-resistant staphylococcus aureus exposure in a hospital model and the potential for environmental intervention," *BMC infectious diseases*, vol. 13, no. 1, pp. 1–11, 2013.

[28] M. A. Rubin *et al.*, "A simulation-based assessment of strategies to control clostridium difficile transmission and infection," *PloS one*, vol. 8, no. 11, e80671, 2013.

[29] R. E. Nelson *et al.*, "An economic analysis of strategies to control clostridium difficile transmission and infection using an agent-based simulation model," *PloS one*, vol. 11, no. 3, e0152248, 2016.

[30] J. Cui*, S. Cho*, M. Kamruzzaman, M. Bielskas, A. Vullikanti, and B. A. Prakash, "Using spectral characterization to identify healthcare-associated infection (hai) patients for clinical contact precaution," *Scientific Reports*, vol. 13, no. 1, p. 16 197, 2023.

[31] J. Cui *et al.*, "Modeling relaxed policies for discontinuation of methicillin-resistant staphylococcus aureus contact precautions," *Infection Control & Hospital Epidemiology*, pp. 1–6, 2024.

[32] J. Cui, J. Heavey, E. Klein, G. Madden, A. Vullikanti, and B. A. Prakash, "Estimating hais importation probability with ehrs and agent-based model," in *preparation for PNAS*.

[33] J. Cui, A. Haddadan, A. A.-U. Haque, B. Adhikari, A. Vullikanti, and B. A. Prakash, "Accurately estimating total covid-19 infections using information theory," *medRxiv*, pp. 2021–09, 2021.

[34] V. Anand*, J. Cui*, J. Heavey, A. Vullikanti, and B. A. Prakash, "H2ABM: Heterogeneous agent-based model on hypergraphs to capture group interactions," in *SDM*, 2024.

[35] L. Kong, J. Cui, Y. Zhuang, R. Feng, B. A. Prakash, and C. Zhang, "End-to-end stochastic optimization with energy-based model," *Advances in Neural Information Processing Systems*, vol. 35, pp. 11 341–11 354, 2022.

[36] P. W. Stone, "Economic burden of healthcare-associated infections: An american perspective," *Expert review of pharmacoeconomics & outcomes research*, vol. 9, no. 5, pp. 417–422, 2009.

[37] A. Sharma, J. Leal, J. Kim, C. Pearce, D. R. Pillai, and A. Hollis, "The cost of contact precautions: A systematic analysis.," *Canadian Journal of Infection Control*, vol. 35, no. 4, 2020.

[38] J. A. Roth *et al.*, "Direct costs of a contact isolation day: A prospective cost analysis at a swiss university hospital," *infection control & hospital epidemiology*, vol. 39, no. 1, pp. 101–103, 2018.

[39] B. P. Howden *et al.*, "Staphylococcus aureus host interactions and adaptation," *Nature Reviews Microbiology*, pp. 1–16, 2023.

[40] F. D. Lowy, "Staphylococcus aureus infections," *New England journal of medicine*, vol. 339, no. 8, pp. 520–532, 1998.

[41] C. Suetens *et al.*, "Prevalence of healthcare-associated infections, estimated incidence and composite antimicrobial resistance index in acute care hospitals and long-term care facilities: Results from two european point prevalence surveys, 2016 to 2017," *Eurosurveillance*, vol. 23, no. 46, p. 1 800 516, 2018.

[42] D. L. Smith, J. Dushoff, E. N. Perencevich, A. D. Harris, and S. A. Levin, "Persistent colonization and the spread of antibiotic resistance in nosocomial pathogens: Resistance is a regional problem," *Proceedings of the National Academy of Sciences*, vol. 101, no. 10, pp. 3709–3714, 2004.

[43] S. K. Perreault, B. Binks, D. S. McManus, and J. E. Topal, "Evaluation of the negative predictive value of methicillin resistant staphylococcus aureus nasal swab screening in patients with acute myeloid leukemia," *Infection Control & Hospital Epidemiology*, vol. 42, no. 7, pp. 853–856, 2021.

[44] J. Luteijn, G. Hubben, P. Pechlivanoglou, M. Bonten, and M. Postma, "Diagnostic accuracy of culture-based and pcr-based detection tests for methicillin-resistant staphylococcus aureus: A meta-analysis," *Clinical microbiology and infection*, vol. 17, no. 2, pp. 146–154, 2011.

[45] B. Dangerfield, A. Chung, B. Webb, and M. T. Seville, "Predictive value of methicillin resistant staphylococcus aureus (mrsa) nasal swab pcr assay for mrsa pneumonia," *Antimicrobial agents and chemotherapy*, vol. 58, no. 2, pp. 859–864, 2014.

[46] D. Pittet *et al.*, "Evidence-based model for hand transmission during patient care and the role of improved practices," *The Lancet infectious diseases*, vol. 6, no. 10, pp. 641–652, 2006.

[47] A. Gawande, "On washing hands," *New England Journal of Medicine*, vol. 350, no. 13, pp. 1283–1286, 2004.

[48] J. D. Siegel, E. Rhinehart, M. Jackson, and L. Chiarello, "2007 guideline for isolation precautions: Preventing transmission of infectious agents in health care settings," *American journal of infection control*, vol. 35, no. 10, S65–S164, 2007.

[49] V. R. Roth *et al.*, "Cost analysis of universal screening vs. risk factor-based screening for methicillin-resistant staphylococcus aureus (mrsa)," *PloS one*, vol. 11, no. 7, e0159667, 2016.

[50] L. D. Croft *et al.*, "The effect of contact precautions on frequency of hospital adverse events," *infection control & hospital epidemiology*, vol. 36, no. 11, pp. 1268–1274, 2015.

[51] J. D. Siegel, E. Rhinehart, M. Jackson, and L. Chiarello, "Management of multidrug-resistant organisms in health care settings, 2006," *American journal of infection control*, vol. 35, no. 10, S165–S193, 2007.

