2011

Nosocomial infection modeling and simulation using fine-grained healthcare data

Christopher Scott Hlady
University of Iowa

NOSOCOMIAL INFECTION MODELING AND SIMULATION USING FINE-GRAINED HEALTHCARE DATA

by

Christopher Scott Hlady

An Abstract

Of a thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Computer Science
in the Graduate College of
The University of Iowa

July 2011

Thesis Supervisor: Professor Alberto Segre

ABSTRACT

Simulation has long been used in healthcare settings to study a range of problems, such as determining ideal staffing levels, allocating patient beds, and assisting with medical decision making. Some of this work naturally focuses on the spread of infection within hospitals, where the importance of hospitals as loci and amplifiers of infection was demonstrated during the 2002-2003 SARS outbreak. Increasingly, fine-grained healthcare data is being collected (e.g., patient care data stored in electronic medical record systems, and healthcare worker data from sources including nurse locator badges), presenting an opportunity to develop models that can drive more realistic simulations. We seek to build a realistic hospital simulator that can be used to answer a wide variety of questions about infection prevention, the allocation and placement of expensive resources, and issues surrounding patient care.

Our simulation framework requires three primary inputs: architectural, healthcare worker, and patient data. We used data from the University of Iowa Hospitals and Clinics to build our virtual hospital. We manually constructed a weighted, directed, 19,000 node graph-theoretic representation of the facility based on printed architectural drawings. Using timestamped location information from electronic medical record system logins and algorithms inspired by prior work on location-aware search, each healthcare worker is modeled by one or more "centers" of activity. Centers are determined using a maximum likelihood approach to fit a location and appropriate decay parameters that best describe the observed data. Finally, we developed

compartmental patient models of varying granularity, with each compartment representing some subset of patient care areas within the hospital. Transition probabilities and patient length of stay were fit using three years of patient data.

In designing our simulator, we were able to minimize assumptions about how healthcare workers and patients move, avoiding the "random mixing" assumption common to many infectious disease simulators. We translated techniques from location-aware search into the hospital environment, developed data structures for use in efficiently processing millions of location data points in tens of thousands of rooms for thousands of healthcare workers, improved the performance of the algorithm for identifying optimal single-center healthcare worker models, and introduced heuristics for training multi-center models. We validated our models by comparing the properties of simulated data to known quantities, and testing against expert expectations. To the best of our knowledge, this is the first agent-level hospital-wide simulator based on fine-grained location and interaction data for healthcare workers and patients.

Abstract Approved: _____
Thesis Supervisor

_____
Title and Department

_____
Date

NOSOCOMIAL INFECTION MODELING AND SIMULATION USING FINE-GRAINED HEALTHCARE DATA

by

Christopher Scott Hlady

A thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy
degree in Computer Science
in the Graduate College of
The University of Iowa

July 2011

Thesis Supervisor: Professor Alberto Segre

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Christopher Scott Hlady

has been approved by the Examining Committee for the thesis requirement for the Doctor of Philosophy degree in Computer Science at the July 2011 graduation.

Thesis Committee: _____
Alberto Segre, Thesis Supervisor

_____
Sriram Pemmaraju

_____
Philip Polgreen

_____
Ted Herman

_____
Kasturi Varadarajan

ABSTRACT

Simulation has long been used in healthcare settings to study a range of problems, such as determining ideal staffing levels, allocating patient beds, and assisting with medical decision making. Some of this work naturally focuses on the spread of infection within hospitals, where the importance of hospitals as loci and amplifiers of infection was demonstrated during the 2002-2003 SARS outbreak. Increasingly, fine-grained healthcare data is being collected (e.g., patient care data stored in electronic medical record systems, and healthcare worker data from sources including nurse locator badges), presenting an opportunity to develop models that can drive more realistic simulations. We seek to build a realistic hospital simulator that can be used to answer a wide variety of questions about infection prevention, the allocation and placement of expensive resources, and issues surrounding patient care.

Our simulation framework requires three primary inputs: architectural, healthcare worker, and patient data. We used data from the University of Iowa Hospitals and Clinics to build our virtual hospital. We manually constructed a weighted, directed, 19,000 node graph-theoretic representation of the facility based on printed architectural drawings. Using timestamped location information from electronic medical record system logins and algorithms inspired by prior work on location-aware search, each healthcare worker is modeled by one or more "centers" of activity. Centers are determined using a maximum likelihood approach to fit a location and appropriate decay parameters that best describe the observed data. Finally, we developed

compartmental patient models of varying granularity, with each compartment representing some subset of patient care areas within the hospital. Transition probabilities and patient length of stay were fit using three years of patient data.

In designing our simulator, we were able to minimize assumptions about how healthcare workers and patients move, avoiding the "random mixing" assumption common to many infectious disease simulators. We translated techniques from location-aware search into the hospital environment, developed data structures for use in efficiently processing millions of location data points in tens of thousands of rooms for thousands of healthcare workers, improved the performance of the algorithm for identifying optimal single-center healthcare worker models, and introduced heuristics for training multi-center models. We validated our models by comparing the properties of simulated data to known quantities, and testing against expert expectations. To the best of our knowledge, this is the first agent-level hospital-wide simulator based on fine-grained location and interaction data for healthcare workers and patients.

TABLE OF CONTENTS

LIST OF TABLES

Table

LIST OF FIGURES

Figure

LIST OF ALGORITHMS

Algorithm

CHAPTER 1
INTRODUCTION

Computer simulation of real-world processes is a technique used to answer a variety of questions in numerous disciplines. Simulation is used in meteorology to predict daily precipitation [100], it is used in logistics to predict inventory levels [109], and it is used in education to prepare medical students for surgery [63]. Simulation also has a long history in healthcare environments. Even before the advent of the modern personal computer, simulation had been used to study problems such as healthcare worker staffing, patient bed allocation, and medical decision making [40, 65]. More recently, modeling has been used to capture the effects of hospital architecture on the frequency of staff visit to patient beds [53], and simulation could be used to test potential architectural layouts.

Simulation has also been used to study the spread of infectious diseases. The first account of disease modeling was its use by Bernoulli in 1786 to defend the introduction of the smallpox inoculation [12]. Since then, the SIR model and its variants have been widely used to model and simulate disease [54], and more recently agent-based simulation has been explored [44]. While epidemiological simulation has been done at the macro level for some time, recently the outbreak of SARS in China and Canada [92] has highlighted the importance of hospitals as vectors for disease.

The aim of this work is to lay out a framework for the accurate simulation of healthcare environments by introducing models which leverage the increasingly fine-grained care data being made available by the recent digitization of healthcare data.

The simulator and models can be used to study a wide range of problems, including the spread of nosocomial infection, cost-effective delivery of care, and patient care issues.

Previous work in discrete-event agent-based simulation with applications in healthcare can be divided into two groups: those that use off-the-shelf operations management software or process definition languages [99, 24, 36, 75, 107, 110, 76], and those that build custom simulators to answer a very specific question [59, 17, 25]. In both cases, the range of applications for a particular simulator are very limited. There are three main problems with these approaches. First, repurposing existing software requires defining problems in terms of processes and resource usage, which may be appropriate for studying some optimization problems, such as bed occupancy, but it is not clear how such a simulator could be used to study other problems, e.g., the spread of infectious diseases. Second, these highly structured simulation frameworks seem problematic for creating a realistic mix of patients. While factory widgets can be assembled within strict tolerances, fully specifying patients with wide ranging diagnoses would be a significant challenge. Third, despite evidence that hospital architecture can greatly impact real healthcare worker interaction patterns [53], the vast majority of previous work either ignores location entirely, focuses on a small geographic area, or does not incorporate physical space meaningfully into modeling efforts. Indeed, at least one set of authors describes physical location in patient modeling as a "distraction" [17]. Our goal is to develop a simulation framework that is useful for a wide variety of applications.

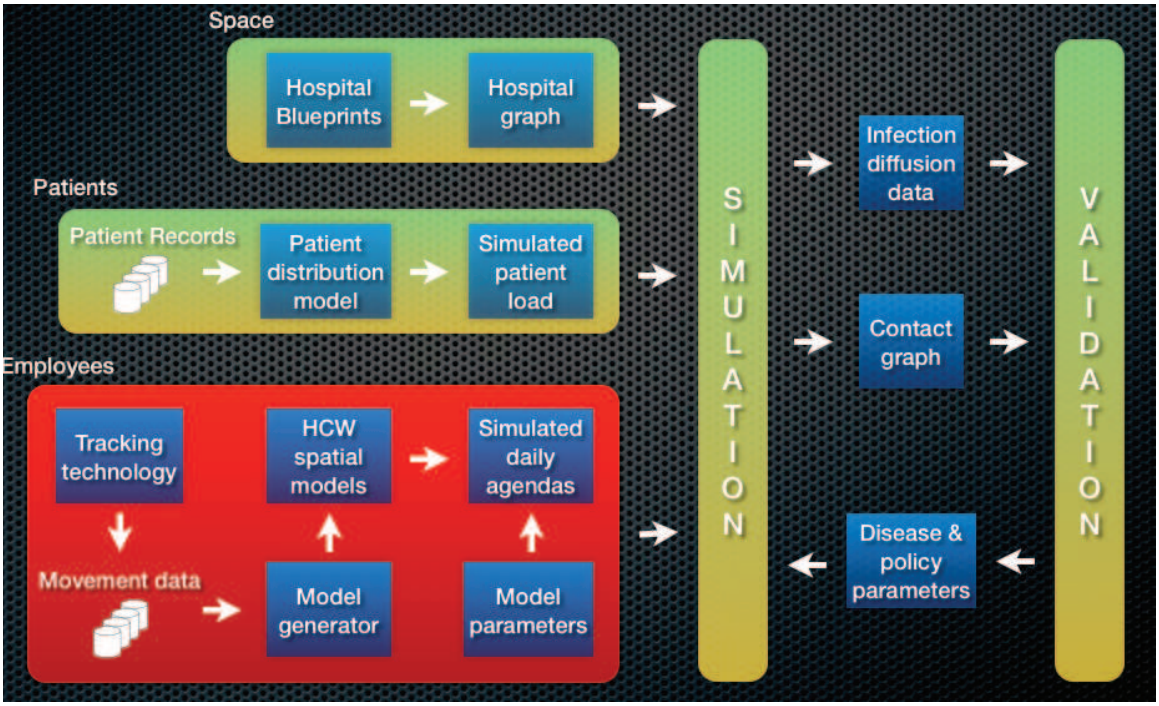

Figure 1.1: Spatial data, patient data, and healthcare worker data are the three main types of input used in our infectious disease simulation framework. See Chapter 2 for more information about healthcare worker modeling, and Chapter 3 for more information about patient modeling. Disease modeling and simulation as well as additional applications are discussed in Chapter 4.

Figure 1.1 shows the components in our simulation framework in the context of infectious disease simulation. Our simulator takes three main types of information as input: an architectural model, parameters for a healthcare worker model and parameters for a patient model. An architectural model of the hospital is the foundation upon which the patient and healthcare worker models are constructed. Our model of the physical space in a hospital is a graph-theoretic model, with a node for every room in the hospital and an edge connecting every pair of nodes that are directly connected (e.g., by a doorway). Healthcare worker and patient model parameters are used to generate agendas, which specify when and where these individuals are located at any given time.

The use of agendas to model movement is unique to our simulator, because our simulator explicitly models geometric architectural constraints. There are two main criteria for healthcare worker agendas. Healthcare workers must show up in each room an appropriate number of times, and the transition from one room to another must respect spatiotemporal constraints, i.e., a healthcare worker cannot reasonably move 250 meters in 25 seconds. We decompose the generation of healthcare worker agendas into two components. First, we build a model generator which produces probabilities a healthcare worker will be in any given room. Our model generator uses movement data from some healthcare worker tracking technology (e.g., nurse locator badges), to train generative models of healthcare workers. The models, in conjunction with model parameters such as the attractiveness of each room in the hospital and the number of nurses on duty, are used to generate agendas for a population of healthcare workers.

A more complex model of healthcare worker movement might even take patient load information as a parameter when generating healthcare worker agendas. There is pre-existing work in spatial modeling in ecology [104], criminology [16], and the study of patterns of human travel at a macro level [15], but our generative healthcare worker modeling draws in particular upon work in location-aware search [7]; to our knowledge, there is no previous work on the use of agendas in healthcare simulators because our simulator is unique in its attention to architectural constraints.

We then use patient records to train a number of patient models that generate agendas for inpatients and outpatients. A number of groups have developed models of patients for use in simulation [27, 41, 112, 8, 70, 62, 38, 40], typically by dividing the population into *compartments*, subpopulations within which individuals are considered interchangeable. Generally, patients are assigned to one of 3-4 compartments in the hospital and 1-2 compartments in the community. There might be a compartment in the hospital to hold acute patients, rehabilitative patients, and long-term patients. The community might have one compartment for the general population, and one for those receiving care in nursing homes. In our models, we have a compartment for every unit in the facility, and transition matrix specifying the probability of transfer from each compartment to every other compartment, as well as an expected length of stay before transferring. A patient agenda is a complete picture of what rooms a patient visits during their visit to the hospital including what rooms are visited and for how long.

The primary application for our realistic hospital simulator is infectious disease

simulation. Until recently, most infectious disease modeling relied on the mass-action assumption [81, 54] which essentially amounts to assuming that all individuals mix randomly with other individuals within their compartment. Meyers et. al show that under certain conditions epidemiological simulation with a random mixing assumption is a poor substitute for an agent-based simulator [81].

We would like to leverage newly available technologies such as Electronic Medical Record systems, nurse locator badges, and other RFID or mote-based technologies that are capable of producing fine-grained location data. Through the use of fine-grained training data for realistic hospital architectural, healthcare worker and patient modeling we intend to develop an accurate discrete agent-based (compartment size = 1) simulator suitable for use in comparing the relative effectiveness of a variety of infection control policies.

1.1 Modeling Healthcare Workers

The University of Iowa Hospitals and Clinics (UIHC) is a 3.2 million square foot facility that employs approximately 8,000 healthcare workers, of which about 4,000 are in the hospital at any given time. The UIHC has, at any given time, over 650 in-patients with an average 3-6 day length of stay. In 2007, 740,000 out-patients were treated at the UIHC. Given this scale of operations, there are numerous opportunities to implement new policies and procedures that could lead to improvements in patient outcomes while simultaneously reducing healthcare costs. Attempting to understand and exploit these opportunities can sometimes lead hospital administrators to ask

very specific questions. Two examples that motivate our work are the following.

(1) During vaccination shortages (e.g., the influenza vaccine shortage of 2004-2005 [111]), which healthcare workers or which categories of healthcare workers should we target for vaccination?

(2) We want to locate 80 time clocks in the UIHC facility for staff to punch in their hours. Where should we locate these in order to minimize overall access time?

There are numerous other examples: time-and-motion efficiency studies for improving healthcare delivery, which hospital units to move into a newly constructed wing, optimal placement of incoming patients with certain diagnosis, etc. In this paper, we present a computational approach for solving these and other similar problems.

Different classes of healthcare workers inhabit different parts of the hospital, and exhibit very different levels of mobility and temporal patterns. Administrators might have an office in which they spend most of their time, while inpatient nurses will regularly move between a small cluster of patient bedrooms. Residents and therapists see patients all over the hospital and are highly mobile. Motivated by work on estimating the spatial distribution of web search queries [7] and older work from spatial biostatistics [104], our approach utilizes the "electronic footprints" that healthcare workers leave behind in the hospital's electronic medical record system to estimate their spatial probability distributions within the UIHC facility.

1.1.1 The Hospital Metric Space and Login Data

After considering using the metric space implied by the Euclidean distance between pairs of room in the UIHC facility, we determined that straight-line shortest paths do not respect hospital geometry or represent reasonable "walking distances" between pairs of hospital rooms. We therefore impose a metric space on the UIHC facility that corresponds to the "walking distance" between pairs of hospital rooms. To construct this metric space, we manually created a graph model by examining detailed printed architectural drawings of the hospital and matching these drawings with a master spreadsheet of hospital rooms and their utilization, provided by the UIHC. We represented each room in the hospital by a node in the so called *hospital graph*, and for every pair of rooms in the hospital between which it is possible to move directly (e.g., through a doorway), we created an edge between the corresponding nodes. In order to have a consistent concept of distance, corridors and large rooms (e.g., cafeterias, atriums) are divided into smaller "room-sized" chunks (See Figure 1.2). Traveling along an edge of the hospital graph is roughly equivalent to walking 5-6 meters. For greater flexibility, we view each edge $\{u, v\}$ as two directed edges (u, v) and (v, u), allowing for distinct weights to be assigned to edges in opposite directions. This helps us in modeling certain natural preferences of people who move about in the hospital, such as using corridors rather than walking through other units (even if that meant a shorter walk) or using an elevator to go up the floors rather than a staircase.

The weights we assign to edges are all between 0.8 and 4.0. Somewhat arbitrar-

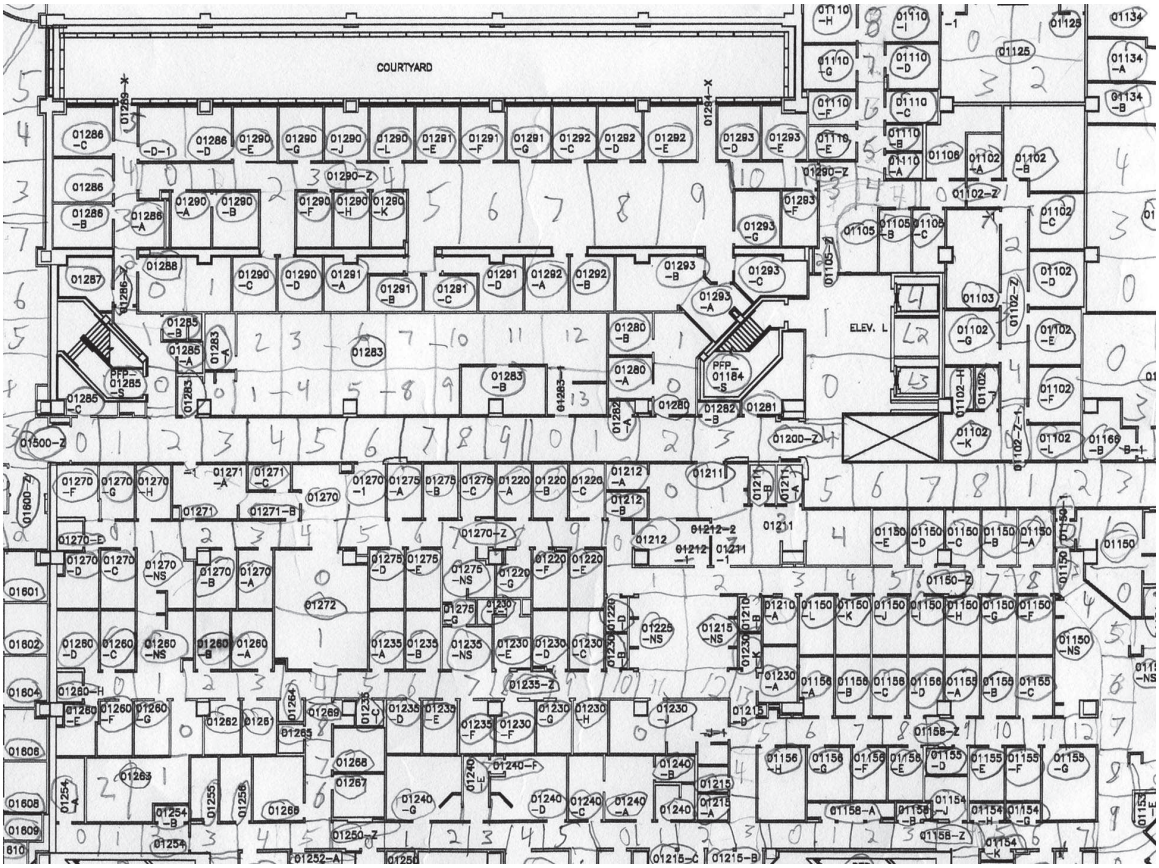

Figure 1.2: Notice that long hallways and large rooms have been subdivided into smaller, room-sized chunks.

ily, we fix these in the following order: walking down a corridor has the lowest weight (0.8 hops), followed by the normal room-to-room cost (1.0 hops), the cost of taking an elevator (2.0 hops, independent of the number of floors), the cost of descending a flight of stairs (3.0), and the cost of ascending a flight of stairs (4.0 hops). This ordering of weights is meant to roughly reflect the effort and time required to travel along an edge. The hospital graph that results from this construction has 18,961 nodes and 46,884 directed edges and we work with the metric space[1] induced by the shortest path distances on this graph. It is worth pointing out that distances in this metric space may be quite different from Euclidean distances between rooms seen as points in 3D space (see Figures 1.3 and 1.4).

The data that form the basis for our models and applications are healthcare worker login records to the UIHC electronic medical record (EMR) system. Each login record to the UIHC EMR system contains information on the healthcare worker initiating the login (uniquely anonymized) including the healthcare worker's current job code and department, the ID of the machine being used, the location of the machine, and the start time-stamp and the end time-stamp of the login. After filtering out roughly four million records with missing location information, and four million additional records with ambiguous or unusable location information, we were left with approximately 11.7 million records from a 22 month period beginning September 2006

[1]Technically, this is not a metric space since distance $d(u, v)$ may be distinct from distance $d(v, u)$. However, since the weights are in the range 0.8 to 4.0, symmetry is satisfied approximately and triangle inequality is satisfied exactly.