[52] Y. Hirano *et al.*, "Machine learning approach to predict positive screening of methicillin resistant staphylococcus aureus during mechanical ventilation using synthetic dataset from mimic-iv database," *Frontiers in medicine*, p. 2222, 2021.

[53] T. Hartvigsen, C. Sen, S. Brownell, E. Teeple, X. Kong, and E. A. Rundensteiner, "Early prediction of mrsa infections using electronic health records.," in *HEALTHINF*, 2018, pp. 156–167.

[54] B. Wilder *et al.*, "Modeling between-population variation in covid-19 dynamics in hubei, lombardy, and new york city," *Proceedings of the National Academy of Sciences*, vol. 117, no. 41, pp. 25 904–25 910, 2020.

[55] S. L. Wu *et al.*, "Substantial underestimation of sars-cov-2 infection in the united states," *Nature communications*, vol. 11, no. 1, pp. 1–10, 2020.

[56] Q. Cao and B. Heydari, "Micro-level social structures and the success of covid-19 national policies," *Nature Computational Science*, vol. 2, no. 9, pp. 595–604, 2022.

[57] P. Padmanabhan, R. Desikan, and N. M. Dixit, "Modeling how antibody responses may determine the efficacy of covid-19 vaccines," *Nature Computational Science*, vol. 2, no. 2, pp. 123–131, 2022.

[58] T. W. Russell *et al.*, "Reconstructing the early global dynamics of under-ascertained covid-19 cases and infections," *BMC medicine*, vol. 18, no. 1, pp. 1–9, 2020.

[59] A. N. Angelopoulos, R. Pathak, R. Varma, and M. I. Jordan, "On identifying and mitigating bias in the estimation of the covid-19 case fatality rate," *arXiv preprint arXiv:2003.08592*, 2020.

[60] W. H. Press and R. C. Levin, "Modeling, post covid-19," *Science*, vol. 370, no. 6520, pp. 1015–1015, 2020.

[61] M. U. Kraemer *et al.*, "The effect of human mobility and control measures on the covid-19 epidemic in china," *Science*, vol. 368, no. 6490, pp. 493–497, 2020.

[62] S. Lai *et al.*, "Effect of non-pharmaceutical interventions to contain covid-19 in china," *Nature*, vol. 585, no. 7825, pp. 410–413, 2020.

[63] C. R. Wells *et al.*, "Impact of international travel and border control measures on the global spread of the novel 2019 coronavirus outbreak," *Proceedings of the National Academy of Sciences*, vol. 117, no. 13, pp. 7504–7509, 2020.

[64] X. Hao, S. Cheng, D. Wu, T. Wu, X. Lin, and C. Wang, "Reconstruction of the full transmission dynamics of covid-19 in wuhan," *Nature*, vol. 584, no. 7821, pp. 420–424, 2020.

[65] M. P. Kain, M. L. Childs, A. D. Becker, and E. A. Mordecai, "Chopping the tail: How preventing superspreading can help to maintain covid-19 control," *Epidemics*, vol. 34, p. 100 430, 2021.

[66] V. Gopalakrishnan *et al.*, "Globally local: Hyper-local modeling for accurate forecast of covid-19," *Epidemics*, vol. 37, p. 100 510, 2021.

[67] B. Adhikari, B. Lewis, A. Vullikanti, J. M. Jiménez, and B. A. Prakash, "Fast and near-optimal monitoring for healthcare acquired infection outbreaks," *PLoS computational biology*, vol. 15, no. 9, e1007284, 2019.

[68] H. Jang, S. Justice, P. M. Polgreen, A. M. Segre, D. K. Sewell, and S. V. Pemmaraju, "Evaluating architectural changes to alter pathogen dynamics in a dialysis unit: For the cdc mind-healthcare group," in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2019, pp. 961–968.

[69] L. Kong, J. Cui, H. Sun, Y. Zhuang, B. A. Prakash, and C. Zhang, "Autoregressive diffusion model for graph generation," in *International Conference on Learning Representations*, 2023.

[70] J. Leskovec, L. A. Adamic, and B. A. Huberman, "The dynamics of viral marketing," *ACM Transactions on the Web (TWEB)*, vol. 1, no. 1, 5–es, 2007.

[71] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proceedings of the 13th international conference on World Wide Web*, 2004, pp. 491–501.

[72] J. O. Kephart and S. R. White, "Measuring and modeling computer virus prevalence," in *Proceedings 1993 IEEE Computer Society Symposium on Research in Security and Privacy*, IEEE, 1993, pp. 2–15.

[73] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 137–146.

[74] C. C. Aggarwal, S. Lin, and P. S. Yu, "On influential node discovery in dynamic social networks," in *Proceedings of the 2012 SIAM International Conference on Data Mining*, SIAM, 2012, pp. 636–647.

[75] N. T. Gayraud, E. Pitoura, and P. Tsaparas, "Diffusion maximization in evolving social networks," in *Proceedings of the 2015 acm on conference on online social networks*, 2015, pp. 125–135.

[76] M. E. Newman, "Spread of epidemic disease on networks," *Physical review E*, vol. 66, no. 1, p. 016 128, 2002.

[77] M. E. Newman, "Threshold effects for two pathogens spreading on a network," *Physical review letters*, vol. 95, no. 10, p. 108 701, 2005.

[78] A. Ganesh, L. Massoulie, and D. Towsley, "The effect of network topology on the spread of epidemics," in *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, IEEE, vol. 2, 2005, pp. 1455–1466.

[79] D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos, "Epidemic thresholds in real networks," *ACM Transactions on Information and System Security (TISSEC)*, vol. 10, no. 4, pp. 1–26, 2008.

[80] U. Jeong, K. Ding, L. Cheng, R. Guo, K. Shu, and H. Liu, "Nothing stands alone: Relational fake news detection with hypergraph neural networks," in *2022 IEEE International Conference on Big Data (Big Data)*, IEEE, 2022, pp. 596–605.

[81] K. Bouafia and B. Molnár, "Hypergraph application on business process performance," *Information*, vol. 12, no. 9, p. 370, 2021.

[82] J. Wang, K. Ding, Z. Zhu, and J. Caverlee, "Session-based recommendation with hypergraph attention networks," in *SDM*, SIAM, 2021, pp. 82–90.

[83] J. Cui, H. Wu, L. Fu, and X. Gan, "De-anonymizing bitcoin networks: An ip matching method via heuristic approach: Poster," in *Proceedings of the ACM Turing Celebration Conference-China*, 2019, pp. 1–2.

[84] D. Hong, R. Dey, X. Lin, B. Cleary, and E. Dobriban, "Group testing via hypergraph factorization applied to covid-19," *Nature communications*, vol. 13, no. 1, p. 1837, 2022.

[85] G. Raschpichler *et al.*, "Development and external validation of a clinical prediction model for mrsa carriage at hospital admission in southeast lower saxony, germany," *Scientific reports*, vol. 10, no. 1, p. 17 998, 2020.

[86] H. Jang, S. Pai, B. Adhikari, and S. V. Pemmaraju, "Risk-aware temporal cascade reconstruction to detect asymptomatic cases: For the cdc mind healthcare network," in *2021 IEEE International Conference on Data Mining (ICDM)*, IEEE, 2021, pp. 240–249.

[87] A. Rodriguez *et al.*, "Data-centric epidemic forecasting: A survey," *arXiv preprint arXiv:2207.09370*, 2022.

[88] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 58, no. 1, pp. 267–288, 1996.

[89] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[90] R. P. Soebiyanto, F. Adimi, and R. K. Kiang, "Modeling and predicting seasonal influenza transmission in warm regions using climatological parameters," *PloS one*, vol. 5, no. 3, e9450, 2010.

[91] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[92] S. Ning, S. Yang, and S. Kou, "Accurate regional influenza epidemics tracking using internet search data," *Scientific reports*, vol. 9, no. 1, p. 5238, 2019.

[93] S. Yang, S. Ning, and S. Kou, "Use internet search data to accurately track state level influenza epidemics," *Scientific reports*, vol. 11, no. 1, p. 4023, 2021.

[94] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[95] K. Cho *et al.*, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[96] A. Vaswani *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[97] B. Adhikari, X. Xu, N. Ramakrishnan, and B. A. Prakash, "Epideep: Exploiting embeddings for epidemic forecasting," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 577–586.

[98] A. Rodriguez, N. Muralidhar, B. Adhikari, A. Tabassum, N. Ramakrishnan, and B. A. Prakash, "Steering a historical disease forecasting model under a pandemic: Case of flu and covid-19," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 4855–4863.

[99] M. R. Ibrahim, J. Haworth, A. Lipani, N. Aslam, T. Cheng, and N. Christie, "Variational lstm autoencoder to forecast the spread of coronavirus across the globe," *PloS one*, vol. 16, no. 1, e0246120, 2021.

[100] S. Deng, S. Wang, H. Rangwala, L. Wang, and Y. Ning, "Cola-gnn: Cross-location attention based graph neural networks for long-term ili prediction," in *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020, pp. 245–254.

[101] P. Butler, N. Ramakrishnan, E. O. Nsoesie, and J. S. Brownstein, "Satellite imagery analysis: What can hospital parking lots tell us about a disease outbreak?" *Computer*, vol. 47, no. 04, pp. 94–97, 2014.

[102] A. Lamb, M. Paul, and M. Dredze, "Separating fact from fear: Tracking flu infections on twitter," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 789–795.

[103] W. G. Van Panhuis *et al.*, "Risk of dengue for tourists and teams during the world cup 2014 in brazil," *PLoS Neglected Tropical Diseases*, vol. 8, no. 7, e3063, 2014.

[104] C. Viboud, P.-Y. Boelle, F. Carrat, A.-J. Valleron, and A. Flahault, "Prediction of the spread of influenza epidemics by the method of analogues," *American Journal of Epidemiology*, vol. 158, no. 10, pp. 996–1006, 2003.

[105] H. Kamarthi, L. Kong, A. Rodriguez, C. Zhang, and B. A. Prakash, "Camul: Calibrated and accurate multi-view time-series forecasting," in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 3174–3185.

[106] B. Wilder, B. Dilkina, and M. Tambe, "Melding the data-decisions pipeline: Decision focused learning for combinatorial optimization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 1658–1665.

[107] J. Mandi and T. Guns, "Interior point solving for lp-based prediction+ optimisation," *Advances in Neural Information Processing Systems*, vol. 33, pp. 7272–7282, 2020.

[108] M. Mulamba, J. Mandi, M. Diligenti, M. Lombardi, V. Bucarey, and T. Guns, "Contrastive losses and solution caching for predict-and-optimize," *arXiv preprint arXiv:2011.05354*, 2020.

[109] S. O. Arik *et al.*, "Interpretable Sequence Learning for COVID-19 Forecasting," *NeurIPS*, 2020.

[110] L. Wang *et al.*, "DEFSI: Deep Learning Based Epidemic Forecasting with Synthetic Information," *AAAI*, 2019.

[111] J. Gao *et al.*, "STAN: Spatio-temporal attention network for pandemic prediction using real-world evidence," *JAMA*, 2021.

[112] B. Wang *et al.*, "Data assimilation and its applications," *PNAS*, 2000.

[113] A. Yazdani, L. Lu, M. Raissi, and G. E. Karniadakis, "Systems biology informed deep learning for inferring parameters and hidden dynamics," *PLoS computational biology*, vol. 16, no. 11, e1007575, 2020.