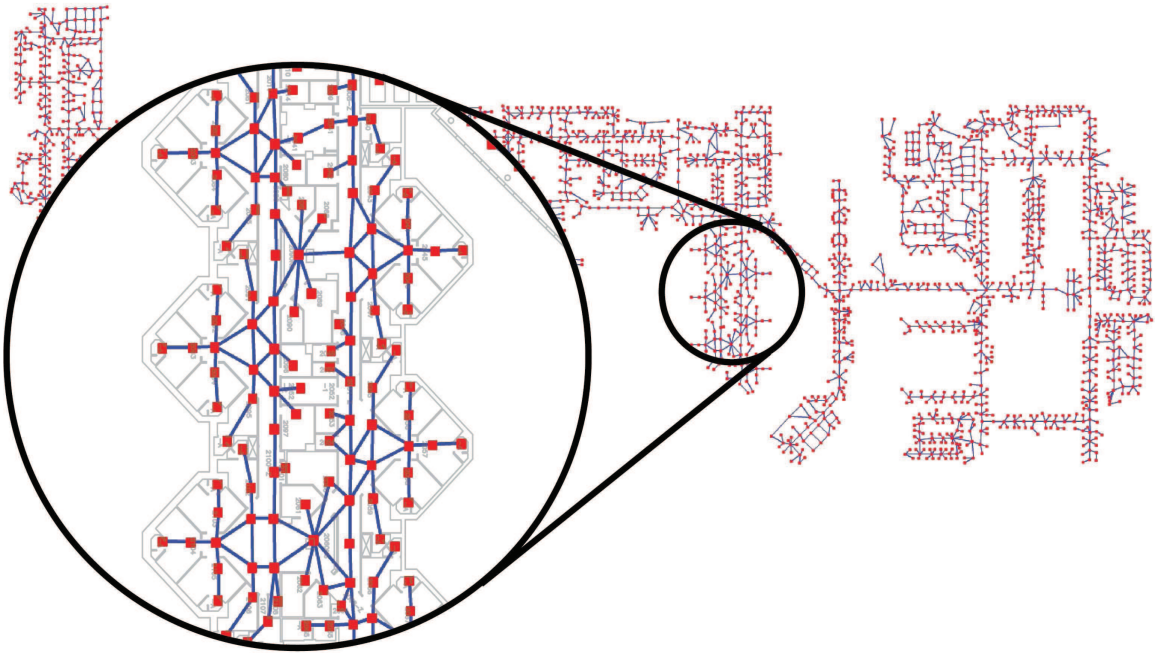

Figure 1.3: A small portion of the hospital graph corresponding to the second floor of the UIHC. Each room or corridor segment is represented by a node, connected by edges to adjacent rooms or corridor segments. This particular image was produced by superimposing the graph onto a CAD drawing of the floor plan.

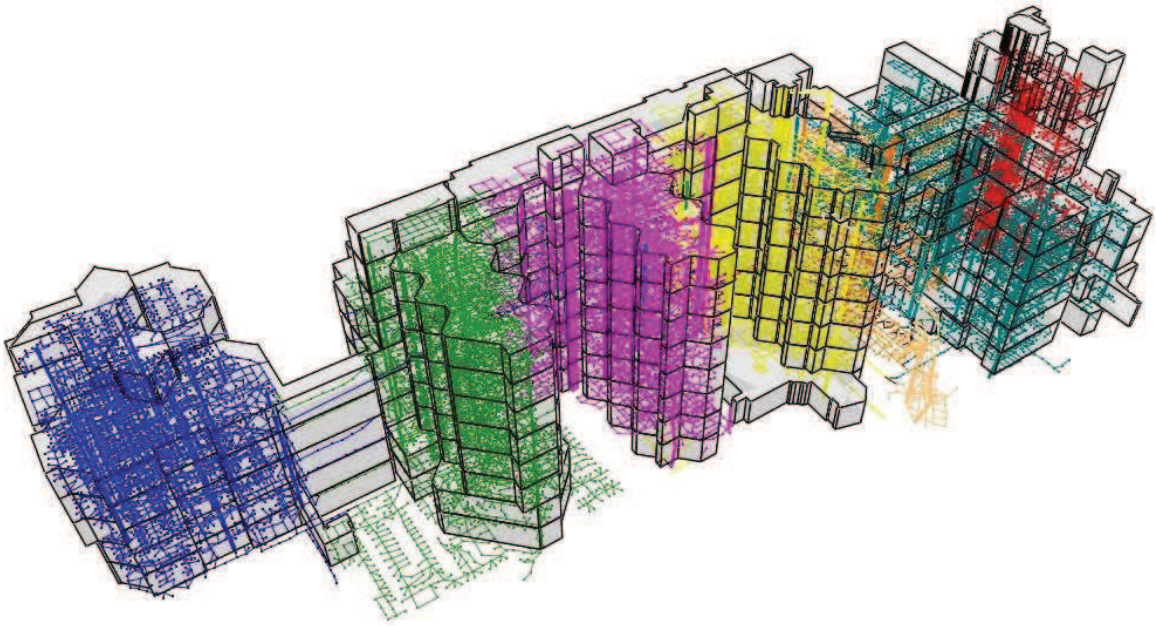

Figure 1.4: The entire hospital graph superimposed on a 3-dimensional architectural drawing of the hospital. Nodes are colored according to the building they belong to.

and ending July 2008. There are 14,596 healthcare workers who login to the EMR at least once during this period using 17,522 different machines distributed over 4,379 locations in the UIHC facility. The machine locations are well spread out in the hospital with the average distance between a room in the UIHC facility and a room with a machine being 2.816 ($\pm$ 2.258). Due to the small average distance to a machine, especially in patient care areas, and a privacy feature that terminates EMR sessions after a period of inactivity, it is reasonable to assume that EMR activity in an area strongly correlates with a healthcare worker spending time in that area.

While the vast majority of healthcare workers access the EMR, there are certain groups of healthcare workers, e.g., housekeeping, information technology, and janitorial staff, who rarely or never access the EMR. Further, we have anecdotal evidence of healthcare-worker behaviors that could bias the EMR data in different ways, e.g., when a group of healthcare workers visits a patient in an intensive care unit, interacting with the EMR tends to be the responsibility of junior-most staff member. In future work one could deal with such "holes" in the EMR data by using a variety of other data gathering mechanisms, including the use of wireless technology [95, 56, 61].

1.1.2 Healthcare Worker Location Model

There are a number of different approaches for developing generative spatial models for healthcare workers. One possibility is using discrete models. For example, each healthcare worker could have an independent probability of being in each room. Ultimately, we use continuous models similar to those described in Backstrom et al.

because they are well understood, compact, and have nice properties allowing for efficient model training [7].

We start with the assumption that every healthcare worker has a geographical *center* for their activity and that their spatial distribution polynomially decays as we move away from the center, where the rate of decay is determined by two additional parameters. Our model is inspired by the work of Backstrom et al. [7] who partition the geographical region of the U.S. into grid cells and estimate, using a similar center and dispersion based model, the probabilities of different queries q being issued from different grid cells. While our model is similar to their model in these aspects, there are important differences due to the fact that queries and people (specifically, healthcare workers) are after all fundamentally different entities. The same query, for example for the "New York Yankees," can be issued from multiple locations simultaneously whereas a healthcare worker can be in at most one location in the hospital at any time. Our primary goal here is to estimate a static spatial distribution for each healthcare worker, and to do this we suppose that each room v has an associated parameter α_v that denotes, in a loose sense, the "attractiveness" of room v to healthcare workers. Certain UIHC rooms, e.g., the nurse's station in a unit, may see a lot more traffic than a nearby patient room, and such rooms will have a correspondingly higher α_v value.

Thus for a setting with m healthcare workers and n rooms, we need to estimate $3m + n$ model parameters. Solving for the maximum likelihood estimators of the model parameters could potentially require solving $\Omega(mn)$ continuous optimization

problems. On the face of it, this is computationally infeasible (since m is roughly 15,000 and n is roughly 19,000). In Section 2.4 we describe algorithms that take advantage of properties of the underlying metric space to prune away from the search space many of the UIHC rooms for each healthcare worker. These algorithms make the model parameter estimation computationally feasible.

1.1.3 Multiple Centers

While the single center models adequately fit the observed data for a significant number of healthcare workers, there are some healthcare workers for which multi-center models are much better at fitting the observed data. We demonstrate how multi-center models may be appropriate for some users by walking through a typical day and examining observed logins for selected healthcare workers. We demonstrate that multi-center models do little to improve a subset of users expected to be adequately modeled with single-center models, but do a vastly superior job of fitting EMR data for a subset of healthcare workers who have clusters of activity in different areas of the hospital.

We consider two classes of multi-center models. *Additive multi-center models* are models where each center contributes some (possibly very little) probability of activity to each room in the graph. *Disjoint multi-center models* are those in which activity in a certain room is solely determined by the center having the greatest influence at that room. That is, additive multi-center models add the probability vectors for each center before scaling, while disjoint multi-center models take the

maximum probability for each room over each center.

We show that unlike the single-center models, neither class of multi-center models is easily optimized numerically, meaning that you may end up with locally and not globally optimal solutions. We thus focus on heuristics for finding good multi-center models; we will discuss why the use of heuristics is appropriate in Chapter 2.

1.1.4 Validation

Given our interest in building a realistic hospital simulator, we now consider how to verify that our models generate activity that in some meaningful way conforms to reality. Our review of healthcare simulation research identifies a number of techniques for validating models and simulators: (1) internal validation (e.g., cross-validation, sensitivity analysis) [17, 24, 59, 2], (2) checking properties of generated data against known quantities [2, 24, 75, 76, 59], (3) visualizing generated data and presenting it to healthcare professionals [36, 17, 99, 24, 75], (4) verifying results are similar for a second dataset [17], and (5) software verification (e.g., unit testing, sanity checks via assertions) [99, 24].

We perform the first three types of validation. For example, (1) we compare models trained on different subsets of our data to verify they are similar, (2) we compare generated locations for Pediatric healthcare workers with the known location of that department, and (3) we compare differences in generated data for various job types with the expectations of those familiar with hospital operations. Note that because this work represents the first effort to model the physical location of

healthcare workers based on fine-grained data, and due to the challenges associated with obtaining the fine-grained location data required to construct our healthcare worker models, we were unable to repeat the entire process at a second facility for validation purposes. We expect the necessary data to become increasingly available and the process increasingly automated over the coming years. While we did not develop a full, formal specification of our software to use for software verification, we do use assertions heuristically to perform sanity checks on certain variables.

1.2 Modeling Patients

No model of a hospital would be complete without a model of patient movement, or flow. Many patients have compromised immune systems and are at an elevated risk of morbidity and mortality, making them an essential element in a realistic simulation of disease, as well as simulations in support of answering operational questions. Patient models are also essential in the study of various resource allocation problems and clearly critical when examining quality of care issues. Here we seek to develop a realistic model of patient care which can be used in simulation to study a wide range of problems.

1.2.1 Model Description

Previous work in patient modeling makes widespread use of compartmental models [70, 62, 38, 40]. Models range in complexity from having just one compartment for acute patients and another for long-term care patients to models having four hospital compartments and two community compartments. In order to take advantage

of the fine-grained data available, we propose a model with one compartment for each patient care unit or clinic in the hospital, and one compartment situated in the community.

Model training is based on a data set from the University of Iowa Hospitals and Clinics. The data set contains every inpatient and outpatient admission, transfer, and discharge (ADT) between January 1, 2006 and June 1, 2009. The data set allows us to nearly completely follow the path of each patient that set foot in UIHC over a 3.5 year period, and additional information about the patient (e.g., patient diagnosis).

We present two different patient models, both of which are trained with the ADT. Each patient has a length of stay assigned from the overall length of stay observed in the compartment he or she is entering. Admissions, transfers and discharges are modeled by a complete graph with edge weights giving transition probabilities from each compartment to every other compartment.

The first patient model is a very simple model where some percentage of patient beds is assumed to be filled at all times. Patient lengths of stay are drawn from a log-normal distribution, and a new patient immediately replaces discharged patients.

The second model uses one compartment for each unit in the hospital. Transfer probabilities and compartment length of stay times are a function of other parameters such as admission time. New patients are admitted after some delay (dependent on the time of day). A second set of length of stay distributions and compartment-to-compartment transfer probabilities is used for patients that have been transferred out of an intensive care unit while it is at or near capacity. Previous work suggests

that such patients may be more likely to be readmitted to intensive care, have longer lengths of stays, and have worse outcomes [39, 21, 19, 108].

1.2.2 Training Parameters

We train our patient models based on three years of UIHC admission, discharge and transfer data (ADT). This data set gives us a complete picture of patient flow for both inpatient units and outpatient clinics down to the time of day a patient was transferred and which bed of which room the patient was transferred to. We also have information about which nursing station and which service the patient is assigned to as well as the patient's primary diagnosis.

Patients in each compartment have some waiting time and some probability of transferring to each other compartment, including discharge into the community. As with all statistical problems, the more complex the model, the more data is required to get a reasonable fit. In cases where the set of training data for each compartment is very small, we determine admission and transfer rates by aggregating data sets from similar compartments.

In our simple model, we need only calculate the average number of beds occupied at any given time and find the maximum likelihood log-normal length of stay distribution. Training our second model requires calculating length of stay distributions and compartment-to-compartment transfer probabilities for each compartment, as well as the second set used for patient transferred out of an intensive care unit at or near capacity.

1.2.3 Validation

Our patient model validation faces similar issues to those outlined in discussion of healthcare worker model validation (Section 1.1.4). As with our healthcare worker model validation, we use internal validation, comparison with expert predictions (on a much smaller scale), and software verification. In this case, however, our training data represents a nearly complete picture of patient flow, so we are able to compare, e.g., patient load in each unit in the observed data with that in the generated data. While we do not have inpatient flow data for a second facility, in this case we are able to verify that the rate of intensive care readmissions in our simulator are in line with those in our observed data as well as other real world patient datasets reported by other groups [19, 79].

1.3 Applications for a Hospital Simulator

A realistic hospital simulator (such as the one described here) can be used to study a range of problems. Hospitals can act as loci and amplifiers of infection during an epidemic, and hospital-acquired infections have long been an issue. And while epidemiological simulation is the primary application we study with our hospital simulator, a hospital simulator could also be used to study staffing issues, patient bed allocation, and the placement of expensive resources such as crash carts.

1.3.1 Epidemiological Simulation

Infectious diseases are responsible for about 15 million deaths annually worldwide [43], and hospital-associated infections are responsible for an estimated 100,000

deaths annually in the United States alone [73]. The 2003 outbreak of severe acute respiratory syndrome (SARS) in Chinese and Canadian hospitals highlighted the fact that infections acquired in the hospital can also be passed back into the community [92]. In fact, 20-40% of infected individuals in the 2003 outbreak of SARS were healthcare workers [90]. Additionally, hospital-acquired infections are a major source of morbidity and mortality, affecting about 2 million Americans every year, and costing over $4.5 billion [50]. A realistic infectious disease simulator allows us to gain insight into the dynamics of infectious disease in the hospital environment and shed light on the relative efficacy of various infectious control policies.

Infectious disease modeling dates from the eighteenth century when Daniel Bernoulli developed a smallpox disease model to argue that smallpox variolation (i.e., intentionally infecting individuals through exposure to a mild case of the disease), was good for the population of England as a whole even though some individuals might become ill or even die as a result [13]. In the twentieth century, disease models incorporated multiple stages of infection [72]: Hethcote offers an excellent survey of such SIR-based models [54]. In the simplest case, the population is treated as homogeneous and the mass-action principle is used to derive a set of three equations governing the transfer between the susceptible, infected, and recovered stages of disease.

While models based on uniform or random mixing assumptions have their uses, it has been argued recently by Newman and Meyers that infectious disease models that assume random mixing can be inappropriate in cases where populations are believed to mix heterogeneously [86, 82, 9]. Specifically, Newman considers contact networks

that exhibit "small-world" properties, and shows that disease simulation with and without the random mixing assumption predicts very different epidemic behavior. Recent work suggests that healthcare worker contact graphs may indeed exhibit small world properties [61, 29, 28], so in this thesis we tend, in the limit, towards agent-based infectious disease simulations (e.g., one compartment for each agent in the simulation rather than one compartment for the entire population). We find that healthcare worker contact networks based on our agent-based simulator do indeed exhibit classic "small world" properties [117, 87], and thus agent-based simulation is critical for understanding and controlling the spread of hospital-acquired infections such as "Clostridium difficile" (C. diff), methicillin-resistant "Staphylococcus aureus" (MRSA), or vancomycin-Resistant "Enterococcus" (VRE).

Our agent-based infectious disease model has a number of parameters making it flexible enough to handle a wide variety of infectious diseases, though we focus in particular on mumps and flu. We obtain parameter values from existing literature on these diseases [54, 14, 3, 102, 93, 106, 80, 78] , and we perform sensitivity analysis to confirm that our results are robust in the face of uncertainty in parameter estimation.

As a sample application of infectious disease simulation, we simulate the spread of mumps within a hospital environment with three different infection control policies in place: (1) Do not quarantine healthcare workers, (2) quarantine healthcare workers displaying symptoms for 5 days, or (3) quarantine healthcare workers displaying symptoms for 9 days. This application is motivated by the fact that the Centers for Disease Control and Prevention, the Healthcare Infection Control Practices Advisory

Committee, and the Academy of Pediatrics all recently changed their mumps quarantine policy from recommending 9 days of quarantine following the onset of symptoms to 5 days [1]. We use an SIR-based model of infectious disease with transmission parameters based on the mumps literature and run discrete agent-based simulation until all agents are either susceptible or recovered. This application is very similar to other work done by the CompEpi group [96, 57].

1.3.2 Resource Allocation Problems

Hospitals are necessarily interested in finding ways to reduce operational costs while maintaining or increasing the quality of patient care. One example that arose from our own UIHC is the placement of time clocks in the facility. One possibility would be to place the clocks near all of the entrances, but administrators calculated that due to the size of UIHC (it can take 15 minutes to walk from one end to the other), placing clocks near where employees actually work would save millions of dollars annually. We use the healthcare worker spatial distributions to find near optimal placement for k time clocks, where the objective is to minimize the average extra distance a healthcare worker has to travel in order to punch in their hours. This application is a canonical example of how the problem of locating resources (e.g., medical equipment, crash carts, pharmaceutical stores, etc.) in a large hospital might be approached.

1.3.3 Intensive Care Recidivism

Readmission to the intensive care unit shortly after being transferred out is strongly associated with increased risk of mortality, longer lengths of stay, and higher costs of care [39, 22, 10, 74, 23]. There is some agreement in the intensive care recidivism literature about medical problems that patients who are readmitted to intensive care share, but there is little to no agreement on what might lead to recidivism.

We take advantage of our fine-grained patient data sets to help shed some light on what leads to intensive care recidivism. In particular, we present evidence that a patient's chance of being readmitted to an intensive care unit goes up as the number of fellow patient in intensive care goes up. A working hypothesis is that as the population of an intensive care unit rises, the incentive for healthcare workers to transfer "marginal" cases out of intensive care increases. We look at the chance of a particular patient being "bounced back" to intensive care within 48 hours relative to the number of patients per healthcare worker on the day the patient was transferred out.

Using the generative healthcare worker models from Section 2.7, we examine how alternate staffing policies might affect recidivism.

1.4 Broader Context

The primary contributions of this thesis are: (1) it is the first agent-based hospital simulator based on fine-grained data (i.e., no random mixing assumption), (2) we present generative healthcare worker and patient models which are compact,

flexible, and can be trained efficiently from real data, and (3) we show how (1) and (2) can be combined to address a variety of practical questions. Early purely analytical work [54] has been more recently followed by studies of the spread of infectious disease on arbitrary contact networks [81], and later, more specifically, contact networks within hospitals [116, 113, 114]. However, this work is the first hospital-wide simulator based on fine-grained location data suitable for operational modeling and agent-based simulation.

1.5 Organization of this Thesis

In Chapter 2, we will introduce the University of Iowa Hospitals and Clinics in more detail, discuss our spatial model, and discuss healthcare worker modeling including both the single-center of multi-center variants. The chapter will also include validation and applications specific to the healthcare worker models and a discussion of how the models are used to generate data for use in simulation. Chapter 3 will focus on patient modeling. It will discuss the benefits of each of two patient models, and issues with training the more complex models. It will also include model validation, patient-centric applications and a discussion of how the models are used to generate data for use in simulation. Chapter 4 will look at applications that combine both healthcare worker and patient models. The primary application incorporates healthcare worker and patient data with infection diffusion data to simulate nosocomial infections and the effect of infection prevention policies.

CHAPTER 2
MODELING HEALTHCARE WORKERS

2.1 Introduction

Our agent-based simulator requires as input some form of *agendas*, i.e., planned trajectories for each HCW participating in the simulation. These agendas should respect temporal and spatial constraints (i.e., a HCW can only move directly between adjacent room within UIHC), reflect different activity levels in various parts of the hospital at different times of day, and be generated efficiently and non-deterministically. We thus are interested in identifying sources of HCW location data, as well as compact, generative models, thousands of which can be trained within a day or two on modern hardware.

Location data is increasingly a subject of interest in healthcare settings. Recent work has looked at small deployments of wireless sensor devices to gather HCW location location data and contact networks. Heo et al. use radio-frequency identification (RFID) devices to study the effects of architecture of nurses' movement patterns, e.g., the frequency a nurse visits certain patient rooms [53]. Other groups use wearable wireless technology within a healthcare setting. Olguin et al. instrumented nurses in a patient care unit with badges and sensors capable of reporting physical activity, speech, and proximity to other badges to study delays in patient care [89]. Isella et al. use wearable sensor to look at mixing patterns within a healthcare setting [61], and other groups have used various means to study contact networks

of healthcare workers [119, 33, 116]. The CompEpi research group has also developed wearable badges to automate hand-hygiene monitoring, and study the contact networks of HCWs [56, 95].

In addition to technologies that directly gather location data, a number of systems that gather location data as a side-effect have been introduced. Jha et al. studied the use of electronic health records (electronic medical records that can be shared between different facilities) around the world and in the United States [66, 67], where $10 - 30\%$ of ambulatory care physicians use an electronic health record (the UK, Netherlands, Australia and New Zealand have EHR usage over 90%). Jha et al. discuss the cost-reduction and patient-care benefits of health information technology and EHR usage. Given the benefits, it seems quite likely that new technology solutions will continue to be introduced into hospitals in the short-term within industrialized nations. One benefit of EMR / EHR introduction is that access logs provide a sample of healthcare worker location data, at least in some implementations. In particular, the UIHC EMR implementation does record room-level location data in access logs.

While there are clear benefits to developing and deploying technologies like RFID in healthcare settings, we are interested in using the data collected by EMR systems or wearable sensors to develop a realistic simulator of a healthcare environment. RFID and other wearable devices are very appealing due to their ability to capture a large and fine-grained sample of location data, however, it is very expensive to retrofit an entire facility with the necessary infrastructure. Given our interest of simulating a facility rather than a single unit within a facility, we focus on the sample

of room-level location data made available by EMR systems.

In Section 1.1.1, we introduced the metric space in our model of UIHC. In Section 2.2 we discuss previous work in spatial modeling, and then we discuss our modeling efforts in the remainder of this chapter.

2.2 Previous Work

Because it is unethical to infect people to study the spread of disease within a healthcare environment, physical experiments are problematic. In other fields where this is the case (e.g., astronomy, economics), researchers must rely on naturally occurring experiments to answer questions of interest. Here, we develop generative models that can be used to make up realistic data to test, e.g., the efficacy of various infection control policies.