[114] A. Rodriguez, J. Cui, N. Ramakrishnan, B. Adhikari, and B. A. Prakash, "Einns: Epidemiologically-informed neural networks," in *AAAI*, vol. 37, 2023, pp. 14 453–14 460.

[115] S. Zheng, A. Trott, S. Srinivasa, D. C. Parkes, and R. Socher, "The ai economist: Taxation policy design via two-level deep multiagent reinforcement learning," *Science advances*, vol. 8, no. 18, eabk2607, 2022.

[116] E. Sert, Y. Bar-Yam, and A. J. Morales, "Segregation dynamics with reinforcement learning and agent based modeling," *Scientific reports*, vol. 10, no. 1, p. 11 771, 2020.

[117] D. Radovic, L. Kruitwagen, C. S. de Witt, B. Caldecott, S. Tomlinson, and M. Workman, "Revealing robust oil and gas company macro-strategies using deep multi-agent reinforcement learning," *arXiv preprint arXiv:2211.11043*, 2022.

[118] Y. Bengio *et al.*, "Predicting infectiousness for proactive contact tracing," *arXiv preprint arXiv:2010.12536*, 2020.

[119] T. Reiker *et al.*, "Machine learning approaches to calibrate individual-based infectious disease models," *medRxiv*, pp. 2021–01, 2021.

[120] A. Chopra *et al.*, "Differentiable agent-based epidemiology," in *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, 2023, pp. 1848–1857.

[121] E. Montella *et al.*, "Healthcare associated infections in the neonatal intensive care unit of the "federico ii" university hospital: Statistical analysis and study of risk factors," in *2021 International Symposium on Biomedical Engineering and Computational Biology*, ser. BECB 2021, Nanchang, China: Association for Computing Machinery, 2022, ISBN: 9781450384117.

[122] J. Heavey, J. Cui, C. Chen, B. A. Prakash, and A. Vullikanti, "Provable sensor sets for epidemic detection over networks with minimum delay," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, 2022, pp. 10 202–10 209.

[123] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-n recommendation with a neural co-attention model," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 1531–1540.

[124] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[125] F. Zhang, M. Wang, J. Xi, J. Yang, and A. Li, "A novel heterogeneous network-based method for drug response prediction in cancer cell lines," *Scientific reports*, vol. 8, no. 1, pp. 1–9, 2018.

[126] M. M. Wolf, A. M. Klinvex, and D. M. Dunlavy, "Advantages to modeling relational data using hypergraphs versus graphs," in *HPEC*, IEEE, 2016, pp. 1–7.

[127] J. L. Anderson, "An ensemble adjustment kalman filter for data assimilation," *Monthly weather review*, vol. 129, no. 12, pp. 2884–2903, 2001.

[128] A. Rodriguez *et al.*, "Deepcovid: An operational deep learning-driven framework for explainable real-time covid-19 forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, 2021, pp. 15 393–15 400.

[129] E. Y. C. et al., "Evaluation of individual and ensemble probabilistic forecasts of covid-19 mortality in the united states," *Proceedings of the National Academy of Sciences*, vol. 119, no. 15, e2113561119, 2022.

[130] E. Y. C. et al., “The united states covid-19 forecast hub dataset,” *Scientific data*, vol. 9, no. 1, p. 462, 2022.

[131] G. Lee, J. Yoo, and K. Shin, “Mining of real-world hypergraphs: Patterns, tools, and generators,” in *CIKM*, 2022, pp. 5144–5147.

[132] R. Coello, J. Glynn, C. Gaspar, J. Picazo, and J. Fereres, “Risk factors for developing clinical infection with methicillin-resistant staphylococcus aureus (mrsa) amongst hospital patients initially only colonized with mrsa,” *Journal of Hospital Infection*, vol. 37, no. 1, pp. 39–46, 1997.

[133] N. A. Christakis and J. H. Fowler, “The collective dynamics of smoking in a large social network,” *New England journal of medicine*, vol. 358, no. 21, pp. 2249–2258, 2008.

[134] P. H. England, *Guidance for stepdown of infection control precautions and discharging covid-190 patients*, 2020.

[135] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos, “Gelling, and melting, large graphs by edge manipulation,” in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012, pp. 245–254.

[136] Y. Zhang and B. A. Prakash, “Dava: Distributing vaccines over networks under prior information,” in *Proceedings of the 2014 SIAM International Conference on Data Mining*, SIAM, 2014, pp. 46–54.

[137] S. Chandrasekaran and S. C. Jiang, “A dose response model for staphylococcus aureus,” *Scientific Reports*, vol. 11, no. 1, pp. 1–10, 2021.

[138] J. Coia *et al.*, “Guidelines for the control and prevention of meticillin-resistant staphylococcus aureus (mrsa) in healthcare facilities,” *Journal of hospital infection*, vol. 63, S1–S44, 2006.

[139] O. of Disease Prevention and H. Promotion, *Health care-associated infections*, https://www.cdc.gov/mrsa/community/environment/index.htm.

[140] P. Shakarian, A. Bhatnagar, A. Aleali, E. Shaabani, and R. Guo, “The independent cascade and linear threshold models,” in *Diffusion in Social Networks*, Springer, 2015, pp. 35–48.

[141] M. Cabrera, F. Córdova-Lepe, J. P. Gutiérrez-Jara, and K. Vogt-Geisse, “An sir-type epidemiological model that integrates social distancing as a dynamic law based on point prevalence and socio-behavioral factors,” *Scientific Reports*, vol. 11, no. 1, p. 10 170, 2021.

[142] B. A. Prakash, D. Chakrabarti, M. Faloutsos, N. Valler, and C. Faloutsos, "Threshold conditions for arbitrary cascade models on arbitrary networks," in *2011 IEEE 11th International Conference on Data Mining*, IEEE Computer Society, 2011, pp. 537–546.

[143] A. Hasan *et al.*, "A new estimation method for covid-19 time-varying reproduction number using active cases," *Scientific Reports*, vol. 12, no. 1, p. 6675, 2022.

[144] Q.-H. Liu, M. Ajelli, A. Aleta, S. Merler, Y. Moreno, and A. Vespignani, "Measurability of the epidemic reproduction number in data-driven contact networks," *Proceedings of the National Academy of Sciences*, vol. 115, no. 50, pp. 12 680–12 685, 2018.

[145] L. Alvarez, M. Colom, J.-D. Morel, and J.-M. Morel, "Computing the daily reproduction number of covid-19 by inverting the renewal equation using a variational technique," *Proceedings of the National Academy of Sciences*, vol. 118, no. 50, e2105112118, 2021.

[146] T. V. Inglesby, "Public health measures and the reproduction number of sars-cov-2," *Jama*, vol. 323, no. 21, pp. 2186–2187, 2020.

[147] H. W. Hethcote, "The mathematics of infectious diseases," *SIAM review*, vol. 42, no. 4, pp. 599–653, 2000.

[148] J. M. Pujol, J. E. Eisenberg, C. N. Haas, and J. S. Koopman, "The effect of ongoing exposure dynamics in dose response relationships," *PLoS computational biology*, vol. 5, no. 6, e1000399, 2009.

[149] K. A. Schmidt and R. S. Ostfeld, "Biodiversity and the dilution effect in disease ecology," *Ecology*, vol. 82, no. 3, pp. 609–619, 2001.

[150] S. M. Bartsch *et al.*, "Modeling interventions to reduce the spread of multidrug-resistant organisms between health care facilities in a region," *JAMA Network Open*, vol. 4, no. 8, e2119212–e2119212, 2021.

[151] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 2012.

[152] J. Alman and V. V. Williams, *A refined laser method and faster matrix multiplication*, 2020. arXiv: 2010.05846.

[153] G. Madden *et al.*, "Deciphering COVID-19-Associated Effects on Hospital MRSA Transmission and Social Networks," *Open Forum Infectious Diseases*, vol. 8, no. S1, S104–S106, Dec. 2021.

[154] A. F. Shorr, D. E. Myers, D. B. Huang, B. H. Nathanson, M. F. Emons, and M. H. Kollef, "A risk score for identifying methicillin-resistant staphylococcus aureus in patients presenting to the hospital with pneumonia," *BMC infectious diseases*, vol. 13, no. 1, pp. 1–7, 2013.

[155] Centers for Disease Control and Prevention, *Multidrug-resistant organisms (mdro) management*, https://www.cdc.gov/infectioncontrol/guidelines/mdro/index.html.

[156] N. E. West, W.-F. Cheong, E. Boone, and N. E. Moat, "Impact of the covid-19 pandemic: A perspective from industry," *European Heart Journal Supplements*, vol. 22, no. Supplement_P, P56–P59, 2020.

[157] C. E. Knezevic, B. Das, J. M. El-Khoury, P. J. Jannetto, F. Lacbawan, and W. E. Winter, "Rising to the challenge: Shortages in laboratory medicine," *Clinical Chemistry*, 2022.

[158] "Nurse staffing and health care-associated infections: Proceedings from a working group meeting," *American Journal of Infection Control*, vol. 30, no. 4, pp. 199–206, 2002.

[159] Centers for Disease Control and Prevention, *Health care-associated infections*, https://www.cdc.gov/mrsa/community/patients.html.

[160] T. S. Naimi *et al.*, "Comparison of community-and health care–associated methicillin-resistant staphylococcus aureus infection," *Jama*, vol. 290, no. 22, pp. 2976–2984, 2003.

[161] P. Vanhems *et al.*, "Estimating potential infection transmission routes in hospital wards using wearable proximity sensors," *PloS one*, vol. 8, no. 9, e73970, 2013.

[162] E. Valdano, C. Poletto, P.-Y. Boëlle, and V. Colizza, "Reorganization of nurse scheduling reduces the risk of healthcare associated infections," *Scientific reports*, vol. 11, no. 1, pp. 1–12, 2021.

[163] Centers for Disease Control and Prevention, *Health care-associated infections*, https://www.cdc.gov/mrsa/community/index.html.

[164] P. J. Carson *et al.*, ""bringing down the flag": Removing the contact precautions label for non-hospitalized patients with prior methicillin-resistant staphylococcus aureus infection or colonization," *infection control & hospital epidemiology*, vol. 36, no. 5, pp. 578–580, 2015.

[165] S. Pei, F. Morone, F. Liljeros, H. Makse, and J. L. Shaman, "Inference and control of the nosocomial transmission of methicillin-resistant staphylococcus aureus," *Elife*, vol. 7, e40977, 2018.

[166] *Oanda. currency converter*, https://www.oanda.com/currency-converter/en/?from=EUR&to=USD&amount=1.

[167] *Statistics usbol. cpi inflaction calculator*, https://data.bls.gov/cgi-bin/cpicalc.pl.

[168] S. Pei, S. Kandula, and J. Shaman, "Differential effects of intervention timing on covid-19 spread in the united states," *Science advances*, vol. 6, no. 49, eabd6370, 2020.

[169] C. E. Practice *et al.*, "Barriers and facilitators to healthcare workers' adherence with infection prevention and control (ipc) guidelines for respiratory infectious diseases: A rapid qualitative evidence synthesis," *Cochrane database of systematic reviews*, vol. 2020, no. 8, 1996.

[170] E. Dong, H. Du, and L. Gardner, "An interactive web-based dashboard to track covid-19 in real time," *The Lancet infectious diseases*, vol. 20, no. 5, pp. 533–534, 2020.

[171] *Interim economic projections for 2020 and 2021*, https://www.cbo.gov/publication/56351.

[172] *4q gdp: Economy expands at a 4.0% annualized rate*, https://finance.yahoo.com/news/4q-gdp-2020-us-economy-coronavirus-pandemic-180133456.html.