A number of groups in various fields have been interested in identifying patterns in samples of location data. Urban economics is concerned with the geographic distribution of urban populations, and work in the field has modeled population density as falling off according to an exponential function of distance from the central business district or other population center [47]. While the extension from single-center to multi-center models is of interest, the assumption that center locations are known a priori is problematic, and it is not clear how to guarantee properties of aggregated activities of multiple models that share the same space.

Several groups have modeled population levels of wildlife using autologistic models [6], linear models [6], gap analysis [37], fractal analysis, and correlated ran-

dom walks [77]. Sharov et al. used exponential models fit by linear regression to study distributions of gypsy moth populations [104]. Similar techniques are used to model the spatial distribution of criminal activity such as assaults [48, 16]. These wildlife models can predict wildlife populations on various tracts of land using sample population data, satellite imagery from which properties of the landscape can be determined, etc. However, HCWs are not searching or foraging, their movement is on a very different scale, their environment is fairly stable over time, and we are interested in saying something meaningful about the movement of individuals in addition to aggregate population levels.

Physicists have used spatial clustering techniques to study the cosmological distribution of dark matter holes [84], and Lévy flights have been used to model the spatial distribution of wildlife and human travel [77, 15]. Brockmann et al. used Lévy flights to model human travel on geographic scales based on the tracking of bank notes across the United States [15]. In the case of dark matter holes, clustering is determined by natural processes (e.g., gravitational attraction) the effects of which the modeling attempts to capture. Again, it is not obvious how to use Lévy flights for individuals and ensure that the proper amount of activity takes place in each room, or that one could capture the idea that HCWs have areas in the hospital in which they spend much of their time, with occasional forays into other areas of the hospital.

The recent need to analyze and mine large databases of spatial data has led to research on a number of techniques for identifying clusters of spatially related points within geographic data sets [42, 11, 71, 51]. This work has to make assumptions

about the number and density of points in each cluster, and/or the shape of each cluster. Similarly, facility location work from the operations research field [115, 91, 103, 105, 49] can also be used to identify clusters of points. These techniques could be useful for determining or validating the number of clusters of activity that HCW models should generate or exhibit.

2.3 Model

In this thesis, we use Backstrom et al.'s work modeling the spatial distribution of search engine queries as the basis for a generative model for healthcare worker movement [7]. For the reasons outlined in Section 1.1.2, we adapt this model model with continuous, polynomial decay functions. While we implemented a number of other decay functions (e.g., exponential, Gaussian), it was not obvious how to directly compare models using different decay functions, and we focus on polynomial decay functions as was done by Backstrom and Griffith [7, 47]. We leave a more careful consideration of alternative decay functions for future work.

Our goal is to estimate the spatial distribution of each of the 14,595 HCWs that appear at least once in our EMR login database. We associate with each HCW i three parameters: a *center* c_i (a room in the UIHC facility), a *dispersion* $\gamma_i \geq 0$, and an *ownership* constant, which says something about the percentage of the activity near the HCW's center that is attributable to that HCW. $\beta_i \geq 0$. Fix a time window T (e.g., T might be a week) and assume that, for any room v in the UIHC facility, the probability that a person chosen randomly from room v at a time instant in T is

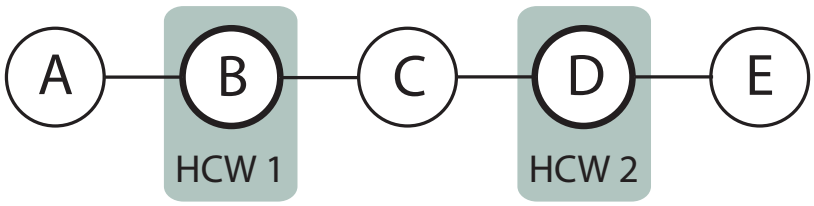

Figure 2.1: A simple 5-room hospital graph populated by two HCWs whose centers are at nodes B and D respectively.

the HCW i is proportional to

$$\beta_i \cdot \Big(d(c_i, v)\Big)^{-\gamma_i}. \tag{2.1}$$

Here, $d(c_i, v)$ is the distance (≥ 1) in the hospital metric space (see Section 1) between nodes v and c_i. For a room v, the HCWs who are most likely to be found in v are those whose β parameter values are high, whose centers are near v, and whose dispersion parameter values (i.e., γ) are small, implying a spatial probability density function that has not fallen off too much before reaching v. While this models tells us for each room v which HCW is more likely to be found at v, it does not directly provide spatial distributions for the HCWs.

Room	Probability of picking HCW 1 / Probability of picking HCW 2	
A	2^{γ}	
B	2^{γ}	
C	1	
D	$2^{-\gamma}$	
E	$2^{-\gamma}$	

Table 2.1: If a HCW is drawn at random from rooms closer to HCW 1's center than HCW 2's center, it is more likely to be HCW that is picked.

The simple example (in Figure 2.1) clarifies this point. The figure shows a "toy" hospital graph with 5 rooms. Suppose that this hospital graph is populated by two HCWs, labeled 1 and 2, whose centers are nodes B and D respectively. Assume that these two HCWs have identical values for β and γ (i.e., $\beta_1 = \beta_2$ and $\gamma_1 = \gamma_2 = \gamma$). Then room C is equally likely to see HCWs 1 and 2, whereas room A is 2^γ times more likely to see HCW 1 than HCW 2 and, symmetrically, room E is 2^γ times more likely to see HCW 2 than HCW 1 (see Table 2.1) . Thus the model only posits for the two HCWs, relative probabilities of occupying various rooms and does not require, for example, HCW 1 to be more likely to be found in room A than in room E.

To obtain absolute probabilities of occupying each room, we make two assumptions. First, we assume that each HCW is located somewhere in the hospital, and thus we scale the probabilities the HCW is in each room such that they sum to 1. Second, we assume that each room in the hospital has a static "attractiveness", which determines the total amount of activity that should be generated in that room across all HCWs. Attractiveness may model, for example, the presence of commonly used equipment, a room with a patient as opposed to one without, or a private office as opposed to a shared workspace. We associate with each room a parameter, α_v, such that the generated activity for all users in room v sums to the attractiveness. The spatial probability distribution of a HCW i over all the UIHC facility rooms is obtained by normalizing the probability decay function values and the α values.

$$\{\alpha_v \cdot \beta_i (d(c_i, v))^{-\gamma_i} \mid v \text{ is a node in the hospital graph}\}. \tag{2.2}$$

Let s_i be the normalizing factor for HCW i, i.e., s_i is a value such that

$$s_i \cdot \beta_i \cdot \sum_v \alpha_v \cdot (d(c_i, v))^{-\gamma_i} = 1. \tag{2.3}$$

Given these spatial distributions for the HCWs, the expected number of HCWs in a room v at any time instant is

$$\alpha_v \sum_i s_i \cdot \beta_i \cdot d(c_i, v)^{-\gamma_i}, \tag{2.4}$$

where the sum is over all HCWs i.

As a special case, if we assume that all of the α_v's are identical then the spatial distribution of each HCW i is "nicely" shaped and completely determined by the parameters c_i, β_i, and γ_i.

2.4 Algorithm

Our algorithm takes as input timestamped location data (i.e., EMR login data in our case), and our hospital graph, and returns a maximum-likelihood center location, ownership constant, and dispersion constant for each HCW. That is, our algorithm finds and returns the center that is most likely to generate the observed data. In our implementation, we use the location data in an aggregated form to calculate attractiveness, but other implementations could either take attractiveness values as parameters or learn them during the maximum-likelihood estimation.

For a given HCW i, we use a maximum-likelihood approach to find a center c_i, dispersion γ_i, and constant β_i. To do this, we first partition the set of EMR logins into two subsets: L_i, consisting of only the logins associated with HCW i and $\overline{L_i}$,

denoting the complement, i.e., the set of logins with which HCW i is not associated. For each login ℓ, let $\mathrm{loc}(\ell)$ denote the location in the UIHC facility at which the login occurs. Then the log of the likelihood that the EMR login records are obtained by sampling from from the distribution of HCWs in each room is:

$$LL_i(c,\beta,\gamma) := \sum_{\ell \in L_i} \log\left(\beta \cdot dist(c, \mathrm{loc}(\ell))^{-\gamma}\right) + \sum_{l \in \overline{L_i}} \log\left(1 - \beta \cdot dist(c, \mathrm{loc}(\ell))^{-\gamma}\right). \tag{2.5}$$

Our problem is to find values of c, β, and γ that maximize this function. These maximizing values are the maximum likelihood estimators of the model parameters for HCW i. Note that we work with log-likelihood instead of likelihood because multiplication of large numbers of small probabilities quickly leads to underflow in practice.

Backstrom et al. [7] show that for a fixed center c, the function $LL_i(c,\beta,\gamma)$ has exactly one local maximum over its 2-dimensional parameter space $0 \leq \beta \leq 1$ and $\gamma \geq 0$. Thus, for a given room c, optimal values of β and γ can be found by any one of various local continuous maximization techniques; we have implemented the well-known *Nelder-Mead* local optimization algorithm [85]. To maximize $LL_i(c,\beta,\gamma)$ we can simply do a brute-force search, calling the Nelder-Mead algorithm for each candidate center c. However, this approach is computationally problematic because there are roughly 19,000 candidate centers (i.e., rooms in the hospital) and $LL_i(c,\beta,\gamma)$ has to be maximized separately for about 15,000 HCWs. This implies more than a quarter billion calls to the Nelder-Mead subroutine. Furthermore, we want to estimate

spatial distributions for HCWs from various subsets of login records (e.g., separately for each month, separately for night and day, etc.) and this requires having to repeat the entire computation 15-20 times.

We employ two tricks to significantly reduce the running time of our computation. We first preprocess the login records, employing a data reduction technique which groups or *buckets* logins with unique (HCW, location) pairs. For example, we compress two EMR logins, one for 2 seconds and one for 3 seconds, from Room 4200 by the HCW with userID 22 to be (5, 4200, 22). The first entry, 5, refers to the total amount of login time of HCW 22 into Room 4200. In a second step, after fixing a candidate center c, we bin rooms at equal distance from the center c into one compressed bin just before the call the Nelder-Mead subroutine. Note that this second level of data reduction is made possible by the fact that there are a limited number of discrete distances in our graph-theoretic model. These two data reduction steps of the logins significantly reduce, by at least an order of magnitude, the number of logarithm and exponentiation operations in practice.

Figure 2.2 shows how well the best model located in each room fits the observed EMR activity for a given physician's nurse assistant. The brute force search needs to consider locating the center in every room in the hospital, but note that the good solutions are are clustered in one area of the hospital.

Our first attempt at an improved algorithm, suggested by a discussion in [7], uses a search pruning technique to significantly reduce the number of candidate centers we need to consider. The technique is similar in spirit to hierarchical clustering

Figure 2.2: This figure shows the quality of fit for the best models located in each room of the first floor for a physician's nurse assistant. The best overall center is located in the room at the center of the red circle, and the quality of fit ranges from poor (blue) to the best (red). These models are trained using EMR data from March, 2007.

used elsewhere [68, 121, 26], although we cluster arbitrarily rather than to maximize some objective function. Our pruning algorithm starts with a single cluster consisting of the entire hospital graph. An arbitrary node in the graph is taken to be the cluster center and the radius of the cluster is bounded above by the diameter of the hospital graph. In a typical iteration, the algorithm considers a collection of clusters all of which have radii bounded above by r_i. The goal in this iteration is to partition each cluster (if necessary) into clusters of radius at most $r_i/2$. For any cluster C that has radius r such that $r_i/2 < r \leq r_i$, we repeatedly pick an arbitrary node v in C, make v a cluster center of a cluster C' and assign to the cluster C' all nodes in C that are at a distance at most $r_i/2$ from v. We then delete C' from C and continue until C becomes empty.

This hierarchy of clusters can be naturally viewed as a tree with the cluster consisting of the entire graph at the root and with the leaves being singleton clusters of radius 0. A greedy search of this tree essentially prunes those subtrees where the root of the subtree fails to meet some solution quality criteria (i.e., the best model for a HCW using the pruned node as the center location is dominated by one of its siblings. Note that this tree has height $O(\log(\text{diam}))$, where diam is the diameter of the hospital graph.

As a result of this pruning, we examine only 3% of the rooms in the hospital on average per HCW, though use of this heuristic sacrifices optimality. Figure 2.3 shows the results of the search using this clustering heuristic. Note that in the case of this particular Physician's Nurse Assistant, even though the log-likelihood in each

room shown in Figure 2.2 appears to be very smooth, the heuristic fails to find the optimal model.

In our work, we take advantage of the fact that the distances in the hospital graph form an approximate metric to prune the search space of candidate centers without sacrificing optimality. Given an upper-bound r on the distance between the center c of a cluster and any other node in the cluster, it possible to obtain an upper-bound on the maximum value of $LL_i(v, \beta, \gamma)$ for any room v in the cluster. The intuition here is that we can take the compressed login activity for the HCW under consideration, move all of that HCW's logins r weighted-hops closer to c, and all other logins r weighted-hops away from c. Finding the log-likelihood with these modified login sets will give the best log-likelihood possible for any potential center within r hops of c. Consider the following 4-parameter function:

$$MLL_i(c, \beta, \gamma, r) := \sum_{\ell \in L_i} \log\left(\beta \cdot (d(c, \mathrm{loc}(\ell)) - r)^{-\gamma}\right) + \sum_{\ell \in \overline{L_i}} \log\left(1 - \beta \cdot (d(c, \mathrm{loc}(\ell)) + r)^{-\gamma}\right) \tag{2.6}$$

Given a cluster C with cluster center c and radius r, we can use the Nelder-Mead algorithm to solve the continuous optimization problem to find values of β and γ that maximize $MLL_i(c, \beta, \gamma, r)$. Our claim, expressed precisely in the following theorem, is that the maximum value of $MLL_i(c, \beta, \gamma, r)$ (for fixed c and r) is an upper bound on the maximum value of $LL_i(v, \beta, \gamma)$ for any node v in cluster C.

Theorem 1 *Fix a node c and let $r > 0$. For any node v, if $d(c, v) \leq r$ then $MLL_i(c, \beta_c, \gamma_c, r) \geq LL_i(v, \beta_v, \gamma_v)$ where β_c and γ_c are values that maximize*

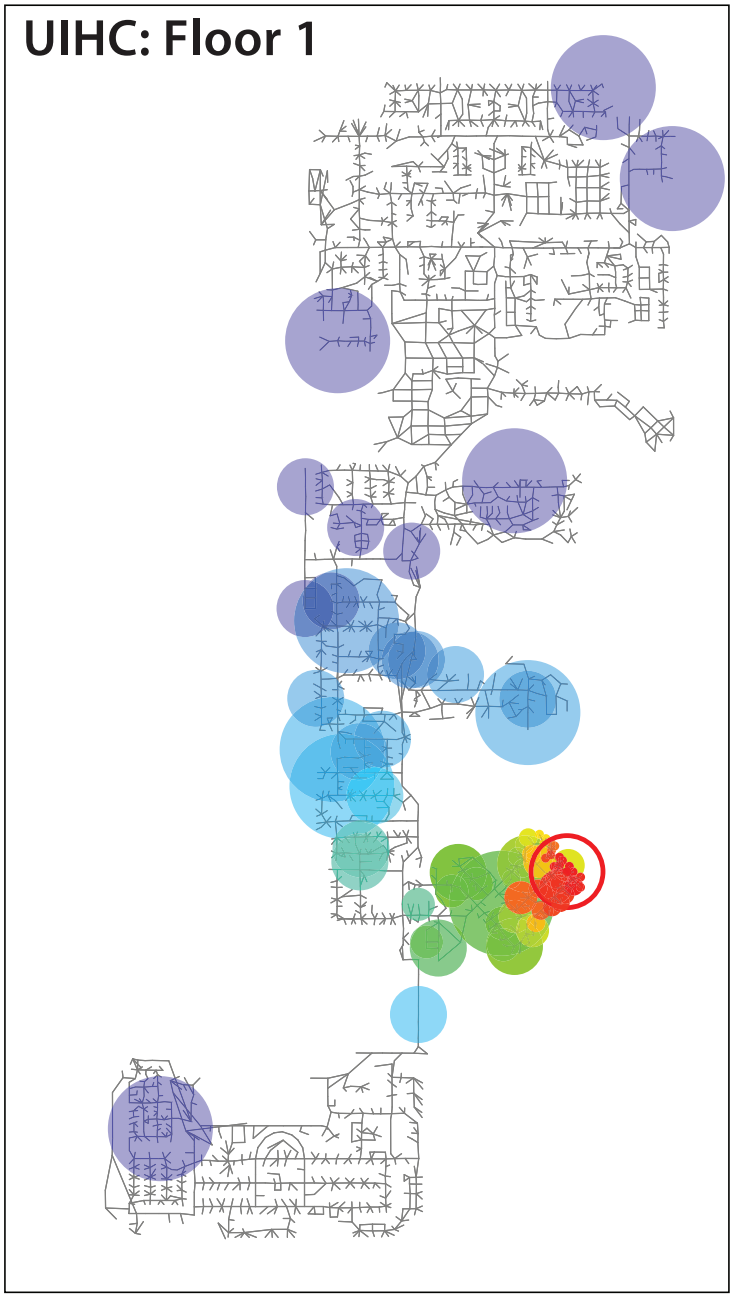

Figure 2.3: This figure shows the quality of fit for the best models located in rooms that the clustering-based heuristic search considers as potential centers for a Physician's Nurse Assistant. The larger discs indicate evaluation of a cluster rather than a single room. The best overall center considered by the algorithm is located in the room at the center of the red circle, and the quality of fit ranges from poor (blue) to the best (red). These models are trained on EMR data from March, 2007. Note that this heuristic does not find the optimal model in this instance.

$MLL_i(c, \beta, \gamma, r)$ and β_v and γ_v are values that maximize $LL_i(v, \beta, \gamma)$.

Proof 1 (Proof of Theorem 1 by contradiction) *Assume that for some v such that $d(c, v) \leq r$:*

$$MLL_i(c, \beta_c, \gamma_c, r) < LL_i(v, \beta_v, \gamma_v)$$

where $\beta_c, \gamma_c, \beta_v$ and γ_v maximum their respective log-likelihoods. Then,

$$MLL_i(c, \beta, \gamma, r) < LL_i(v, \beta, \gamma)$$

where $\beta = \beta_v$ and $\gamma = \gamma_v$.

Since $d(a, b) \geq 1 \quad \forall a, b$ either

$$\sum_{l \in L} log(\beta \cdot (d(c, l) - r)^{-\gamma}) < \sum_{l \in L} log(\beta \cdot d(v, l)^{-\gamma})$$

or

$$\sum_{l \in \overline{L}} log(1 - \beta \cdot (d(c, l) + r)^{-\gamma}) < \sum_{l \in \overline{L}} log(1 - \beta \cdot d(v, l)^{-\gamma})$$

Assume the former is true. That implies for some $l \in L$ $d(c, l) - r > d(v, l)$. This in turn implies that $max(1, d(c, l) - r) > max(1, d(v, l))$ which implies that

$$d(c, l) - r > d(v, l)$$

However, since $d(c, v) \leq r$, and $d(c, l) \leq d(c, v) + d(v, l)$:

$$d(c, l) - d(v, l) \leq d(c, v) \leq r \qquad d(c, l) - d(v, l) > r$$

This is a contradiction. Assuming the latter inequality is true instead leads to a similar contradiction. Therefore the original assumption does not hold. MCLL upper-bounds the log-likelihood of any room in a cluster.

Theorem 1 allows us to use a branch-and-bound approach to process the hierarchy of clusters while pruning away clusters that are guaranteed not to contain a "good enough" center. In a typical step, we process a cluster C somewhere in the cluster tree. Suppose that C has cluster center c and radius r. We compute the maximum value of $MLL_i(c, \beta, \gamma, r)$ over values of β and γ by making a call to the Nelder-Mead algorithm. If this value is no greater than the current best log-likelihood seen by the algorithm, we can stop searching this cluster immediately. Otherwise, we compute the maximum value of $LL_i(c, \beta, \gamma)$ and mark the nodes in the cluster for further processing.

Once the c, β, and γ values have been found for all HCWs, the attractiveness parameter α_v can be calculated for each node v in the hospital. While there are a number of ways to calculate attractiveness for rooms (e.g., based on whether the room is an office, or occupied patient bedroom), to calculate the attractiveness value we first calculate the total number of seconds of login activity in each room that has an EMR terminal. Let M be the set of all the rooms with an EMR terminal in the hospital and for each $u \in M$ let T_u be the total number of seconds any HCW is logged into room u. Note that there are many rooms in the hospital with no EMR terminal, and thus M is a proper subset of the set of all rooms in the hospital. For each room v in the hospital we then calculate a smoothed activity level T'_v which reduces the effect of outliers and assigns activity values to rooms with no EMR terminals. For each room v in the hospital we set

$$T'_v = s_v \sum_{u \in M} T_u \cdot d(u, v)^{-\delta}$$

Algorithm	min calls	mean calls	max calls
Brute Force	18961	18961	18961
Heuristic	222	675	1172
Improved	89	658	6185

Table 2.2: The brute-force search uses far more calls to Nelder-Mead than the heuristic or the improved algorithm.

The heuristic actually uses slightly more calls to Nelder-Mead on average than our improved algorithm, though our improved algorithm is not as consistent.

where s_v is a scaling factor chosen so that $s_v \cdot \sum_u d(u,v)^{-\delta} = 1$. δ is a constant that sets the level of decay on the smoothing. For our experiments we found $\delta = 6$ resulted in minimal amounts of activity spilling over from areas of the hospital known to be in use at a particular time to areas known to be closed. The attractiveness value for each room v can then be found by solving for α_v in the following equation

$$\alpha_v \cdot \sum_i \beta_i \cdot d(c_i, v)^{-\gamma_i} = T'_v \tag{2.7}$$

where the summation is over all HCWs i. Figure 2.2 shows the number of seconds of login in each room during one shift in the month of February, 2007 and the smoothed attractiveness values.

Figure 2.5 shows the log-likelihoods for the rooms searched by our improved algorithm when finding the best center for our example HCW. In this particular case, the improved algorithm has to search more rooms than the heuristic, but, unlike the heuristic, it returns the optimal solution in this and every instance. As shown in Table 2.2, the heuristic and improved algorithm both require far fewer calls to Nelder-Mead

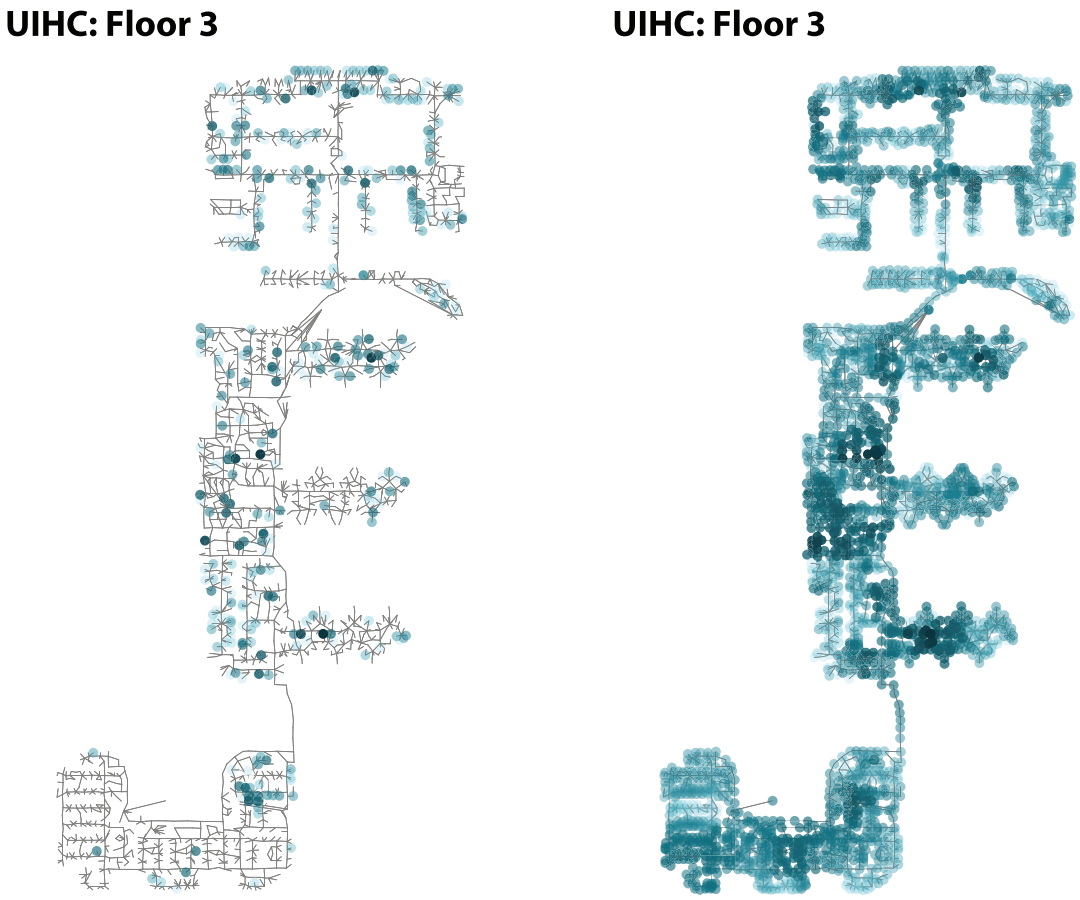

Figure 2.4: These graphics show a comparison between the observed login data and attractiveness in each room on the 3rd floor of UIHC. The graphic on the left shows the number of seconds of HCW logins in each room, with darker dots indicating more activity. The graphic on the right shows the attractiveness of each room after smoothing is performed. Smoothed attractiveness values range from 0.4 to $1,682,801$ with a mean of $65,510$ and a median of $33,278$. Rooms with attractiveness values near 0 may have terminals that are not used for accessing patient data, and rooms with attractiveness values over 1 million are likely to contain one or more HCWs at any given time. It seems reasonable that some areas (e.g., patient care areas) have a lot of activity, while many or most rooms (offices, closets, mechanical equipment rooms, etc.) are actually empty much of the day.

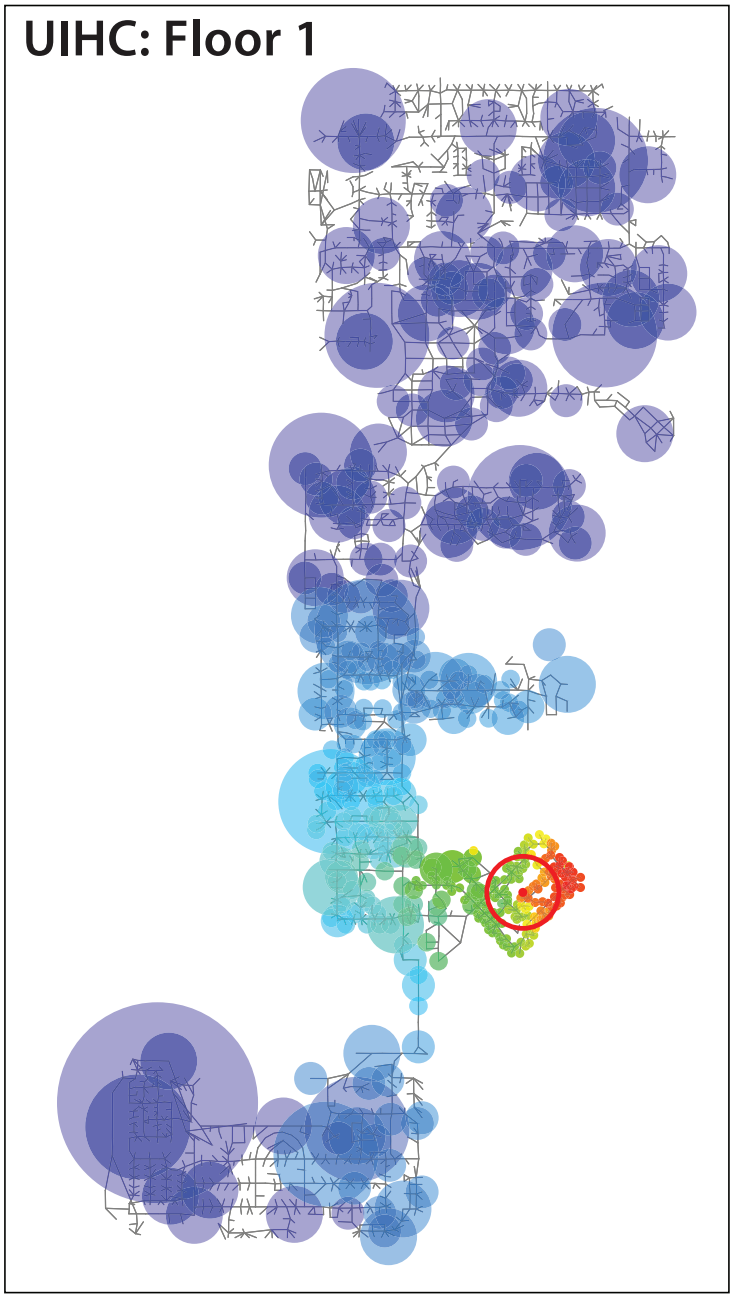

Figure 2.5: This figure shows the quality of fit for models searched by our improved algorithm. There is a circle in each room considered as a candidate center by our algorithm, with the best overall center located in the room at the center of the red circle. The quality of fit ranges from poor (blue) to the best (red). These models are based on EMR data from March, 2007. Note that this algorithm does not search all the rooms in the facility, but still zeros in on the optimal model.

than the brute-force search algorithm. While the improved algorithm uses slightly fewer calls on average, there are cases where it uses significantly more calls than the heuristic.

2.5 Multi-center Extension

While these single-center models do a reasonable job of fitting the observed data, some job types may be better modeled as having multiple centers, each of which makes some contribution to a healthcare worker's stationary probability distribution. Consider a physician who spends some time in a departmental office, and some time on an inpatient unit. A single-center model would have to attempt to find one center location and set of decay parameters that accounts for activity in both locations.

2.5.1 Motivation for Multi-center Models

In Section 2.5.3.2 we will discuss one way to decide which HCWs are best modeled with single-center models and which are best modeled with multi-center models. Before we present our assumptions and model for the multi-center problem, we present some visualizations of EMR activity for several healthcare workers that supports the intuition that there are some individuals or job types in the hospital better modeled by multi-center models. Keep in mind when examining the figures in this section that the HCWs whose EMR activity are being visualized are not necessarily typical users. A sample of HCWs' activities were checked during a one month period for "interestingness".

First, consider Figures 2.6-2.9. For each of the HCWs shown, all EMR activity

occurs on a single floor, typically within a visually apparent cluster. Figures 2.6 and 2.7 show HCWs that seem very well suited to a single-center model, while Figures 2.8 and 2.9 show HCWs for which a single-center model is probably appropriate, though the observed activity patterns may be a little more difficult for a single-center model to fit well. In all four cases, activity is tightly clustered.

Contrast these examples with Figures 2.10 and 2.11, where multi-center models seem more appropriate. In each case, there are two clusters of observed EMR activity each in different units, or on different floors.

2.5.2 Multi-center Models

Note that some of the previous work on identifying spatial distributions of wildlife, people, and search engine queries did have extensions to multi-center models. Of particular interest, Backstrom et al. [7] considered that some search engine queries could have multiple centers of interest. For example, queries for "Delta Airlines" might be expected to cluster around Delta's hub cities. Griffith also discusses multi-center models of population density in his work in modeling urban populations [47].

The extension of multiple centers to models of HCWs requires deciding how each of the centers should influence activity in each room of the hospital. We focus two simple options, though other options (e.g., weighting contribution from each center) exist. (1) *disjoint* multi-center models, in which activity in each room is the maximum of the activities the various centers assign to that room. (2) *additive* multi-center models, in which activity in each room is determined by summing the activity

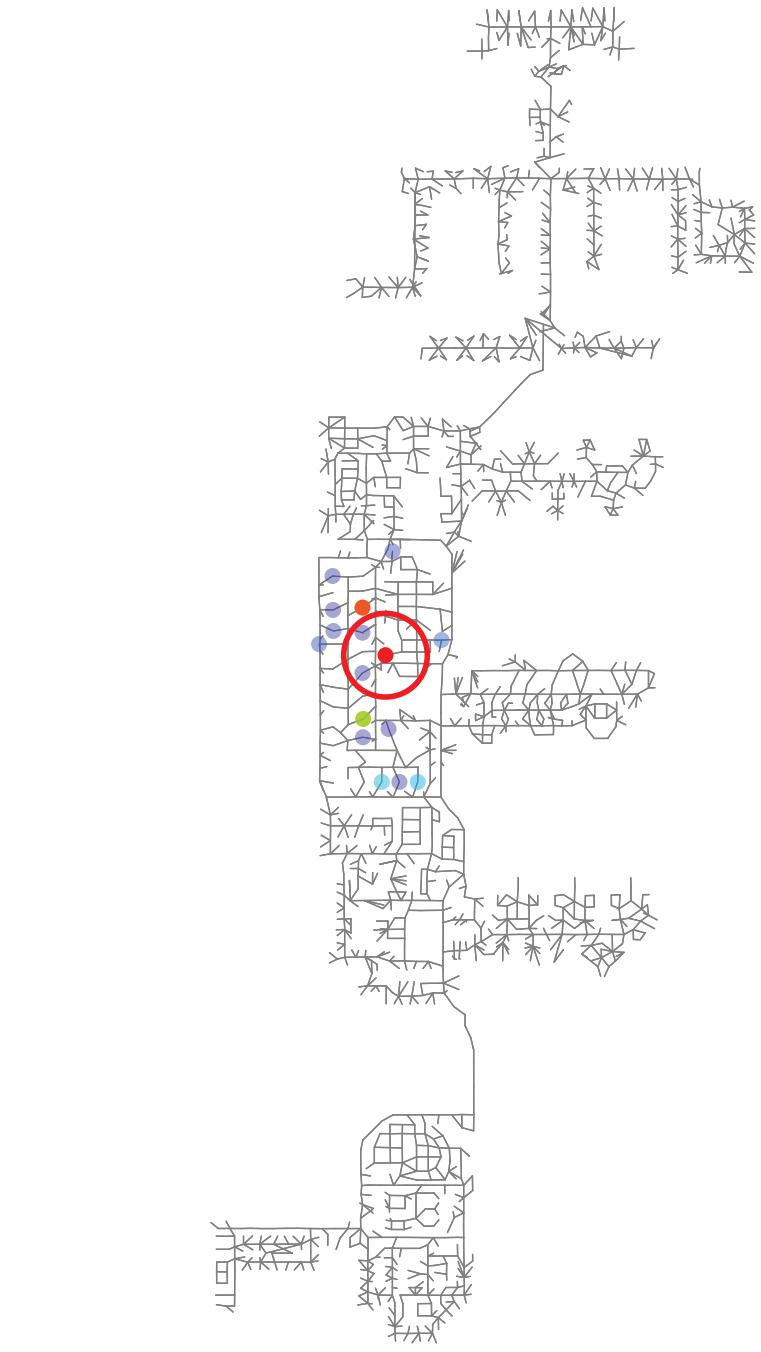

Figure 2.6: A Staff Nurse II with EMR activity restricted to a unit on the 5th floor of UIHC.

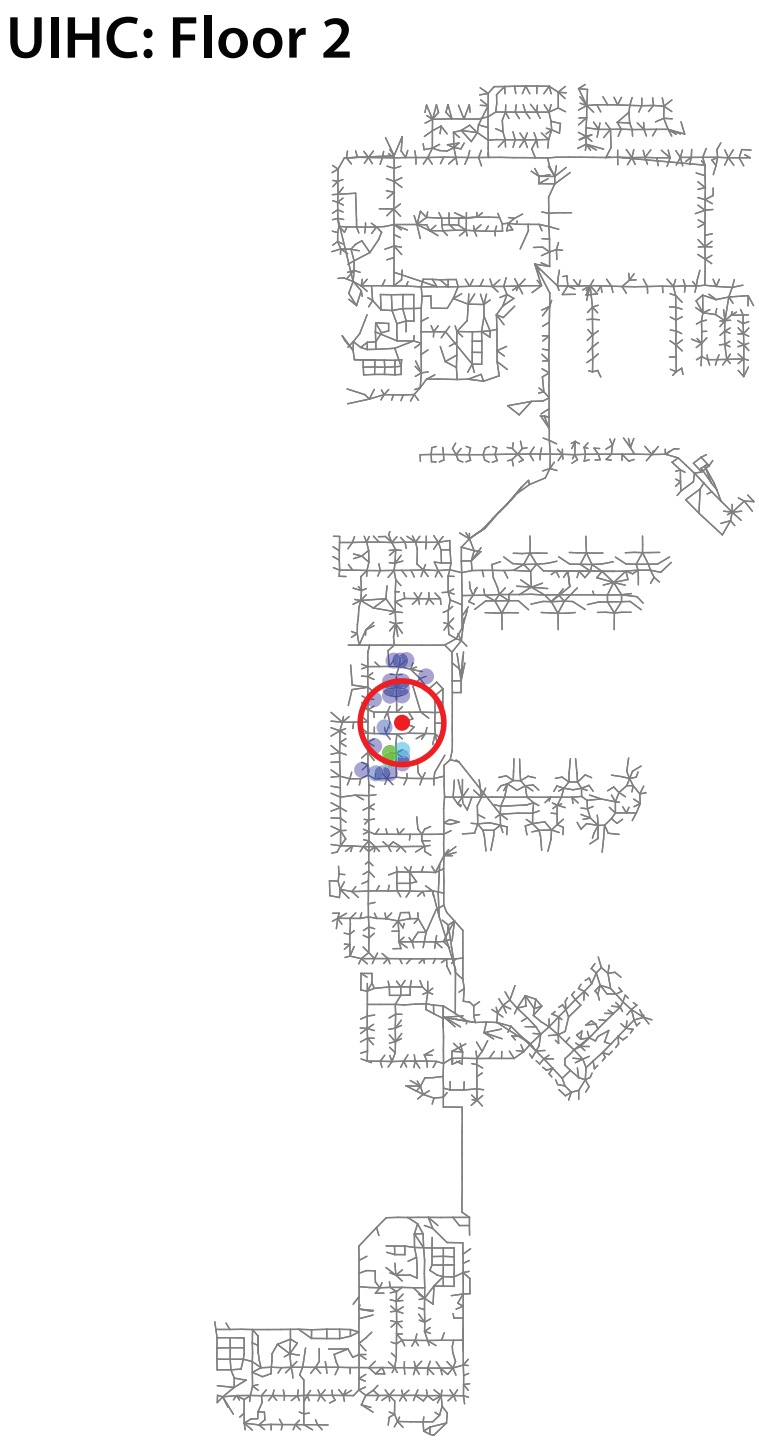

Figure 2.7: A Nurse Assistant with EMR activity restricted to a unit on the 2nd floor of UIHC.

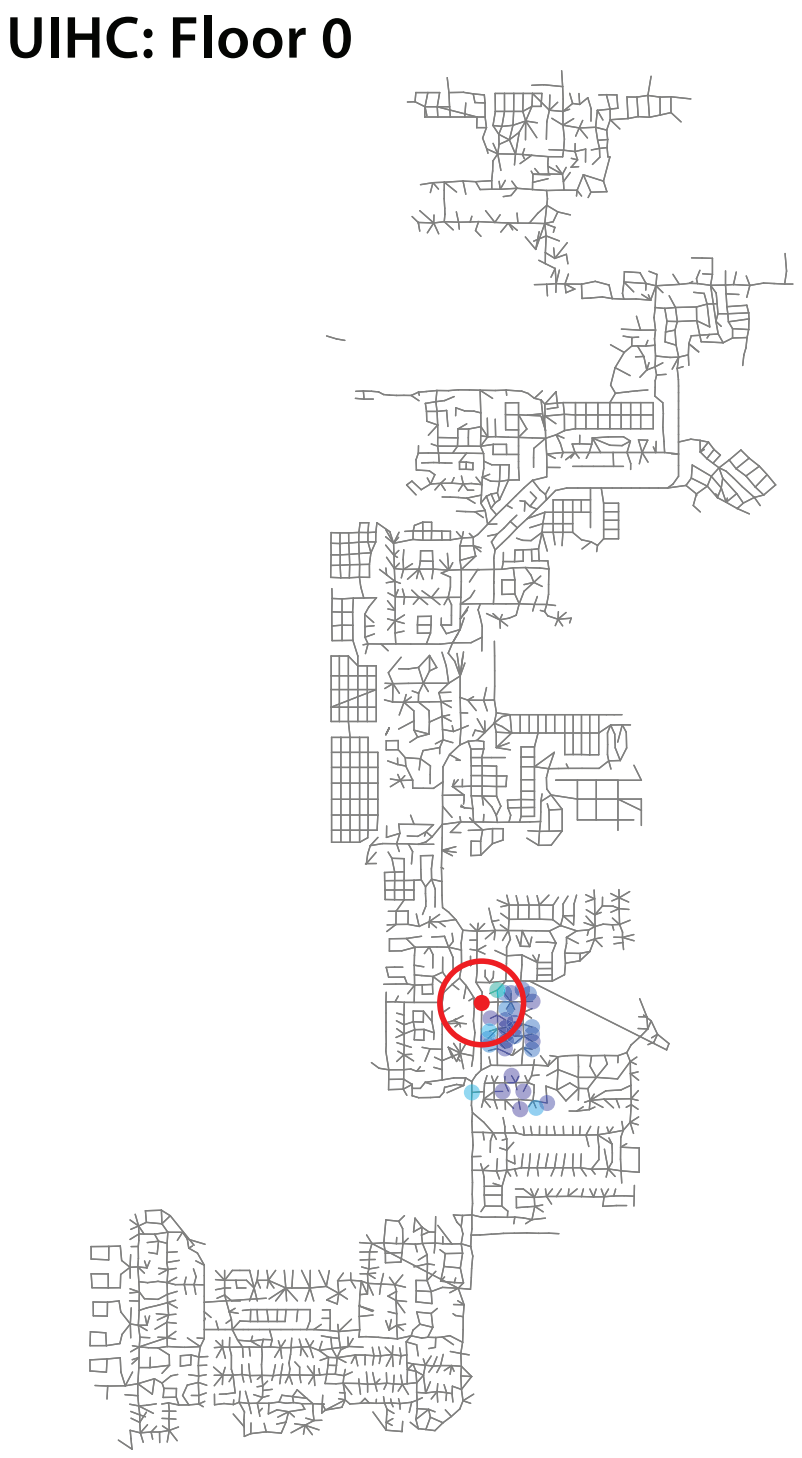

Figure 2.8: A Staff Nurse II with EMR activity restricted to a unit on the subground floor of UIHC.