[173] T. Bedford *et al.*, "Cryptic transmission of sars-cov-2 in washington state," *Science*, vol. 370, no. 6516, pp. 571–575, 2020.

[174] *Hidden outbreaks spread through u.s. cities far earlier than americans knew, estimates say*, https://www.nytimes.com/2020/04/23/us/coronavirus-early-outbreaks-cities.html.

[175] S. Tiwari, C. Vyasarayani, and A. Chatterjee, "Data suggest covid-19 affected numbers greatly exceeded detected numbers, in four european countries, as per a delayed seiqr model," *Scientific reports*, vol. 11, no. 1, pp. 1–12, 2021.

[176] *Us covid cases likely more than double official count, experts say.* https://www.cidrap.umn.edu/news-perspective/2021/07/us-covid-cases-likely-more-double-official-count-experts-say, 2021.

[177] N. J. Irons and A. E. Raftery, "Estimating sars-cov-2 infections from deaths, confirmed cases, tests, and random surveys," *Proceedings of the National Academy of Sciences*, vol. 118, no. 31, 2021.

[178] S. Pei, T. K. Yamana, S. Kandula, M. Galanti, and J. Shaman, “Burden and characteristics of covid-19 in the united states during 2020,” *Nature*, vol. 598, no. 7880, pp. 338–341, 2021.

[179] R. Subramanian, Q. He, and M. Pascual, “Quantifying asymptomatic infection and transmission of covid-19 in new york city using observed cases, serology, and testing capacity,” *Proceedings of the National Academy of Sciences*, vol. 118, no. 9, 2021.

[180] S. M. Moghadas *et al.*, “The implications of silent transmission for the control of covid-19 outbreaks,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 30, pp. 17 513–17 515, 2020.

[181] N. Sood *et al.*, “Seroprevalence of sars-cov-2–specific antibodies among adults in los angeles county, california, on april 10-11, 2020,” *JAMA*, vol. 323, no. 23, pp. 2425–2427, 2020.

[182] F. P. Havers *et al.*, “Seroprevalence of antibodies to sars-cov-2 in 10 sites in the united states, march 23-may 12, 2020,” *JAMA internal medicine*, vol. 180, no. 12, pp. 1576–1586, 2020.

[183] W. Zhang *et al.*, “Analysis of genomic characteristics and transmission routes of patients with confirmed sars-cov-2 in southern california during the early stage of the us covid-19 pandemic,” *JAMA network open*, vol. 3, no. 10, e2024191, 2020.

[184] *Commercial laboratory seroprevalence survey data*, https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/commercial-lab-surveys.html.

[185] *Covid-19 test price information*, https://www.questdiagnostics.com/business-solutions/health-plans/covid-19/pricing.

[186] E. K. Accorsi *et al.*, “How to detect and reduce potential sources of biases in studies of sars-cov-2 and covid-19,” *European Journal of Epidemiology*, pp. 1–18, 2021.

[187] F. S. Lu *et al.*, “Estimating the cumulative incidence of covid-19 in the united states using influenza surveillance, virologic testing, and mortality data: Four complementary approaches,” *PLOS Computational Biology*, vol. 17, no. 6, e1008994, 2021.

[188] D. Koutra, U. Kang, J. Vreeken, and C. Faloutsos, “Vog: Summarizing and understanding large graphs,” in *Proceedings of the 2014 SIAM international conference on data mining*, SIAM, 2014, pp. 91–99.

[189] K. Budhathoki and J. Vreeken, “Origo: Causal inference by compression,” *Knowledge and Information Systems*, vol. 56, no. 2, pp. 285–307, 2018.

[190] B. Adhikari, P. Rangudu, B. A. Prakash, and A. Vullikanti, "Near-optimal mapping of network states using probes," in *Proceedings of the 2018 SIAM International Conference on Data Mining*, SIAM, 2018, pp. 108–116.

[191] B. A. Prakash, J. Vreeken, and C. Faloutsos, "Spotting culprits in epidemics: How many and which ones?" In *2012 IEEE 12th International Conference on Data Mining*, IEEE, 2012, pp. 11–20.

[192] P. D. Grünwald, *The minimum description length principle*. MIT press, 2007.

[193] E. L. Ionides, D. Nguyen, Y. Atchadé, S. Stoev, and A. A. King, "Inference for dynamic and latent variable models via iterated, perturbed bayes maps," *Proceedings of the National Academy of Sciences*, vol. 112, no. 3, pp. 719–724, 2015.

[194] F. Gao and L. Han, "Implementing the nelder-mead simplex algorithm with adaptive parameters," *Computational Optimization and Applications*, vol. 51, no. 1, pp. 259–277, 2012.

[195] A. Reinhart *et al.*, "An open repository of real-time covid-19 indicators," *Proceedings of the National Academy of Sciences*, vol. 118, no. 51, 2021.

[196] J. A. Salomon *et al.*, "The us covid-19 trends and impact survey: Continuous real-time measurement of covid-19 symptoms, risks, protective behaviors, testing, and vaccination," *Proceedings of the National Academy of Sciences*, vol. 118, no. 51, 2021.

[197] *United states covid-19 cases, deaths, and laboratory testing (naats) by state, territory, and jurisdiction*, https://covid.cdc.gov/covid-data-tracker/cases.

[198] M. U. Kraemer *et al.*, "Data curation during a pandemic and lessons learned from covid-19," *Nature Computational Science*, vol. 1, no. 1, pp. 9–10, 2021.

[199] T. Kustin *et al.*, "Evidence for increased breakthrough rates of sars-cov-2 variants of concern in bnt162b2-mrna-vaccinated individuals," *Nature medicine*, vol. 27, no. 8, pp. 1379–1384, 2021.