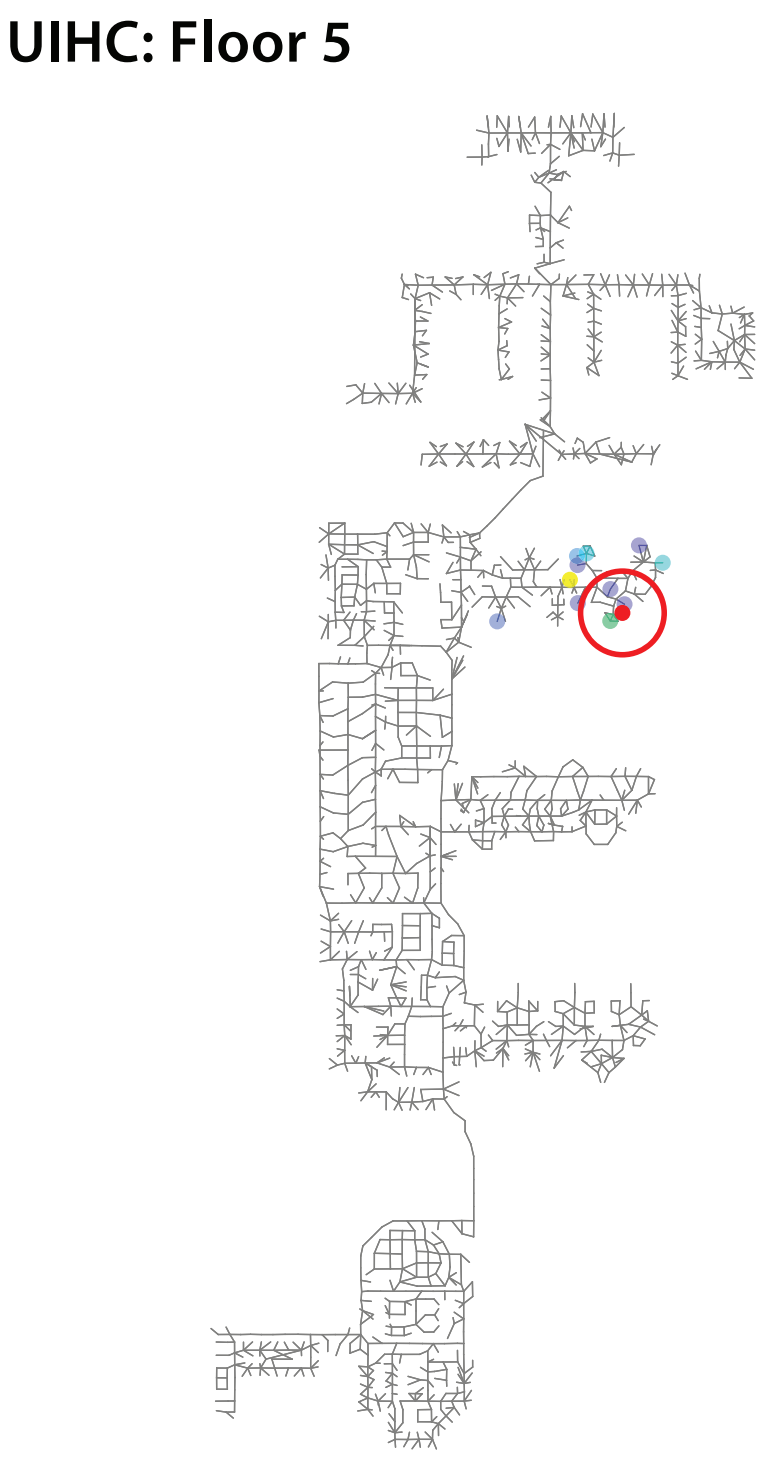

Figure 2.9: A Staff Nurse II with EMR activity restricted to a unit on the 5th floor of UIHC.

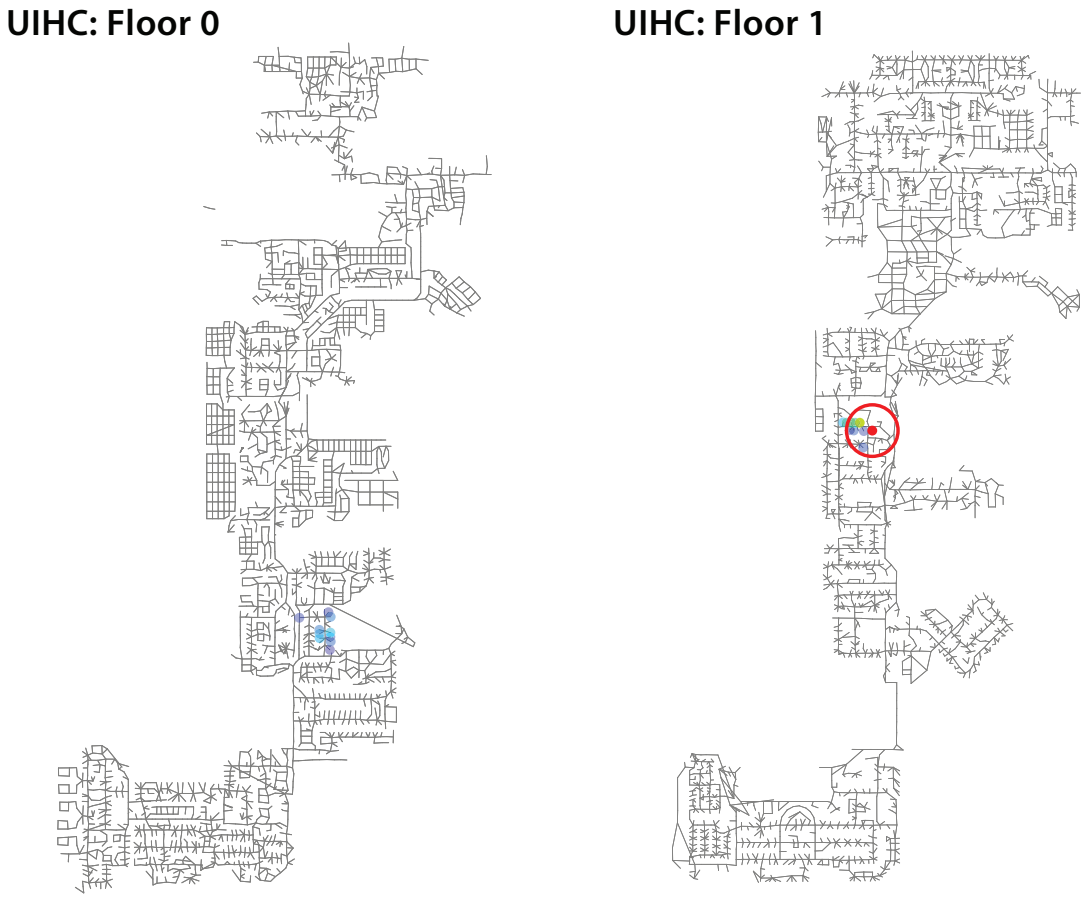

Figure 2.10: An Advance Reg. Nurse Practitioner with EMR activity on the lower two levels of the UIHC facility. Note that when forced to use a single-center model, the algorithm picks a center in the cluster on the ground floor. This suggests that generated activity would be far less than expected in the cluster on the other floor and/or generate more activity than expected in rooms along the shortest path between the two clusters. A multi-center model would be able to generate activity in both clusters without generating any unwanted activity between the two clusters.

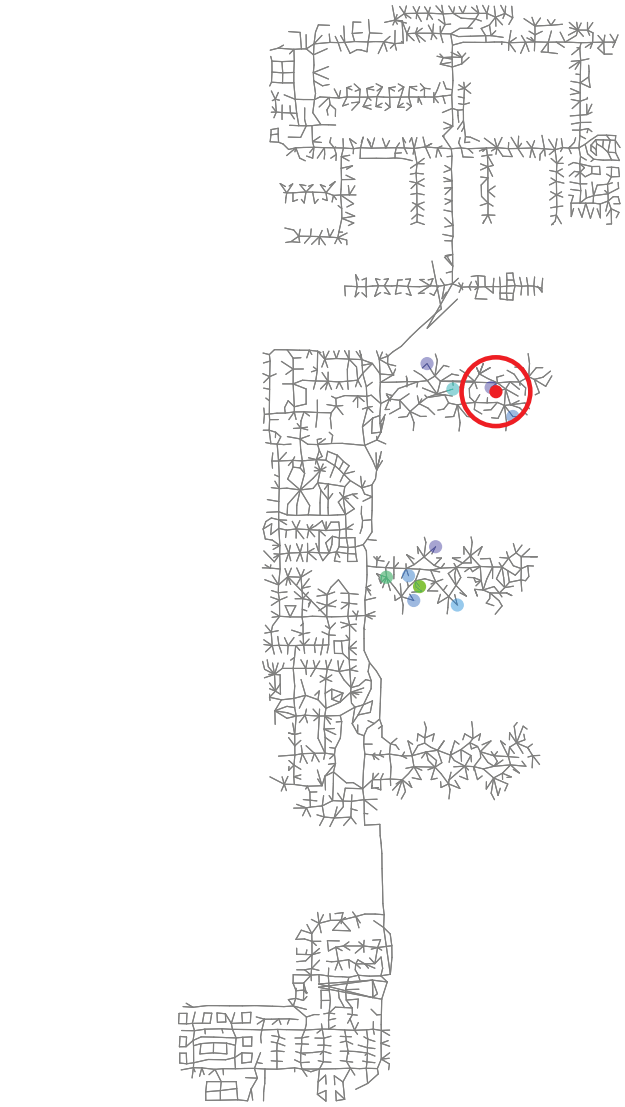

Figure 2.11: A HCW with an unknown job title has EMR activity only on one floor, but it appears in distinct clusters in two different units. A single-center model for this HCW would be usable, but a multi-center model would do a better job of fitting the observed data. Note that this example is also illustrative of one of the benefits of using distances on top of a graph-theoretic model rather than just using Euclidean distance: the realistic walking distance (shortest weighted path distance) between some points in the two clusters is over 50% longer than the straight-line distance.

generated by all centers.

More formally, Let C_i be a set of centers of the form $< c_j, \beta_j, \gamma_j >$ for HCW i. Then, for the class of disjoint multi-center models, we assume the probability that an HCW i picked from some room v is proportional to

$$p_{disj} = \max_{<c_j,\beta_j,\gamma_j>\in C_i} \beta_j \cdot d(c_j, v)^{-\gamma_j}. \tag{2.8}$$

In additive multi-center models, the probability that an HCW i picked from some room v is proportional to

$$p_{add} = \sum_{<c_j,\beta_j,\gamma_j>\in C_i} \beta_j \cdot d(c_j, v)^{-\gamma_j}. \tag{2.9}$$

The two classes of models are very similar in general, but it is worth considering where they differ. For a 2-center model, imagine a topological map with concentric rings around each center for each discrete value of the decay function for that center. Consider a room at the intersection of two identically labeled rings. Without loss of generality, assume that the rings only intersect in one room. In the additive model, that room will have a higher decay function value than any other room on either of the two rings, meaning that the modeled probability that a HCW chosen in this room is HCW i is greater than in any other room on either ring. In the disjoint model, however, the modeled probability is the same everywhere on the two rings.

The previous work on search engine queries uses disjoint multi-center models [7], while the work on urban population density uses additive multi-center models [47]. In our own work, we choose to interpret models as being additive. It seems natural and likely that a HCW would have somewhat increased activity levels in the

area geographically between two clusters of activity, whether or not such increased activity shows up in our sample of location data.

2.5.2.1 Updated Equations

Keep in mind that it is quite possible that the same model will do a similarly good job fitting the observed data when interpreted as a disjoint and an additive model. While we focus on the additive class of models, it is generally straightforward to transform the additive model description, various probability functions, and algorithms into their disjoint analogs.

The spatial probability distribution of a HCW i over all the UIHC facility rooms is obtained by normalizing the values

$$\{\alpha_v \cdot \sum_{<c_j,\beta_j,\gamma_j>\in C_i} \beta_j \cdot d(c_j, v)^{-\gamma_j} \mid v \text{ is a node in the hospital graph}\}. \tag{2.10}$$

Let s_i be the normalizing factor for HCW i, i.e., s_i is a value such that

$$s_i \cdot \sum_v \alpha_v \sum_{<c_j,\beta_j,\gamma_j>\in C_i} \beta_j \cdot d(c_j, v)^{-\gamma_j} = 1. \tag{2.11}$$

Given these spatial distributions for the HCWs, the expected number of HCWs in a room v at any time instant is

$$\alpha_v \sum_i s_i \cdot \sum_{<c_j,\beta_j,\gamma_j>\in C_i} \beta_j \cdot d(c_j, v)^{-\gamma_j}, \tag{2.12}$$

where the outer sum is over all HCWs i. Then the log of the likelihood that the EMR login records are obtained by sampling from HCW i's assumed distribution is:

$$MCLL_a := \sum_{l \in S} \Big[log\big(\sum_{<c_j,\beta_j,\gamma_j> \in C_i} \beta_j \cdot d(c_j, r_l)^{-\gamma_j} \big) \Big] \tag{2.13}$$

$$+ \sum_{l \notin S} \Big[log\big(1 - \sum_{<c_j,\beta_j,\gamma_j> \in C_i} \beta_j \cdot d(c_j, r_l)^{-\gamma_j} \big) \Big]$$

2.5.3 Multi-center Algorithms

While Backstrom et al. proposed using a heuristic to find optimal single-center models for search engine queries, we presented an algorithm in Section 2.4 that is both fast in practice and guaranteed to return the optimal single-center model. Unfortunately, the solution space for multi-center models is much more complex. Recall that using gradient ascent to find the best β, γ pair for a fixed HCW and center location relied upon the fact that the single-center log-likelihood function (Equation 2.5) is unimodal (i.e., a unique local maximum) for polynomial decay functions.

We found that the log-likelihood functions for both the additive and disjoint class of multi-center models are multimodal (i.e., they have multiple local maxima) even in fairly trivial examples, and thus even if the placement of the centers in the optimal multi-center model is known *a priori* we cannot use gradient ascent to find the globally maximal decay parameters. Empirical verification was done by fixing two centers for a HCW and running the Nelder-Mead Simplex algorithm with observed EMR activity repeatedly with different initial values for the decay parameters. While, as expected, the same (optimal) solution is returned every time for single-center models, with multi-center models this procedure returns a variety of very different solutions, even in the 2-center case. The procedure was repeated over a number of

HCWs, and to minimize the possibility of software bugs, we used a standard library version of Nelder-Mead [69] rather than the more efficient custom C implementation that we use in practice.

In addition to this empirical experiment, we present an analytic proof of multimodality for the additive class of models. (1) We model a simple five room facility like the one pictured in Figure 2.1. (2) We place two HCWs in our facility, the first having two logins, and the second having one. (3) We pick two centers for the first HCW and find the log-likelihood function for that instance. (4) We find the Hessian of the log-likelihood function, and the eigenvector of the matrix. (5) We fix three of the four decay parameters, and run a local search to maximize one of the eigenvalues. (6) We show that one of the eigenvalues is positive for some choice of decay parameters. This implies that the Hessian is not negative semi-definite, which in turn implies that the log-likelihood function is non-concave. While our full proof is too long to reproduce in its entirety, Appendix 5.1.0.4.1 lists the Mathematica commands necessary to reproduce our results.

As this empirical and analytic work shows, it is impossible to guarantee that in general Nelder-Mead will find the best decay parameters for any pair of centers even when given the center locations *a priori*. While it would be possible to try all combinations of β and γ to some arbitrary level of precision and use the best values, this becomes exponentially more expensive as required numerical precision is increased. We thus resort to heuristics to identify good multi-center models.

We consider three heuristics for finding good multi-center models for HCWs,

input : HCW i, number of centers k, *HospitalGraph*, and location data *Loc*
output: A multi-center model for HCW i

```
1  bestCntrs ←RandCntrs(k);
2  for iter ← 1 to MaxIter do
3      cntrs ←RandCntrs(k);
4      lastCntrs ← ∅;
5      while lastCntrs ! = cntrs do
6          lastCntrs ←cntrs;
7          // Reoptimize each center to cover entire facility
8          for j ← 1 to k do
9              cntrs[j]←NelderMead(HospitalGraph, Loc, cntrs [j]);
10             roomProbs[j]←ProbByRoom(HospitalGraph, cntrs[j]);
11         end
12         // Redo the cluster assignments
13         reset cluster assignments;
14         for u ∈ HospitalGraph do
15             j ←argmax(roomProbs[]);
16             append u to cluster[j];
17         end
18         // Reoptmize each center to cover its cluster
19         for j ← 1 to k do
20             cntrs[j]←ReoptimizeInCluster(cntrs[j], cluster[j], Loc);
21         end
22         if MultEval(cntrs) > MultEval(bestCntrs) then
23             bestCntrs ←cntrs;
24         end
25     end
26 end
27 return bestCntrs;
```

Algorithm 1: k-clustering heuristic: see text for more details

based on the general idea of reducing multi-center problems to a number of 1-center problems for which we have efficient algorithms. The first heuristic is based on the discussion in Backstrom et al. [7]. We make the assumption of disjoint influence only for model training, and interpret the model as additive during the generation phase. This assumption lends itself to the k-clustering based heuristic seen in Algorithm 1. The function RandCntrs(k) returns a list of k centers with random locations and arbitrary β and γ values, while the NelderMead() call uses Nelder-Mead to find the optimal β and γ values for a center c located in a specified room. The ProbByRoom(Graph, cntr) function takes our hospital graph and a single-center model and returns the probability that a HCW picked from each room is the HCW under consideration. The ReoptimizeInCluster(cntr, subgraph, logins) function solves the single-center subproblem based in the cluster rather than the entire facility, while the MultEval(cntrs) function returns the log-likelihood of a given multi-center model.

The algorithm repeatedly independently optimizes a set of centers based on observed activity in the entire facility as if each was the only center, partitions the hospital based on which center assigns the highest probability to each room, and chooses a new center within each cluster based only on observed activity within that cluster. The best solution seen after repeating this many times is chosen as the model.

Next, we consider the greedy heuristic shown in Algorithm 2, which uses a single-center algorithm (our optimal algorithm) as a subroutine. The idea is that initially when finding centers for some HCW, all login data are unaccounted for. As centers are added to the model, they will generate some activity in rooms for which

input : HCW i, stopping criteria *StopCrit*, *HospitalGraph*, and login data *Loc*
output: A multi-center model for HCW i

```
1  // Initially, all observed logins are unaccounted for
2  // U is the set of unaccounted for logins
3  U ←Copy(Logins);
4  cntrs ← ∅;
5  while not CriteriaMet(StopCrit) do
6  |  // Use a single-center algorithm as a subroutine
7  |  // to find the best center for just the
8  |  // logins that are unaccounted for
9  |  center ←GetSingleCenter(HospitalGraph,U);
10 |  append center to cntrs ;
11 |  T ←ExpectedTimeByRoom(HospitalGraph,U,center);
12 |  // Subtract logins accounted for by this new center
13 |  U ←RemoveLogins(U,T);
14 end
15 // Verify cntrs is locally maximal
16 // This could be done after each iteration, but is
      expensive
17 NelderMead(HospitalGraph,Logins,cntrs);
18 return cntrs;
```

Algorithm 2: greedy heuristic

the HCW under consideration has logins. Such activity "accounts" for some of the unaccounted for login data, and an appropriate amount of time is subtracted away from the unaccounted for data. In our implementation of ExpectedTimeByRoom(Graph, logins, cntr), we calculate the probability that a HCW picked at random from each room is the HCW under consideration, and multiple this value by the attractiveness of each of those rooms giving an estimate of the amount of time the HCW will spend in that room. The RemoveLogins(logins, expectedTime) function simply sees if there are any unaccounted for logins remaining that an element of expectedTime accounts for and removes those logins if so. We repeatedly find the best single-center model to account for the remaining login data, add that to the list of centers, and remove login data that is accounted for by that new center.

Although we showed earlier that gradient ascent methods are not guaranteed to find the globally optimal model, we do call Nelder-Mead immediately before returning a model to see if changes to decay parameters might yield a slightly better model (i.e., we seek a nearby local optima). Note that the stopping criteria for this heuristic can be set such that it will stop after k centers have been found, or it can use other stopping criteria to choose an appropriate k for each HCW individually. For example, the algorithm could terminate after the improvement in fit for the model after adding another center falls below some threshold, or after some percentage of the HCW's activity is accounted for. In practice, we use a mix of fixed-k models and free-k models where k is allowed to increase until 95% of activity has been accounted for, or k increases to the number of unique rooms HCW i logs into in the observed data.

input : HCW i, stopping criteria *StopCrit*, *HospitalGraph*, and login data *Loc*
output: A multi-center model for HCW i

```
1  // Initially, all login data are unaccounted for
2  U ←Copy(Loc);
3  cntrs ← ∅;
4  while not CriteriaMet(StopCrit) do
5      if cntrs contains 2 or more centers then
6          for j ← 1 to Length(cntrs) do
7              // Remove jth center from the list of centers
8              RemoveCenter(cntrs, j);
9              // Calculate activity unaccounted for by the new
                  list of centers
10             U ←RecalcUnaccounted(Loc, cntrs);
11             center ←GetSingleCenter(HospitalGraph,U);
12             // Insert new center at index j
13             InsertCenter(cntrs, j, center);
14         end
15     end
16     center ←GetSingleCenter(HospitalGraph,U);
17     append center to cntrs ;
18     // Subtract logins accounted for by this new center
19     T ←ExpectedTimeByRoom(HospitalGraph,U,center);
20     U ←RemoveLogins(U,T);
21 end
22 // Verify cntrs is locally maximal
23 // This could be done after each iteration, but is
      expensive
24 NelderMead(HospitalGraph,Loc,cntrs);
25 return cntrs;
```

Algorithm 3: replacement heuristic

The fact that the first center is chosen in isolation is a serious limitation of our greedy heuristic (and greedy heuristics in general), which we address in a third heuristic. Consider what happens when the greedy heuristic is placing the kth center (line 5). Despite the fact that the first $k-1$ centers generally account for most of the activity for the HCW, the algorithm only has complete information about the placement of other centers when placing that kth center. This means that the decay parameters and the location for the first $k-1$ centers were placed to account for all of the remaining activity at the expense of accounting for nearby activity. Meanwhile, the later centers will do a much better job of accounting for activity in those other areas anyway. To address this concern, the *replacement heuristic* Algorithm 3 adds an additional step (lines 5-11) which adjusts each of the existing centers before each additional center is added.

There are some options for tuning the replacement heuristic, e.g., when to do the replacement, how many times to repeat the replacement step, placing restrictions on the location of the new center, how unaccounted for activity is updated, etc., but based on comparing the quality of multi-center models for a sample of HCWs this seemed a good compromise between the fit of the models and model training time. One final note, despite the lack of an optimality guarantee, once we have fixed the final set of centers and some reasonable decay parameters, we run the model through Nelder-Mead to see if we are already at a local optimum or if the decay parameters can be adjusted slightly to improve the model.

2.5.3.1 Comparing Heuristics

The replacement heuristic is nearly always better than the greedy heuristic. To see why this is so, consider that at the point that the algorithms diverge, the replacement heuristic is replacing some center with a new center that, together with the other centers, very likely does a better job of fitting the observed data. Otherwise, the algorithm would have just picked the original center again. See Figure 2.12 for a comparison of the greedy heuristic with and without replacement. Note that points on the line indicate that the two heuristics returned a multi-center model with the same log-likelihood of generating the observed data. Points above the line indicate the replacement heuristic outperforms the greedy heuristic. Because of the clear advantages of the replacement heuristic, in the remainder of this section we focus on comparing the k-clustering heuristic with the replacement heuristic.

While the greedy heuristic was initially developed to have something to compare the k-clustering heuristic with, it turns out to be at least 20 times faster in practice. Given this speed improvement, we might prefer the replacement heuristic over the k-clustering heuristic even if the models were of somewhat lower quality. However, Figure 2.13 shows that the replacement heuristic consistently outperforms the k-clustering based heuristic. A paired, one-tailed Wilcoxon rank sum test confirms statistical significance ($p < 2.2e - 16$).

It is worth mentioning that both disjoint and additive multi-center models found by the greedy heuristics are in rare cases worse than single-center models. In the greedy and replacement heuristics, decay parameters for each center are optimized

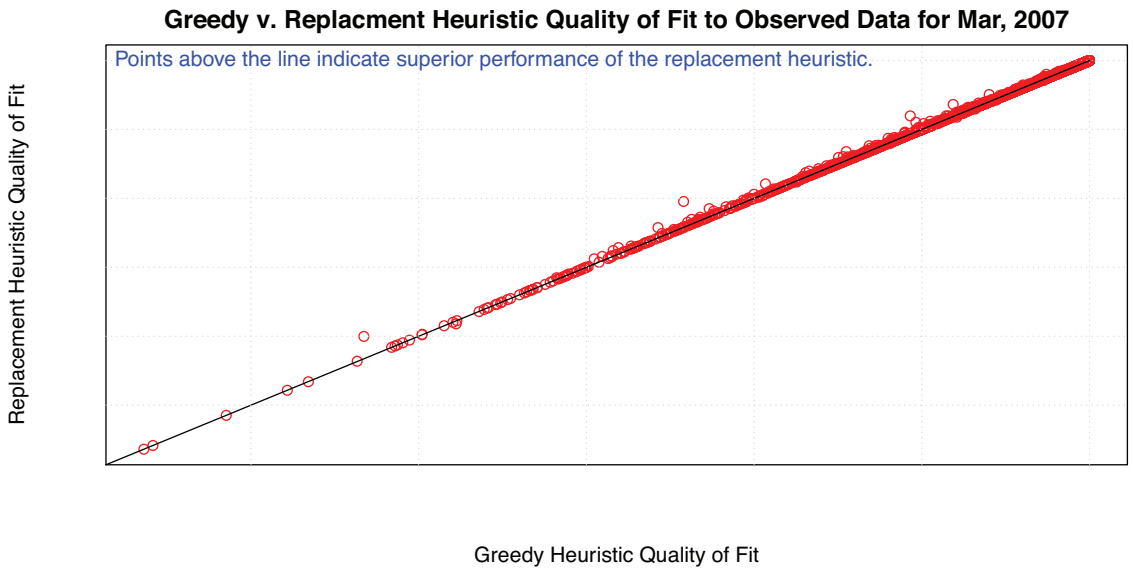

Figure 2.12: A comparison between the greedy heuristic with and without replacement where points above the line show that the replacement heuristic outperforms the greedy heuristic for a particular HCW. There is a point in the plot for each of the 6,078 HCWs that uses the EMR system between 7am and 12pm during the month of March, 2007. Models were trained using data from that same month.

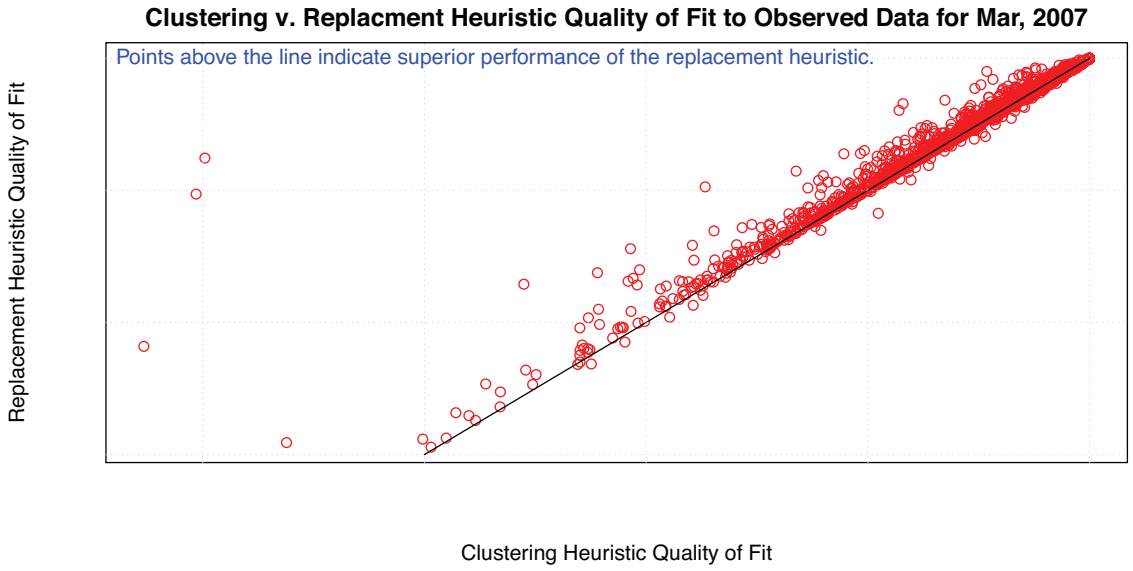

Figure 2.13: A comparison between the replacement heuristic and k-clustering heuristic where points above the line show that the replacement heuristic outperforms the k-clustering heuristic for a particular HCW. There is a point in the plot for each of the 6826 HCWs that access the EMR system any time during the month of March, 2007. Notice that in many cases the points are far about the line, indicating a vastly better model produced by the replacement heuristic for that HCW.

independently prior to returning the final model, and this lack of information sharing between centers is to blame. In the case of disjoint models, a second model is placed assuming that activity it generates will actually result in increased activity for the user. It will generate activity in some rooms where this user should have no activity in order to reach some rooms that in the end won't end up generating activity in anyway. On the other hand, the additive models are adding centers to reach more of the HCWs login areas. While contributions to each room do add up, so do the additional centers' contributions to rooms the HCW does not appear in. Further, even in rooms the HCW does spend a lot of time in, there's a tendency to over-cover rooms. Especially in cases where the HCW spends a lot of time in a small cluster of room, but other HCWs also spend a little time there, over-covering will be severely penalized by the log-likelihood function since p will approach 1. However, in practice these conditions only cause problems when the heuristics pick two centers within a couple hops of each other: indeed nearly all cases where multi-center models are worse than single-center models occur when the two centers are within 2 weighted hops. This occurs infrequently (less than 0.1% of the models), and we take it as an indication that the HCW should just use a single-center model. Note that right before the call to Nelder-Mead at the end of the greedy heuristics, an order of magnitude more $(1 - 4\%)$ multi-center models are actually worse at fitting observed data than their single-center counterparts, which underscores the importance of that step.

2.5.3.2 Identifying Candidates for Multi-center Models

Now that we have identified which heuristic we will use for finding multi-center models for HCWs, we move on to discussing how to decide which HCWs or classes of HCWs should use single-center models and which should use multi-center models.

One option is to always use the replacement heuristic to determine the number of centers for each HCW. Setting the threshold of required EMR activity that must be accounted for to 95% generates single-center models for about 1/3 of HCWs, with the rest having multi-center models. By choosing appropriate stopping criteria, it is possible to make the decision at the individual level without additional work. However, there is a risk of over-fitting the data; in the extreme case e.g., using one center for each room a HCW logs into.

We consider an alternative approach: is it possible to aggregate individuals by their job title and decide at a higher level which groups of HCWs should use single-center or multi-center models? With the help of a physician with knowledge of healthcare operations, we have aggregated the 477 job titles used by our $14,595$ HCWs into 36 job classes (for example, job titles such as Nurse Managers, Staff Nurse I, and Staff Nurse II are all aggregated into the Nurse job class). We then focus on identifying classes of HCWs for which single-center models are sufficient, and which are better suited to multi-center models. We run the static greedy heuristic with $k = 2$ as the stopping criteria. We then compare the fit of the model using just the first center returned with the fit of the model using both centers. We use this approach because we only need to keep one set of models around to do the comparison, and

because we are looking for dramatic improvements in log-likelihood (not something subtle). Figure 2.14 shows the results of this comparison. We group the job classes into three groups. The job classes in the blue circle group have both a high mean and a high median improvement when moving to a 2-center model; these are clear candidates for multi-center models. The job classes in the black diamond group have low means and low median improvements when moving to a 2-center model; HCWs within these classes will be restricted to single-center models. The last group of job classes, the red square group, is ambiguous. A low median and high mean indicates that some individuals see a vast improvement when switching to multi-center models, but most do not. Note that we see the expected job classes in each group: Physician and Resident are in the multi-center group, while Unit Clerk, Administrator, and those working in laboratories get single centers. The group that needs further study includes Therapist, Social Worker and a group categorized as Miscellaneous Patient Care. At this point, we limit our multi-center modeling efforts to 2-center models.

For those job classes requiring further study, we drill down to the level of position. The results, shown in Figure 2.15, indicate fellows and clinical professors should use multi-center models, while secretaries and imaging technicians should use single-center models. While there are still some positions in the ambiguous red square group, because more than half of the members see no improvement moving to multi-center models and the groups are already small, we assign these positions to the single-center group rather than drill down further. Based on this analysis, about 20% of HCWs use multi-center models when doing agenda generation for simulation.

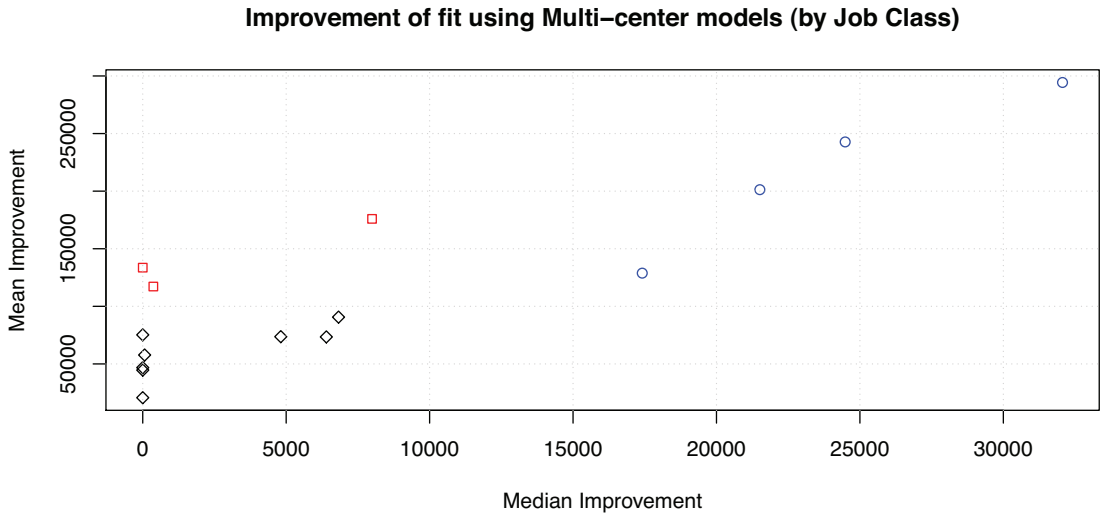

Figure 2.14: The improvement in fitting observed data when switching from single-center to 2-center models aggregated at the job class level. The blue circles are job classes which will use multi-center models, the red squares are job classes which require further review at the position level, and the black diamonds are job classes which will use single-center models. The blue circle group includes job classes such as Physicians, Residents and Pharmacists. The red square group includes the Misc. Patient Care Clerk, Therapists and Social Worker classes. The black diamond group includes Administration, Laboratory, Information Technology and Unit Clerk job classes.

Physician	Center in Office	Center on Clinic / Unit
1	Faculty Office	Main Operating Room
2	Administrative Staff	Hem/Onc Physician Workroom
3	Cardiovascular Diseases	Heart Clinic Physician Workroom
4	Internal Medicine	Rheumatology Clinic Exam Room

Table 2.3: Physicians' centers appear in the expected places.

In many cases where we have complete information about how a room is used, a physician will have a center in a clinic or on an inpatient care unit and one in an office.

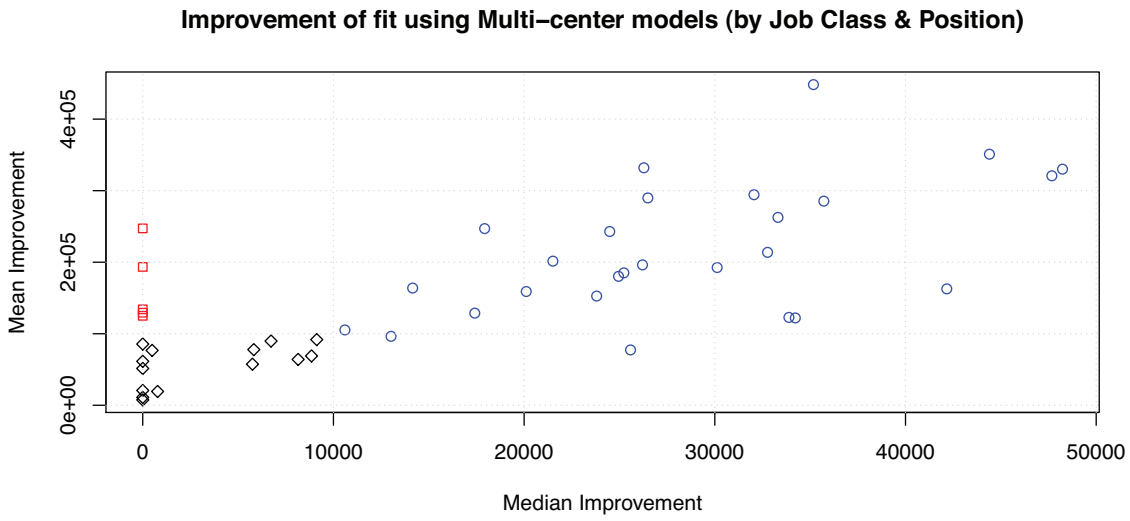

Figure 2.15: The improvement in fitting observed data when switching from single-center to 2-center models including aggregation at the position. Job classes and positions represented by blue circles use multi-center models, and the rest use single-center models. Note that there are a handful of positions which have high means and low medians, but rather than drill down to the individual level we assume single-center models will suffice.

When motivating the use of multi-center models, we pointed in particular to physicians, and we do see what seem to be reasonable pairs of centers for physicians. Figure 2.3 shows a small sample of the roughly 50% of physicians with one center in an office and one in a clinical area. Most of the remaining physicians have two centers in clinical areas, and the rest have either two centers in office area or one center that is not in an office or a clinical area.

2.6 Validation

Recall from Section 1.1.4 that the three suggested mechanisms for validating simulations that we employ are (1) internal validation, (2) checking properties of simulation against known quantities, and (3) comparing simulation properties and

results with the expectations of domain experts.

In order to verify that spatial dispersions estimated by our model match up with real-world HCW behavior, we consulted with a diverse 4 person team of staff and administrators from the UIHC consisting of members from industrial engineering, infection control, general medicine and information technology. This behavioral consulting team is well equipped to describe behavior in the hospital because their work entails a thorough understanding of when and where HCWs are within the facility. We asked the consulting team to tell us about expected HCW behavior along three dimensions: (1) where in the UIHC facility would we find different types of HCWs? (2) which are the most mobile and the most static departments and job categories? and (3) are there particular times of the day or week during which activity would change significantly? In this section we validate our results against what happens in a real hospital setting according to these consultants.

2.6.1 Centers

The Pediatrics department at UIHC consists of specialists that only deal with children and thus the majority of HCWs working in Pediatrics spend almost all of their time in the Pediatrics unit. Figure 2.16 shows the centers locations for HCWs from Pediatrics on the 2nd floor for a typical month. The cluster of dark dots in the middle of the hospital shows that, based on our model, a large number of Pediatric HCWs have centers located there and this is where the Pediatrics unit is located in the hospital. Additionally, the behavioral consultants noted that some members of Pediatric department work in the Neonatal Intensive Care Unit and the Cystic

Fibrosis Unit. Our results confirm that there are a small number (not shown here) of Pediatric HCWs that have their center in the Neonatal Intensive Care Unit or Cystic Fibrosis Unit.

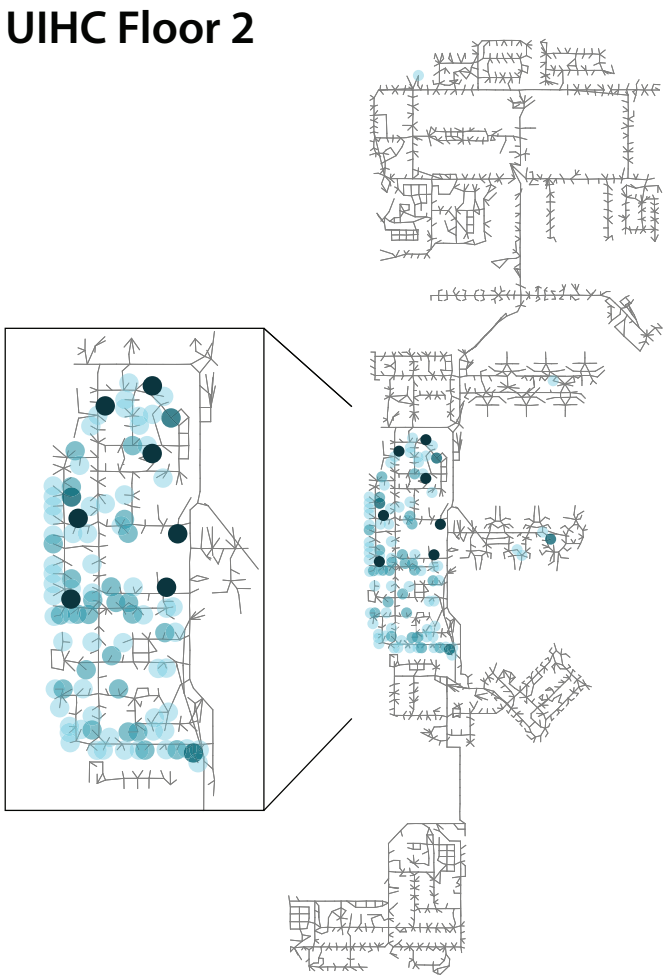

Figure 2.16: Dots mark the centers for HCWs in the Pediatrics department at the UIHC (2nd floor) based on EMR data from March, 2007, with darker dots indicating multiple centers for different HCWs at one location.

2.6.2 Dispersions

To validate the dispersions of HCWs in our models our behavior consultants identified the most and least mobile types of HCWs. Among the most mobile job types are House Staff (Residents), Respiratory Therapists, and Dietitians. CT Service Techs, Unit Clerks, and Secretaries were identified as being some of the least mobile job types in the hospital. To capture the notion of mobility, we introduce the concept of a t-radius. The *t-radius* of a healthcare worker i is the minimum r such that

$$\text{Prob}[i \text{ is in } Disk(c_i, r)] \geq t.$$

Here c_i is center of HCW i and $Disk(c_i, r)$ is the set of nodes in the hospital graph within r hops of c_i. Thus, a HCW with a higher t-radius than another HCW moves further from their center more frequently and is more mobile. For $t = 0.8$, Figure 2.4 shows the average t-radii for each job category in the hospital. Notice that CT Service Techs have very low 0.8-radii indicating they spend 80% of their time in only a handful of rooms near their center. On the other hand, House Staff have relatively high 0.8-radii indicating they tend to spend time in many rooms further from their centers. The job categories not shown in Figure 2.4 and not mentioned by our behavior consultants as being very mobile or very static generally lie in the middle of these extremes.

Our behavior consultants noted that even within the same job type, seniority and mobility may be (negatively) correlated. It was observed that senior members tend to move around less and assign tasks requiring going to other parts of the hospital to their junior counterparts. We looked at the difference in t-radii for House Staff.

Position Title	Average t-radius ($t = 0.8$)	Multi-Center
CT Service Tech	1.50	No
Secretary	5.06	No
Unit Clerk	7.20	No
Nurse Manager	11.0	No
Sonographer	13.6	No
Pharmacy Tech	14.0	No
Clinical Lab Scientist	16.5	Some
Professor	20.1	Most
Social Worker	21.2	Most
Dietician	21.4	Yes
Imaging Tech	25.6	Yes
Respiratory Therapist	25.8	Yes
House Staff	30.3	Yes

Table 2.4: Average t-radii with $t = 0.8$ for selected job categories.

The higher the average t-radii, the more mobile the job category tends to be. Note that this data was calculated using single-center models for all HCWs, though we indicate in the Multi-Center column whether this type of HCW uses multi-center models for simulator agenda generation. Note that the ordering of t-radii meshes nicely with our decision to use single vs. multi-center models for each type of HCW. Within some position titles, the multi-center decision is not all-or-none because there are multiple subtypes, e.g., professors can either be clinical or not and also have an academic rank (assistant, associate, etc.).

Job category	Average t-radius ($t = 0.8$)
House Staff I	35.6
House Staff II	31.3
House Staff III	31.8
House Staff IV	25.0
House Staff V	29.6

Table 2.5: Average t-radius for House Staff.

Senior House Staff (IV and V) tend to move around less than the younger House Staff (I and II).

Figure 2.5 shows that while all house staff HCWs are fairly mobile, there is a notable difference in how rank plays a role in the mobility.

As additional validation, we use spatial dispersions to calculate the expected distance on the hospital graph between two HCWs at any given time. Within almost all job types, the expected distance between two HCWs is about 40 with a couple of notable exceptions. Staff Nurse Anesthesiologists have a significantly lower average expected distance of 20, hinting that these job types have a single focus point for activity. Within departments, the expected distance between two HCWs is lower, being in the range of 25-35 weighted hops. The fact that departments cluster together more tightly than job categories is to be expected since departments tend to have one or two units in which their employees will spend the majority of their time. A single job type will normally have HCWs spread across many departments all over the hospital. Notably, Internal Medicine is the most distributed department with a mean expected distance of 43.0, and Nursing department has an expected distance of 37.3. These two departments are large in size and include a majority of House Staff and Staff Nurses, two of the most mobile job categories.

2.6.3 Activity

Many areas in the UIHC are shut down overnight. While in-patient and out-patient units on the fifth floor are both open during the day, at night the out-patient units close and open units generally have reduced staffing levels. Figure 2.17 shows the expected number of people in each room of the hospital's 5th floor from 2am to 3am. For these pictures we calculated values for our centers model considering

only the people who login overnight. You'll notice that there are entire parts of the hospital where no activity is expected during these hours.

During the night there are also significantly fewer HCWs active in the hospital. The HCWs that are active tend to cover more territory, so we would expect that the dispersion values for these healthcare workers would see a significant change. HCWs that login from 2am to 3am have a t-radii of 16.4 compared to a t-radii of 4.8 for those HCWs that login from 10am to 11am, indicating HCWs logged in at night do move around more.

2.6.4 Internal Consistency

We expect that centers and dispersions (γ values) should be stable over time. Figure 2.6 shows the median change in center location and γ-values for each pair of consecutive months in 2007. Indeed, centers move very little from one month to the next, and γ-values are stable as well. Our behavior consultants told us that members of House Staff I are reassigned every month and this group of HCWs exhibited the largest change in center location from month to month, as can be seen in Figure 2.7.