[200] D. Planas *et al.*, "Reduced sensitivity of sars-cov-2 variant delta to antibody neutralization," *Nature*, vol. 596, no. 7871, pp. 276–280, 2021.

[201] D. Geers *et al.*, "Sars-cov-2 variants of concern partially escape humoral but not t-cell responses in covid-19 convalescent donors and vaccinees," *Science Immunology*, vol. 6, no. 59, 2021.

[202] A. Nande *et al.*, "The effect of eviction moratoria on the transmission of sars-cov-2," *Nature communications*, vol. 12, no. 1, pp. 1–13, 2021.

[203] N. Hoertel *et al.*, "A stochastic agent-based model of the sars-cov-2 epidemic in france," *Nature medicine*, vol. 26, no. 9, pp. 1417–1421, 2020.

[204] Y. Tian, A. Sridhar, O. Yağan, and H. V. Poor, "Analysis of the impact of mask-wearing in viral spread: Implications for covid-19," in *2021 American Control Conference (ACC)*, IEEE, 2021, pp. 3132–3137.

[205] S. Chang *et al.*, "Mobility network models of covid-19 explain inequities and inform reopening," *Nature*, vol. 589, no. 7840, pp. 82–87, 2021.

[206] S. Pei, X. Teng, P. Lewis, and J. Shaman, "Optimizing respiratory virus surveillance networks using uncertainty propagation," *Nature communications*, vol. 12, no. 1, pp. 1–10, 2021.

[207] Q. Bi *et al.*, "Epidemiology and transmission of covid-19 in 391 cases and 1286 of their close contacts in shenzhen, china: A retrospective cohort study," *The Lancet infectious diseases*, vol. 20, no. 8, pp. 911–919, 2020.

[208] J. Jay *et al.*, "Neighbourhood income and physical distancing during the covid-19 pandemic in the united states," *Nature human behaviour*, vol. 4, no. 12, pp. 1294–1302, 2020.

[209] J. R. Goldstein and R. D. Lee, "Demographic perspectives on the mortality of covid-19 and other epidemics," *Proceedings of the National Academy of Sciences*, vol. 117, no. 36, pp. 22 035–22 041, 2020.

[210] F. K. Ho *et al.*, "Is older age associated with covid-19 mortality in the absence of other risk factors? general population cohort study of 470,034 participants," *PloS one*, vol. 15, no. 11, e0241824, 2020.

[211] W. E. Hart, J.-P. Watson, and D. L. Woodruff, "Pyomo: Modeling and solving mathematical programs in python," *Mathematical Programming Computation*, vol. 3, no. 3, pp. 219–260, 2011.

[212] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, "A rewriting system for convex optimization problems," *Journal of Control and Decision*, vol. 5, no. 1, pp. 42–60, 2018.

[213] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[214] P. Donti, B. Amos, and J. Z. Kolter, "Task-based end-to-end model learning in stochastic optimization," *Advances in neural information processing systems*, vol. 30, 2017.

[215] A. Agrawal, B. Amos, S. Barratt, S. Boyd, S. Diamond, and J. Z. Kolter, "Differentiable convex optimization layers," *Advances in neural information processing systems*, vol. 32, 2019.

[216] B. Amos and J. Z. Kolter, "Optnet: Differentiable optimization as a layer in neural networks," in *International Conference on Machine Learning*, PMLR, 2017, pp. 136–145.

[217] O. Vinyals, M. Fortunato, and N. Jaitly, "Pointer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[218] E. Khalil, H. Dai, Y. Zhang, B. Dilkina, and L. Song, "Learning combinatorial optimization algorithms over graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[219] K. Li and J. Malik, "Learning to optimize," *arXiv preprint arXiv:1606.01885*, 2016.

[220] A. Perrault, B. Wilder, E. Ewing, A. Mate, B. Dilkina, and M. Tambe, "End-to-end game-focused learning of adversary behavior in security games," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 1378–1386.

[221] K. Wang, A. Perrault, A. Mate, and M. Tambe, "Scalable game-focused learning of adversary models: Data-to-decisions in network security games.," in *AAMAS*, 2020, pp. 1449–1457.

[222] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "A tutorial on energy-based learning," *Predicting structured data*, vol. 1, no. 0, 2006.

[223] S. Kim, R. Pasupathy, and S. G. Henderson, "A guide to sample average approximation," *Handbook of simulation optimization*, pp. 207–243, 2015.

[224] B. Verweij, S. Ahmed, A. J. Kleywegt, G. Nemhauser, and A. Shapiro, "The sample average approximation method applied to stochastic routing problems: A computational study," *Computational optimization and applications*, vol. 24, no. 2, pp. 289–333, 2003.

[225] A. J. Kleywegt, A. Shapiro, and T. Homem-de-Mello, "The sample average approximation method for stochastic discrete optimization," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2002.

[226] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*, PMLR, 2016, pp. 1050–1059.

[227] L. Kong, J. Sun, and C. Zhang, "Sde-net: Equipping deep neural networks with uncertainty estimates," in *International Conference on Machine Learning*, PMLR, 2020, pp. 5405–5415.

[228] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, *et al.*, "Maximum entropy inverse reinforcement learning.," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Chicago, IL, USA, vol. 8, 2008, pp. 1433–1438.

[229] M. Wulfmeier, P. Ondruska, and I. Posner, "Maximum entropy deep inverse reinforcement learning," *arXiv preprint arXiv:1507.04888*, 2015.

[230] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

[231] Y. Du and I. Mordatch, "Implicit generation and modeling with energy based models," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[232] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[233] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.

[234] C. Maddison, A. Mnih, and Y. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *Proceedings of the international conference on learning Representations*, International Conference on Learning Representations, 2017.

[235] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *International conference on machine learning*, PMLR, 2014, pp. 1278–1286.

[236] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Representation Learning*, 2015.

[237] M. L. Bynum *et al.*, *Pyomo–optimization modeling in python*, Third. Springer Science & Business Media, 2021, vol. 67.