2.6.5 Empirical Validation

In [96] we manually "shadowed" 148 unique HCWs representing 15 job classes for a total of 606 hours to acquire data on their contacts. The limitation of this attempt is of course that our sample size is necessarily tiny (6,654 contacts) because this approach is so labor intensive. However, contacts within each job type for which HCWs were "shadowed" and for which EMR access occurs were strikingly similar.

We looked at the contacts for each job class that appears in both the shadow-

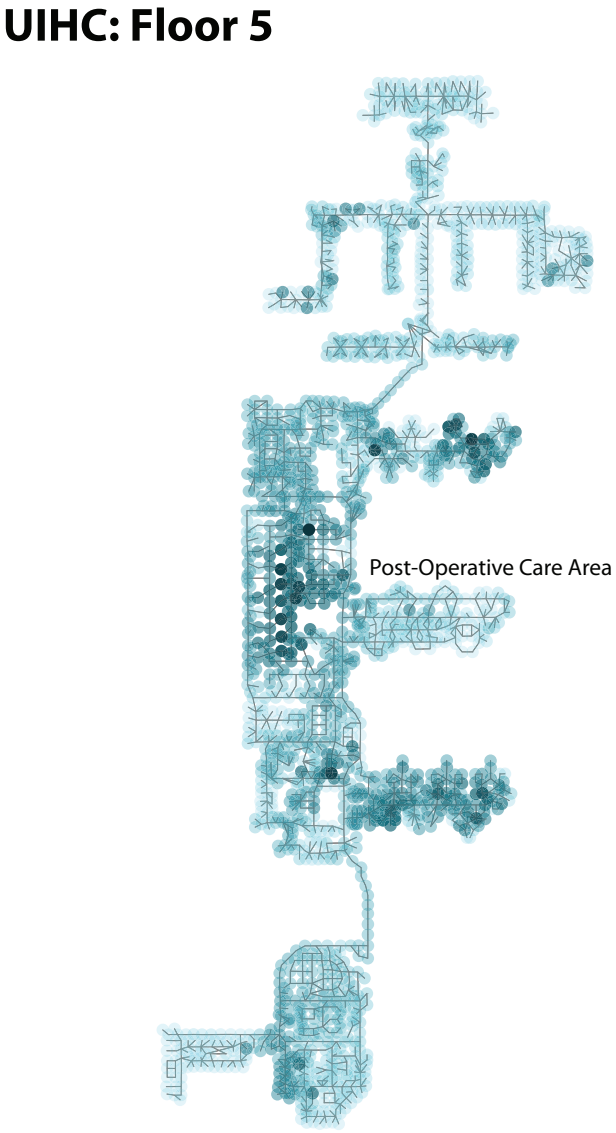

Figure 2.17: Activity generated by single-centers models of UIHC HCWs trained using all EMR activity between 2 a.m. and 3 a.m over our 22 months of data show, as expected, relatively little activity in the out-patient units in the topmost building. The post-operative care unit and recovery rooms located in the middle pavilion also sees a reduction in activity during this time frame since surgeries performed at night tend to be emergencies and patients will initially recover in an intensive care unit.

Months	Center movement	γ movement
Jan-Feb	5.0	0.728
Feb-Mar	1.0	0.472
Mar-Apr	1.0	0.483
Apr-May	1.0	0.490
May-Jun	1.0	0.459
Jun-Jul	1.0	0.508
Jul-Aug	1.0	0.489
Aug-Sep	1.0	0.469
Sep-Oct	1.0	0.447
Oct-Nov	3.0	0.619
Nov-Dec	1.0	0.443

Table 2.6: Median difference in centers and γs in consecutive months in 2007.

While a few HCWs have centers in very different parts of the hospital from one month to the next, the vast majority move less than a few weighted hops. γ values also change very little for most people. See Figure 2.7.

Job Class	Center movement	γ movement
Administration	0.0	0.447
Physician	0.0	0.373
Unit Clerk	0.0	0.852
Nurse	2.0	0.468
Resident Physician	17.0	0.426

Table 2.7: Median difference in centers and γ in consecutive months in 2007 for selected job types.

Note that for most HCWs, centers move very little, but House Staff HCWs have centers that move more than most. One possible explanation for this is that House Staff HCWs rotate to different units on a monthly basis as part of their training. Even though physicians and residents are both suited to multiple centers, physicians' centers are much more stable from month to month.

based contact graphs and contact graphs generated from the centers models. For both contact graphs, we noted the top three job classes each class came in contact with in a list of "top neighbors". On average, each class shared 2.2 out of 3 possible classes in their *top neighbors* list. In a few cases, classes from the two graphs shared all 3 possible neighbors, and in one case the lists were exactly the same.

Keep in mind that there are some limitations with comparing the two contact graphs. The job classes in the two different data sets do not map one-to-one, and for roughly 20% of HCWs we have no job title information. Despite these limitations, we do see clear similarities between contact graphs generated from the centers models and these observed contact graphs.

2.7 Applications to Simulation

Knowing the spatial distributions of HCWs provides a natural way for generating HCW movement within the hospital as well as HCW contact networks. Here we describe our approach to generating this movement and these networks and provide a brief analysis. In Chapter 4, we aim to use these networks to evaluate control strategies such as vaccination, quarantining, cohorting, etc. for the mitigation of hospital-acquired infections.

The basic idea is to generate, for each HCW i, a random walk in the hospital metric space whose stationary distribution is the static spatial distribution of the healthcare worker. If a HCW is in room v at time step t, then in time step $t+1$ she either stays in room v or moves to an adjacent room according to the transition probabilities of the random walk. Our goal therefore is to solve for these transition

probabilities, given the target stationary distribution. Fix a person i and let p_v be the probability of HCW i being in room v. Recall that according to our model,

$$p_v = \alpha_v \cdot \beta_i \cdot d(c_i, v)^{-\gamma_i},$$

where c_i is i's center, γ is i's dispersion, β_i is the constant of proportionality, and α_v is room v's relative "attractiveness". Note that $\sum_v p_v = 1$.

Assuming there are n nodes in the giant component of our hospital graph labeled $1, 2, \ldots, n$ let $\mathbf{p}$ be the spatial distribution vector $(p_1, p_2, \ldots, p_n)$ for i. Let $T = (t_{uv})_{n \times n}$ be HCW i's transition matrix with t_{uv} being the probability of i moving from u to v. Note that t_{uv} and t_{vu} may be distinct and if there is no edge between u and v, $t_{uv} = t_{vu} = 0$. We are interested in finding a T such that $\mathbf{p} \cdot T = \mathbf{p}$ and the following additional constraints are satisfied. Here $N[u]$ denote the closed neighborhood of u, i.e., the neighborhood including u.

1. $\sum_{v \in N[u]} t_{uv} = 1$ for all nodes u in the hospital graph.
2. $t_{uv} \geq 0$ for all nodes u and v in the hospital graph.

The following claim (see for e.g., Chib and Greenberg [20] for a simple proof) leads to a simple algorithm for constructing T.

Lemma 1 *Suppose that for all $u \neq v$, $p_u \cdot t_{uv} = p_v \cdot t_{vu}$ and $\sum_{v \in N[u]} t_{uv} = 1$, then $\mathbf{p} \cdot T = \mathbf{p}$.*

This claim suggests that one way to generate entries in T is to first generate for each corresponding pair of directed edges (u, v) and (v, u) $(u \neq v)$ the values t_{uv} and t_{vu} satisfying the constraint that $p_u \cdot t_{uv} = p_v \cdot t_{vu}$. Once these values are all generated, it is possible that the sum $\sum_{v \in N(u)} t_{uv}$ exceeds 1. If so, we scale down all the t_{uv} values

so that for all u, the sum $\sum_{v \in N(u)} t_{uv}$ is at most 1. Note that this scaling does not cause any violation of the $p_u \cdot t_{uv} = p_v \cdot t_{vu}$ constraint. Finally, for each node u, the residual probability $1 - \sum_{v \in N(u)} t_{uv}$ is assigned to t_{uu}. Clearly, this algorithm ensures that T is nonnegative and that $\sum_{v \in N[u]} t_{uv} = 1$ is satisfied for all u. Furthermore, since the $p_u \cdot t_{uv} = p_v \cdot t_{vu}$ constraint is satisfied, from the above lemma we obtain that $\mathbf{p} \cdot T = \mathbf{p}$. The algorithm is described in Algorithm 4 and we obtain the following lemma via the above argument.

```
   input  : HospitalGraph G
            stationary probability distribution p for some HCW i
   output: A transition matrix for HCW i
 1 initialize transition matrix T ;
 2 for each edge (u, v) ∈ G where u < v do
 3 |   T_uv ←random();        // T_uv can be set to arbitrary value
 4 |   T_vu ← p_u · T_uv/p_v;
 5 end
 6 S ← max_{u∈G} Σ_{v∈N[u]} T_uv;
 7 if S > 1 then
 8 |   for each node u ∈ G do
 9 |   |   T_uv ← T_uv/S;
10 |   end
11 end
12 for each node u ∈ G do
13 |   T_uu ← 1 − Σ_{v∈N[u]} T_uv;
14 end
15 return T;
```

Algorithm 4: GENTRANSIT

Lemma 2 *Algorithm* GENTRANSIT *constructs an $n \times n$ matrix T such that $t_{uv} \geq 0$ for all nodes u and v, $\sum_{v \in N[u]} t_{uv} = 1$ for all u, and $\mathbf{p} \cdot T = \mathbf{p}$.*

We wish to use the these HCW models to generate agendas suitable for use in the agent-based simulator we discuss in detail in Chapter 4.

2.8 Comparison with simple models

While validation showed that centers-based models produce reasonable patterns of activity for HCWs, we introduce simple unit-based models for comparison. For each HCW i, (1) sum up observed activity in each unit of UIHC, (2) record the unit HCW i spends the most time in, h_i, and the percentage of time spent there, t_i. The stationary probability distribution for HCW i is thus $\frac{t_i}{size(h_i)}$ (where size(h_i) is the number of rooms on unit h_i) within h_i, and $\frac{1-t_i}{size(UIHC)-size(h_i)}$ everywhere else. This probability distribution can then be used in agenda generation in place of the centers-based models to facilitate comparison.

To compare agendas from centers-based models against those from our simple models, we look at how they differ first in terms of the total distance traveled by all HCWs, and second in terms of their contact networks. HCWs travel over 5 times as far in agendas based on the simple models as they do in agendas based on the centers models. Without instrumenting HCWs with pedometers or using fine-grained tracking data it is difficult to estimate how far the various HCWs at UIHC actually travel per shift, but this does indicate a significant difference between the simple models and centers-based models. Because HCWs in the simple unit-based model do not prefer one room in their home unit over any of the adjacent rooms, the random walks do not leave them in the same room for very long.

Next, we look at contact network graphs based on agendas generated from

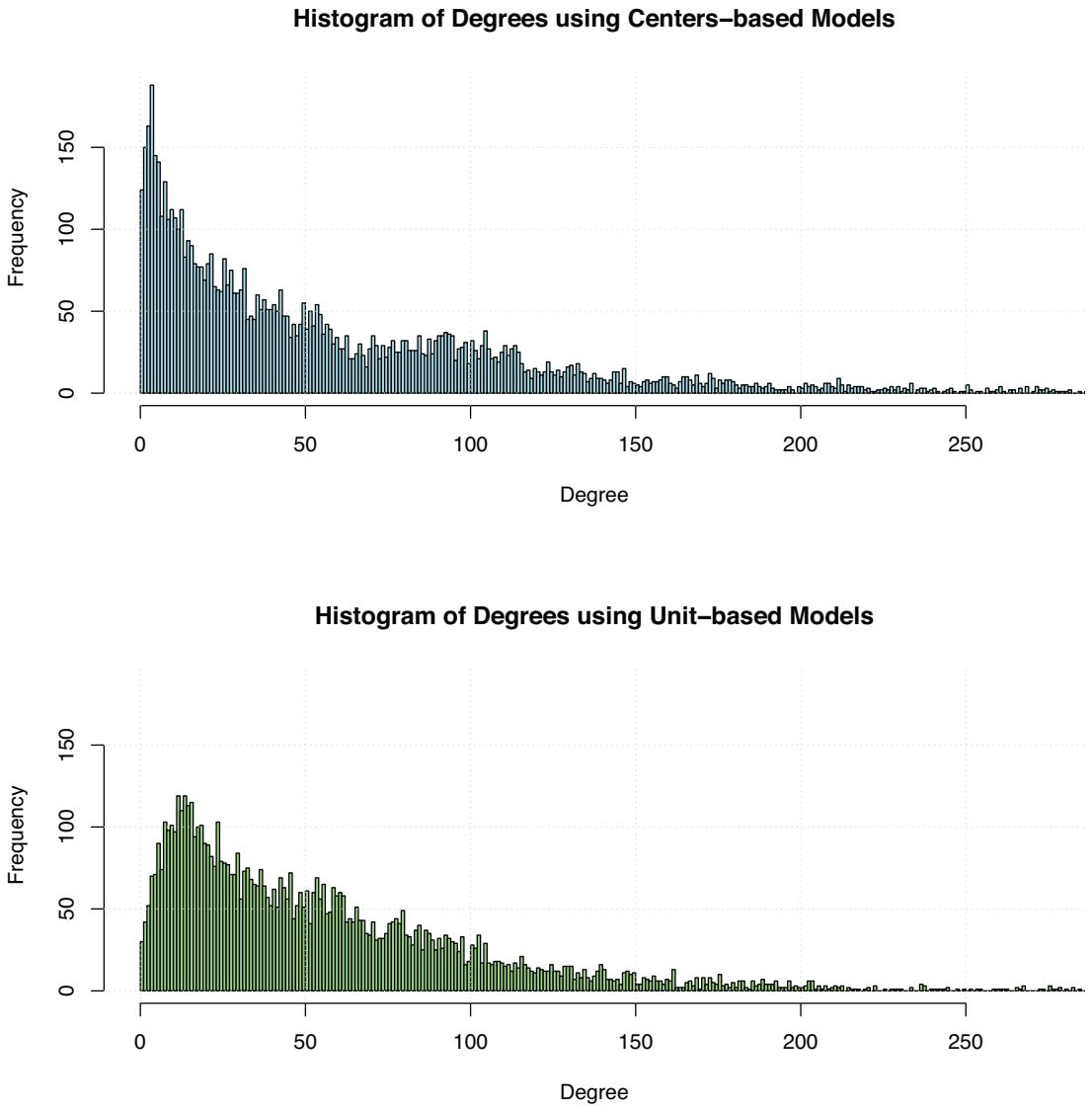

Figure 2.18: These degree distributions highlight a difference between our centers-based and our simple unit-based HCW models. Figure 2.18(a) shows the degree distribution for the contact graph generated by simulating activity generated by the centers models. Figure 2.18(b) shows the same distribution for activity generated by the simple unit-based models. Note that both distributions are heavy-tailed, however, the mode of the centers-based contact graph degree distribution is 4 contacts, while the mode of the unit-based contact graph degree distribution is 12. Further, the mode occurs 188 times in the former and only 119 times in the latter. See Figure 2.8 for more detailed graph statistics.

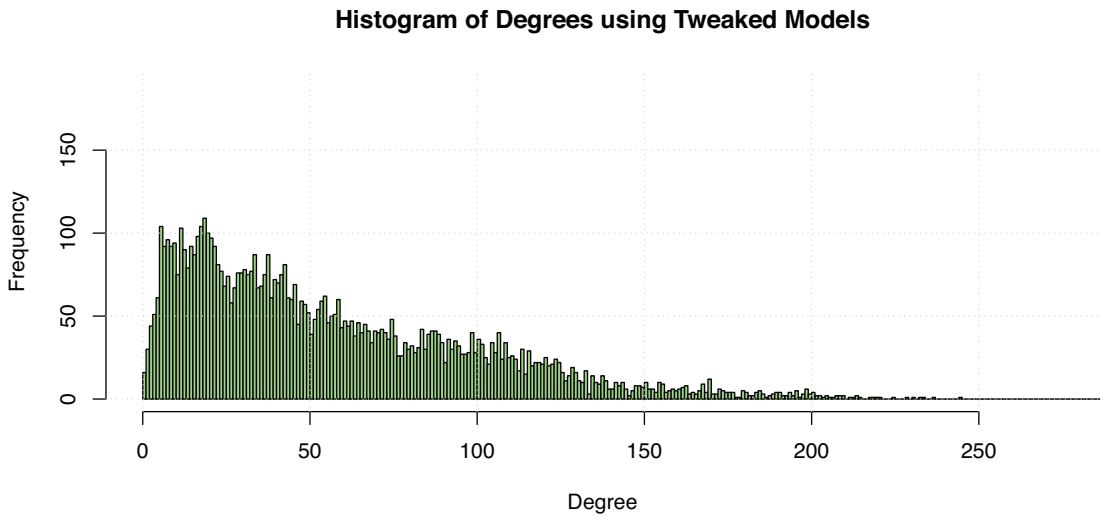

Figure 2.19: Reducing the γ values of the HCW models does indeed result in contact graphs with degree distributions that resemble those based on the unit-based models. Note the mode is shifted to the right and occurs much less frequently than in the original centers-based models. See Figure 2.8 for more detailed graph statistics.

Graph Statistic	Centers-based Graph	Unit-based Graph	Depressed Graph
Num. Nodes	6878	6878	6878
Num. Edges	193986	193986	193986
Clustering	0.6955	0.5734	0.6820
Mean Degree	56.4	56.4	56.4
Max Degree	390	414	245
Degree Std. Dev.	56.5	48.9	43.3
Degree Dist. Mode	4	12	19
Occurrences of Mode	188	119	110
Diameter	15	9	14
Avg. Path Length	4.2	3.5	4.6

Table 2.8: Centers-based and unit-based contact graphs have small world properties such as high clustering coefficients, low diameter, and low average path length.

The depressed contact graph, which is generated based on tweaked centers models with increased mobility for low-mobility HCWs, shares some properties with the centers-based graph and some with the unit-based graph.

the unit-based models and those generated from the centers-based models. While both sets of contact graphs exhibit many "small world" properties such as high clustering coefficients, low diameter and average path length between individuals, and heavy-tailed degree distributions, as shown in Figures 2.18 and 2.8, there are some differences in the degree distributions. Note how quickly the frequency peaks in the centers-based model contact graph degree distribution. The unit-based contact graph degree distribution has a higher mode, which also occurs much less frequently. One potential explanation for this discrepancy is that unit-based models result in more random mixing with other HCWs with the same home unit, which makes sense given that HCWs travel farther in the unit-based models. To test our hypothesis that more random mixing within a unit is responsible for these flatter degree distributions, we look at a second set of degree distributions based on slightly modified centers models. Figures 2.19 and 2.8 show what happens to the centers-based contact graph degree distributions if you artificially limit the γ values to be at or below 1.5, effectively forcing the more stationary HCWs to move around more. The degree distribution in this case shares some properties with the unit-based contact graph degree distribution, suggesting that one of the reasons for the flatter degree distribution in the case of the unit-based models is a higher level of mixing within units.

Comparing the unit-based contact graphs to the "shadow data" contact graphs discussed in Section 2.6.5 suggests a more fundamental problem. The average number of common job classes in the "neighbor list" for each class falls from 2.2 to 1.5, and in one case there are exactly zero common neighbors between the two sets of contact

graphs. This suggests a significant difference in the level of interaction between various job classes.

In addition to these problems, there are many other non-trivial shortcomings with the unit-based modeling: (1) it requires artificially imposing compartment (unit, in this case) boundaries on the hospital geometry, (2) it ignores expected room activity levels, (3) it cannot be expected to handle HCWs with multi-center models very well, and (4) it makes meaningful grouping / aggregation at the job title or job class levels more difficult. Regardless of which model is used, agenda generation is a significant time investment in itself, the computation required to generate agendas based on these simple-to-compute unit-based models is non-trivial.

Extensions to the unit-based model to deal with these and other issues are possible, but it seems a slippery slope with the centers-based models at the bottom. Extensions need to deal with non-uniform room activity levels, avoid increasing model complexity and memory usage, avoid over-fitting the observed data, and either address a serious weakness or come with a clever trick for efficiently generating agendas that bypasses the use of transition matrices.

CHAPTER 3
PATIENT MODELING

3.1 Introduction

Our model of a hospital is incomplete without an adequate model of patient flow. Patient flow models, when combined with HCW models, allow us to study numerous problems, including patient care issues, e.g., intensive care recidivism. Intensive care recidivism occurs when a patient is transferred or discharged out of an intensive care unit (ICU), closely followed by readmission to an ICU.

Such recidivism is associated with increased mortality rates, lengths of stay (LOS), and cost of care: more specifically, mortality rates are up to six times higher for patients readmitted to ICUs, and such patients are up to 11 times more likely to die in the hospital [39]. It should be noted that while ICU recidivism is widely recognised as a serious problem, there is no consistently used definition. Some studies include ICU readmissions only within the first 48 hours [88], while many do not mention a time window and presumably include entire visits [10, 74, 23], and still others consider ICU readmissions across hospital visits [22]. For the purposes of this work, we classify a pair of transfers as indicating recidivism if they occur during the same visit and within one week of each other, the first transfers the patient out of an ICU to a non-surgery unit, and the second indicates a readmission to any ICU. Our definition includes all patients hospital-wide, rather than focusing on a specific unit or age group.

In addition to ICU recidivism, we are also interested in clinical and operational issues that may occur when a hospital's ICUs operate near capacity. Previous work

looking at clinical outcomes, LOS, and cost as a function of patient load in ICUs has generated mixed results. A few studies have found that increased load (i.e., increased patient census) negatively affects patient outcomes and / or LOS [21, 19, 108], though Iwashyna et al. found no impact on mortality in their study [64]. Chalfin et al. looked at the effects of delayed transfer to intensive care on patient outcomes and found that such delays resulted in increased LOS, and increased mortality [18]. Hospitals thus face difficult decisions when ICUs begin to fill up.

With our interest in exploring similar patient care issues in mind, we examine previous work in patient modeling. Unlike spatial modeling of HCWs, the flow and care of patients has attracted the attention of many groups over the last 40 years. In a review of the literature, Marshall et al. point out that clinical patient flow models, in which each patient transitions through a series of health statuses, can be built using data that is already commonly collected. However, patient flow models based on an operational perspective, in which patients occupy specific locations in a healthcare facility, can be much more complex and it can be expensive and time consuming to collect data to train such models [79]. While developing patient flow models based on an operational perspective does require more data and complexity, both our interest in epidemiological simulation and our interest in exploring the issues surrounding ICU recidivism are better served by operational models, and we are fortunate to have access to fine-grained patient flow data from UIHC.