[238] S. M. Moghadas *et al.*, "Projecting hospital utilization during the covid-19 outbreaks in the united states," *Proceedings of the National Academy of Sciences*, vol. 117, no. 16, pp. 9122–9126, 2020.

[239] I. C.-1. forecasting team, "Modeling covid-19 scenarios for the united states," *Nature medicine*, 2020.

[240] H. Kamarthi, L. Kong, A. Rodriguez, C. Zhang, and B. A. Prakash, "When in doubt: Neural non-parametric uncertainty quantification for epidemic forecasting," *Advances in Neural Information Processing Systems*, vol. 34, 2021.

[241] N. Altieri *et al.*, "Curating a covid-19 data repository and forecasting county-level death counts in the united states," *arXiv preprint arXiv:2005.07882*, 2020.

[242] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[243] K. Wang, B. Wilder, A. Perrault, and M. Tambe, "Automatically learning compact quality-aware surrogates for optimization problems," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9586–9596, 2020.

[244] A. Washburn and K. Wood, "Two-person zero-sum games for network interdiction," *Operations research*, vol. 43, no. 2, pp. 243–251, 1995.

[245] M. Fischetti, I. Ljubic, M. Monaci, and M. Sinnl, "Interdiction games and monotonicity, with application to knapsack problems," *INFORMS Journal on Computing*, vol. 31, no. 2, pp. 390–410, 2019.

[246] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," in *2010 43rd Hawaii International Conference on System Sciences*, IEEE, 2010, pp. 1–10.

[247] L. Csanky, "Fast parallel matrix inversion algorithms," in *16th Annual Symposium on Foundations of Computer Science (sfcs 1975)*, IEEE, 1975, pp. 11–12.

[248] A. Krishnamoorthy and D. Menon, "Matrix inversion using cholesky decomposition," in *2013 signal processing: Algorithms, architectures, arrangements, and applications (SPA)*, IEEE, 2013, pp. 70–72.

[249] Y. Song and D. P. Kingma, "How to train your energy-based models," *arXiv preprint arXiv:2101.03288*, 2021.

[250] Y. Du, S. Li, J. Tenenbaum, and I. Mordatch, "Improved contrastive divergence training of energy-based models," in *International Conference on Machine Learning*, PMLR, 2021, pp. 2837–2848.

[251] E. Nijkamp *et al.*, "Mcmc should mix: Learning energy-based model with neural transport latent space mcmc," in *International Conference on Learning Representations*, 2021.

[252] Centers for Disease Control and Prevention, *Antibiotic Resistance Threats in the United States*, 2019.

[253] M. S. Mietchen, C. T. Short, M. Samore, E. T. Lofgren, and C. M. I. D. in Healthcare Program (MInD-Healthcare), “Examining the impact of icu population interaction structure on modeled colonization dynamics of staphylococcus aureus,” *PLoS Computational Biology*, vol. 18, no. 7, e1010352, 2022.

[254] E. van Kleef, J. V. Robotham, M. Jit, S. R. Deeny, and W. J. Edmunds, “Modelling the transmission of healthcare associated infections: A systematic review,” *BMC infectious diseases*, vol. 13, pp. 1–13, 2013.

[255] F. Haghpanah, G. Lin, E. Klein, and C. M.-H. Program, “Deconstructing the effects of stochasticity on transmission of hospital-acquired infections in icus,” *Royal Society Open Science*, vol. 10, no. 9, p. 230 277, 2023.

[256] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer, 2009, vol. 2.

[257] C. Bishop, “Pattern recognition and machine learning,” *Springer google schola*, vol. 2, pp. 5–43, 2006.

[258] D. A. Martinez *et al.*, “Early prediction of acute kidney injury in the emergency department with machine-learning methods applied to electronic health record data,” *Annals of emergency medicine*, vol. 76, no. 4, pp. 501–514, 2020.

[259] M. Kamruzzaman *et al.*, “Improving risk prediction of methicillin-resistant staphylococcus aureus using machine learning methods with network features: Retrospective development study,” *JMIR AI*, vol. 3, no. 1, e48067, 2024.

[260] S. Pei *et al.*, “Challenges in forecasting antimicrobial resistance,” *Emerging infectious diseases*, vol. 29, no. 4, p. 679, 2023.

[261] M. Reveil and Y.-H. Chen, “Predicting and preventing covid-19 outbreaks in indoor environments: An agent-based modeling study,” *Scientific Reports*, vol. 12, no. 1, p. 16 076, 2022.

[262] X. Xiao, T. B. Mudiyanselage, C. Ji, J. Hu, and Y. Pan, “Fast deep learning training through intelligently freezing layers,” in *2019 international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData)*, IEEE, 2019, pp. 1225–1232.

[263] R. Dutta, M. Schoengens, J.-P. Onnela, and A. Mira, "Abcpy: A user-friendly, extensible, and parallel library for approximate bayesian computation," in *Proceedings of the platform for advanced scientific computing conference*, 2017, pp. 1–9.

[264] T. C. Lee, "An introduction to coding theory and the two-part minimum description length principle," *International statistical review*, vol. 69, no. 2, pp. 169–183, 2001.

VITA

Jiaming Cui was born in Jinzhou, China. He completed his B.S. in Information Engineering and B.S. in Finance from Shanghai Jiao Tong University in 2019. His research aims to bridge public health with clinical decisions using epidemiological modeling and machine learning (ML). He has closely collaborated with clinicians, published in leading science journals and top CS venues such as PNAS, NeurIPS, ICML, AAAI, and has organized workshops at leading conferences like KDD. His work has also significantly contributed to pandemic prediction and prevention in the past several years, including winning best poster award in SDM 2024, helping decision-making in healthcare facilities, and being part of the CDC's Modeling Infectious Diseases in Healthcare Network (MInD–Healthcare) group.