Operational patient flow models often involve the use of discrete event simulation. For these simulations, the population in the hospital is divided into 2-4

compartments, and the population in the community is divided into 1 or 2 compartments. The population in one compartment has some rate of transfer to a subset of other compartments [70, 62, 38, 40]. Our patient models are inspired by this work, though we also draw from clinical modeling techniques since we are particularly interested in modeling at the agent-level to support epidemiological simulation. From a clinical perspective, stochastic models for patient flow and optimal priority bed distribution include Markov models, phase-type models, conditional phase-type models, mixed exponential distributions, compartmental modeling, and simulation modeling [27, 41, 112, 8, 79]. Many of the applications for patient models also draw from or are related the broad field of operations research / management sciences work, e.g., modeling manufacturing operations [52] and the automated control of internet services [4].

3.2 Model Description

3.2.1 Single Compartment Model

We consider two patient flow models. The first, the *single compartment* model, is a simple three parameter model. The *fill rate* specifies the number of total hospital beds to be occupied at any given time and the other two parameters specify the patient LOS distribution. While it is widely acknowledged that frequency of patient LOS does not follow a normal distribution (e.g., [118]), there is some disagreement in the patient modeling literature about what probability distribution should be used for modeling patient LOS. A number of groups use geometric or exponential distributions [19, 97], and indeed we initially chose an exponential distribution for the sake of simplicity and

ease of training. However, we find that this distribution tends to poorly fit observed data for very short hospital stays since it peaks immediately, while LOS in observed patient data peaks on the second day. Thus, as suggested by Frick et al. and others [46, 101], we also consider a log-normal distribution for LOS.

3.2.2 Unit-based Compartmental Model

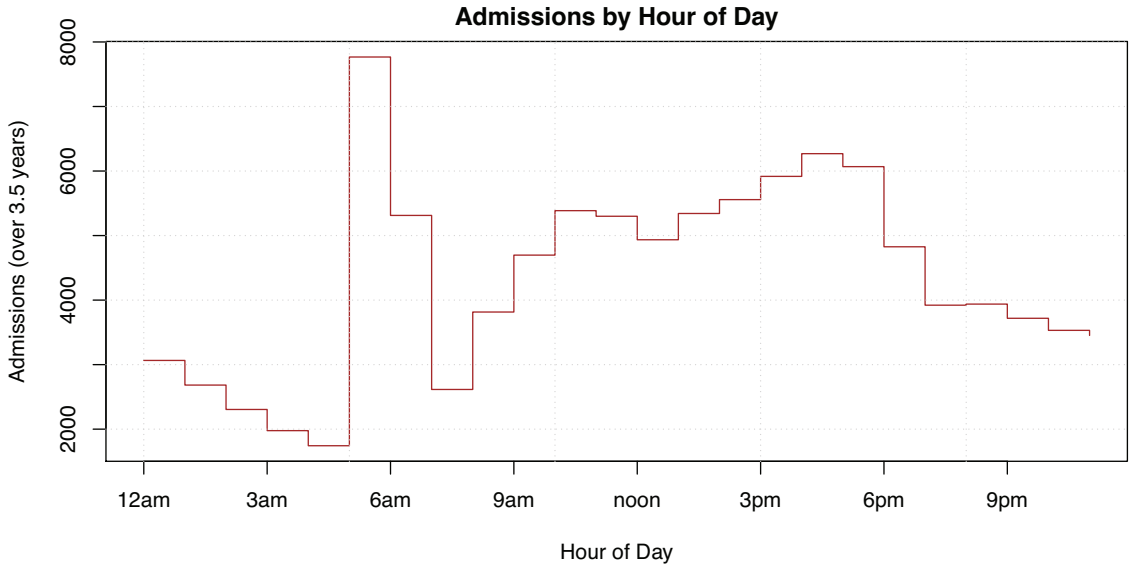

Figure 3.1: Somewhat higher numbers of patients are admitted between 9am and 5pm than during other hours of the day. A spike is clearly visible between 5am and 7am, when elective surgeries are typically scheduled.

We model the space of the hospital at the unit level. There is a compartment in our model for each unit in the hospital, as well as an additional compartment representing the community. Each compartment has a list of patient bedrooms, a capacity, and is either an ICU, a surgery unit, or a general unit. The community compartment has infinite capacity, and transfer to and from the community represents

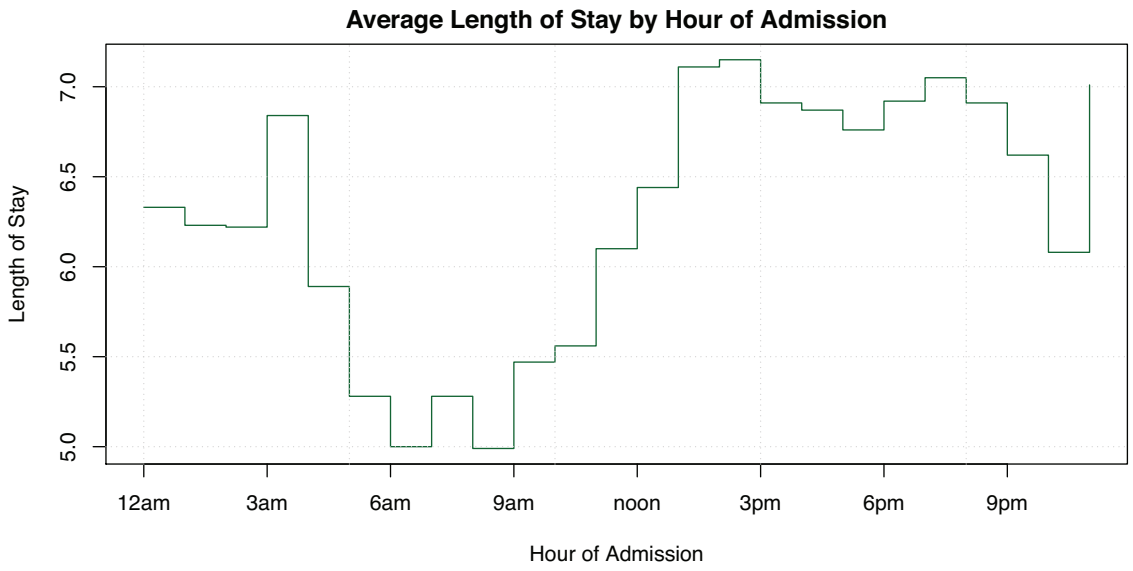

Figure 3.2: Patients admitted between 5am and 9am spend an average of 1-2 fewer days in the hospital than those admitted during other hours of the day, due in large part to the large number of patients admitted for elective surgeries scheduled early in the morning.

admission and discharge (or death), respectively.

Given the current compartment of a patient, the next location and the waiting time before transfer are a function of the patient's time of admission into the hospital, as well as the *mode* of the patient. The inclusion of a patient mode is inspired by research suggesting significant differences in clinical outcomes and operational efficiency based on the patient load in ICUs [19, 21, 108] at the time of transfer or discharge. *Demand-driven* discharges occur when relatively stable patients are transferred out of an ICU to free up beds for less stable incoming patients. While few would admit such triage takes place (and often it is not explicit policy), Chan et al. found evidence that it does, going so far as to study different policies for choosing which patients should be transferred first when ICUs begin to fill up [19]. Due to difficulties associ-

ated with identifying demand-driven discharges in observed data directly, we instead identify *potentially* demand-driven transfers. A potentially demand-driven discharge is a transfer out of an ICU while the ICU load is relatively high. Our patient mode can either either be *normal*, the default upon entering the hospital, or *post-demand-driven* indicating that the patient has previously been transferred out of an ICU at a time when the ICU was under a high load. This mode is "sticky", and persists through the remained of the patient's stay. We call the parameter which determines what ICU load is considered high the *ICU high load* threshold.

Figure 3.1 and Figure 3.2 motivate our incorporation of admission time into our model. The rate of admissions, transfers, and especially discharges varies by time of day with the 9-5 workday seeing more patient activity than other hours of the day. The workday effect is especially dramatic with respect to discharges, with patients being over 100 times more likely to be discharged at 3pm than 3am. The average patient LOS also exhibits a time of day dependency. The two figures taken together show a spike in admissions between 5am and 7am, and a low average patient LOS between 5am and 11am. This is likely due to the fact that patients admitted for elective surgeries are typically scheduled in the early morning, while patients admitted through the emergency room and other sources are more evenly spread out over the day. Elective surgeries typically do not result in long hospital stays, so such admissions drive down the average LOS. To allow our patient flow model to capture this daily variation, we divide the day up into four six-hour blocks and train separate compartment-compartment transition matrices and LOS distributions for each (block,

mode) pair. That is, e.g., (block=12pm-6pm, mode=normal), has associated with it a full transition matrix and an LOS distribution for each possible transition. For the reasons outlined in 3.2.1, we use the two parameter lognormal distribution for our LOS distributions.

We also model patient transfers as being randomly extended such that the number of admissions, transfers, and discharges per hour fit some distribution (e.g., the corresponding values observed in the observed patient data). For example, when a patient arrives on some unit (current), the next unit is picked from the appropriate transition matrix. The patient's LOS on the current unit is pulled from the LOS distribution for the (current, next) transition, and then randomly extended such that the probability of transfer during each hour of the day matches the observed transfer proportions. For example, if drawing from the LOS distribution yields 4pm Tuesday as the transfer time, the transfer has some appropriate probability of occurring at each hour between 4pm Tuesday and 4pm Wednesday.

Note that while our model does not explicitly capture the idea of patient health status, it could be extended in the future after appropriate training data are made available.

3.3 The ADT Dataset

One of the datasets made available to us by UIHC contains every patient admission, discharge, and transfer (ADT) between January, 2006 and June, 2009. For each of the 104,543 inpatient visits, the ADT dataset gives us the date and time of admission and discharge, a classification of the service caring for the patient, and

information about the bed, room, and unit to which the patient was admitted. For each of 307,692 transfers, we are given the date and time of transfer, and the bed, room, and unit of both the source and destination of the transfer.

Unfortunately, initial analysis revealed a number of problems with the ADT data, which had to be dealt with before it was suitable for training patient flow models. As with the EMR data used to train HCW models, many (171,122 in the case of the ADT data) rows contain no room level data or refer to rooms that do not map to a room in our hospital graph. Fortunately, we have fairly complete information about the location of patient rooms within UIHC, and even when visit or transfer records are missing room-level information, the vast majority still contain unit-level location data: only 0.1% of transfers are missing this unit-level information, and no visits are missing at least unit-level information. Since we train models at the unit level, we only care about completeness of the room-level information to the point where we can assign each patient bedroom to a unit. While some rooms are used by multiple (potentially many) units, there are only a handful of rooms for which there is no clearly dominant unit. We thus map each room to the unit that uses it the most. Throughout the 3.5 years we have data for, we identify roughly 650 rooms patients can stay in, and 800 patient beds (including operating rooms and nursery beds).

Of more serious concern, we observe many cases (on the order of 25% of transfers) where transfer sources, destinations, or timestamps are inconsistent with either other transfers or the visit-level information. For example, there are cases where transfer timestamps are exactly the same time as admission time, and some where

transfer times are past the time of discharge. There are also cases where location information in consecutive transfers does not match. For example, a transfer at 10am specifies destination Room 4205, but the next transfer occurs at 3pm the same day and cites Room 4825 as the source. In these cases, we have to make a decision about how to patch the hole in our data. We identified and implemented three strategies for dealing with such problems:

1. Accept earliest data, and modify later data: in this example, this means modifying the second transfer's source to be Room 4205.
2. Accept newest data, and modify the earlier data: in this example, this means modifying the first transfer's destination to be Room 4825.
3. Accept both transfers, and insert a new transfer to split the difference: in this example, this means inserting a new transfer from Room 4205 to Room 4825 at 12:30pm with a source.

In all cases, we ignore transfers that occur less than one second after admission or another transfer and we ignore transfers occurring after discharge.

The more transfers for a patient visit, the more likely the data contain errors; half of all visits exhibit one of these transfer problems. Nonetheless, we believe there is signal in the patched datasets. Wherever there was a choice in how to deal with a problematic transfer, we used each of the repair options and verified that models trained were similar to each other with respect to metrics such as average population level per unit.

3.4 Model Training

Training our single compartment models is straightforward. We simply count the total number of patients admitted to UIHC and the total number of seconds they have spent there. Knowing how many days we have data for, this allows us to compute the average hospital-wide patient load. We then find the maximum likelihood log-normal distribution to account for the observed lengths of stay.

We train our compartmental models in several steps. The first step is to classify each admission, transfer, and discharge as belonging to one of our eight states by looking at the time of day of admission, and determining if and when the patient left an ICU that was potentially capacity-strained. Because a patient transfer actually takes on the order of a couple hours and there is some noise in the ADT data, we transition a patient visit into the post-demand-driven mode if the volume of patients seen in an ICU on the day of the transfer is in the top quartile of patient volumes for ICUs within the hospital. We then go through each transfer (admission, discharge) and update the transition matrix and LOS distribution parameters for the (time of day, mode) pair associated with that transfer.

Lastly, we calculate the proportion of admissions, transfers, and discharges that take place during each hour of the day.

3.5 Agenda Generation

Patient agendas are generated by agent-based discrete-event simulators.

3.5.1 Single Compartment Model

For single compartment models, the patients are placed in random empty patient bedroom in the facility until the fill rate is satisfied, and each draws a length of stay from the log-normal distribution. As soon as a patient is discharged, a new patient is placed randomly in an empty patient bedroom. This process continues until the desired amount of data has been generated.

3.5.2 Unit-based Compartment Model

Algorithms 5 and 6 give an overview of the process by which agendas are generated from a unit-based compartment model. The simulator takes as input the number of patients that should be in the hospital at any given time, the probability distribution by hour of day for admissions, transfers, and discharges, and a unit-based compartment model.

Note that the community compartment has infinite capacity, and a population is not maintained there. All admitted patients are assumed to be unique and thus recidivism is only within-visit, not across visits.

3.5.2.1 Initialization

As shown in Algorithm 5, enough patients are queued up for admission at initialization such that the fill rate is satisfied, and each is assigned a time of admission pulled from the admission probability distribution observed in the ADT data. The simulator is run for some *warm-up* period (e.g., six month) to get the simulator into a steady state before agendas are actually recorded.

```
   input  : FillRate, admitsByHour
   output: initialized patient queue, Queue
 1 // Initialize the priority queue (sorted by time of
 2 // next transfer)
 3 Queue ← ∅;
 4 // Enqueue the required number of patients for admission
 5 while number of patients < FillRate do
 6 |   // GenPatientAdmission creates new patient, admission
 7 |   // respects admitsByHour
 8 |   patient ←GenPatientAdmission(simTime = 0);
 9 |   Enqueue(Queue,patient);
10 end
```

Algorithm 5: Unit-based Model Agenda Generation (Initialization)

3.5.2.2 Generation

Algorithm 6 outlines the process of generating agendas from unit-based models. Potential transfers, of which admissions and discharges are just special cases, are stored in a priority queue based on the earliest time the transfer can take place. When a transfer is dequeued, the capacity of the destination compartment is checked. If there is an unoccupied bed, the transfer goes through immediately, after which the next transfer for that patient is generated and added to the queue by consulting the appropriate transition matrix and drawing from the corresponding LOS distribution. The transfer time is then randomly extended by choosing an hour of day according to the transfers by hour probability distributed observed in the ADT data. If there is no free bed, the transfer will be queued and reattempted after some delay, unless it has already been delayed for multiple subsequent attempts. In that case, the destination is updated by resampling from the distribution of target compartments (excluding

```
input  : Queue, admitsByHour, dischargesByHour, transByHour,
         daysNeeded
output: patient agendas are recorded to disk
simTime ← 0;
while simTime < daysNeeded do
    patient ←Dequeue(Queue);
    simTime ←TimeForTransfer(patient);
    set patient mode to post-demand-driven if in high load ICU;
    if CanTransfer(patient) then
        Transfer(patient);
        if IsDischarged(patient) then
            if simTime > warmupTime then
                RecordAgendaToDisk(patient);
            end
            // Replace with a new patient
            patient ←GenPatientAdmission(simTime);
            Enqueue(Queue,patient);
        else
            // GenNextTransfer picks transfer time such that
            // transByHour is respected
            Enqueue(Queue,GenNextTransfer(patient, simTime));
        end
    else
        if TooManyDelays(patient) then
            // Attempt to route patient to unit with open bed
            dest ←NewDestination(patient);
            if dest == ∅ then
                Enqueue(Queue,ApplyDelay(patient));
            else
                UpdateDest(patient,dest);
                Transfer(patient);
                Enqueue(Queue,GenNextTransfer(patient, simTime));
            end
        else
            Enqueue(Queue,ApplyDelay(patient));
        end
    end
end
```

Algorithm 6: Unit-based Model Agenda Generation

the community) until a compartment with an unoccupied bed is chosen. If no such compartment exists, the transfer is queued and delayed once again. This process continues for each agent until the agent is transferred to the community compartment, which indicates discharge or death. At that point, a new patient is queued for admission.

Two parameters are tuned using statistics based on the output of agenda generation simulation. The first parameter that needs to be tuned, the ICU high load threshold, specifies the level at which ICUs should be considered full or near-full. We set the threshold such that 75% of the hospital ICU daily loads are less than or equal to this value. Since changing this threshold will change the dynamics of the simulation, it may take several iterations to find a good threshold. Finally, we scale the compartment LOS distributions such that the overall LOS distribution for patients matches the observed patient LOS distribution.

3.6 Validation

Figures 3.3, 3.5, and 3.4 show the LOS distributions for the observed UIHC patients, and typical LOS distributions for patient agendas based on single compartment models and unit-based compartment models. Note that observed LOS distribution and simulated (unit-based) LOS distribution both exhibit cyclically occurring peaks and troughs. Patients are much more likely to have a LOS near an integer number of days than they are to have a LOS halfway between two full days. This is explained by the fact that patients are more likely to be admitted during the day, and much, much more likely to be discharged during the day. Note that all of these LOS distributions

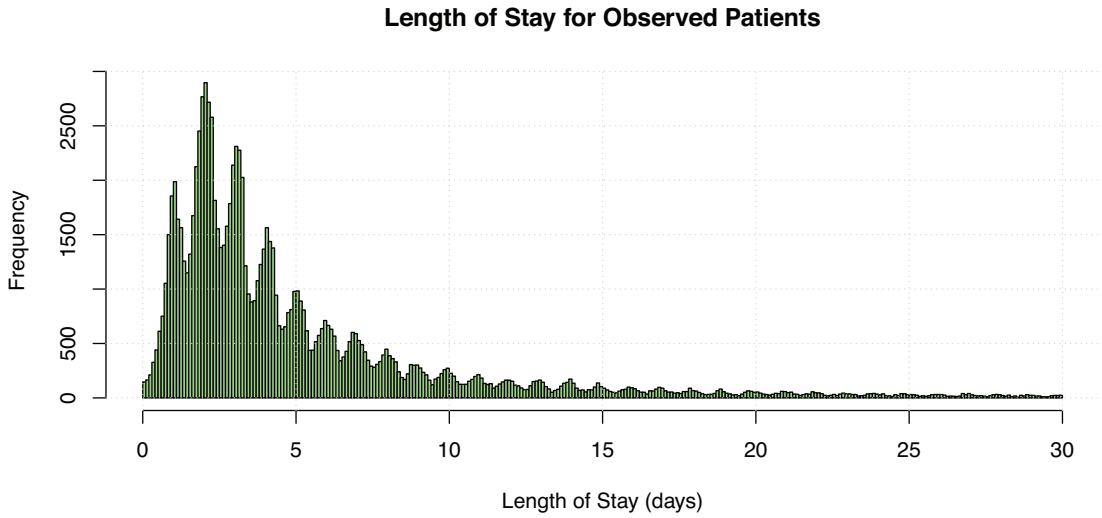

Figure 3.3: The observed LOS distribution based on the ADT data contains periodic spikes near full days indicating patients stay full days rather than partial days. This behavior occurs because admissions are disproportionately likely to occur during the day, and discharges are very unlikely at night or very early in the morning.

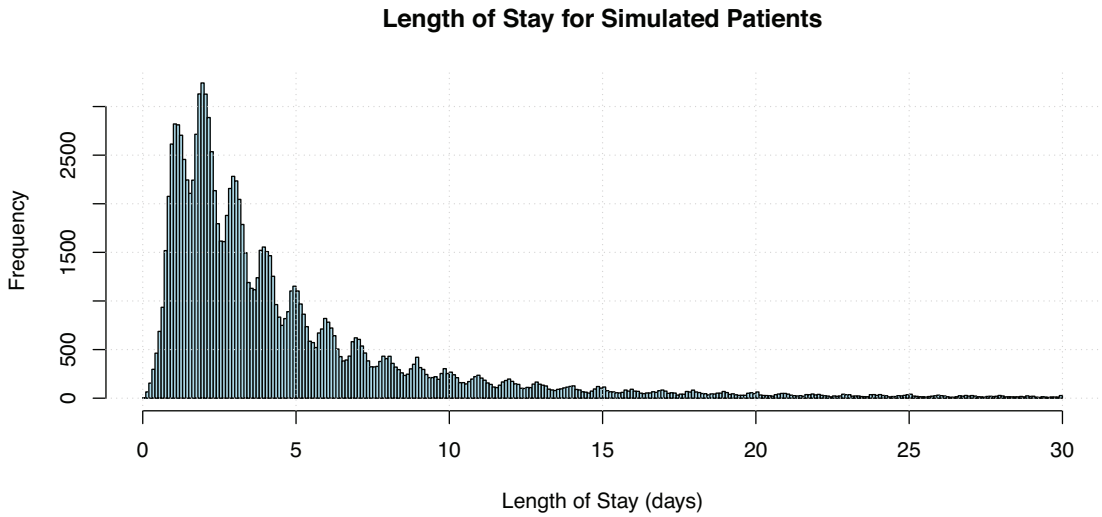

Figure 3.4: Simulated LOS are heavy tailed, and share many characteristics with the observed LOS distribution. Patients are likely to stay near full days rather than partial days, the distribution peaks on day 2, and the distribution is heavy tailed.

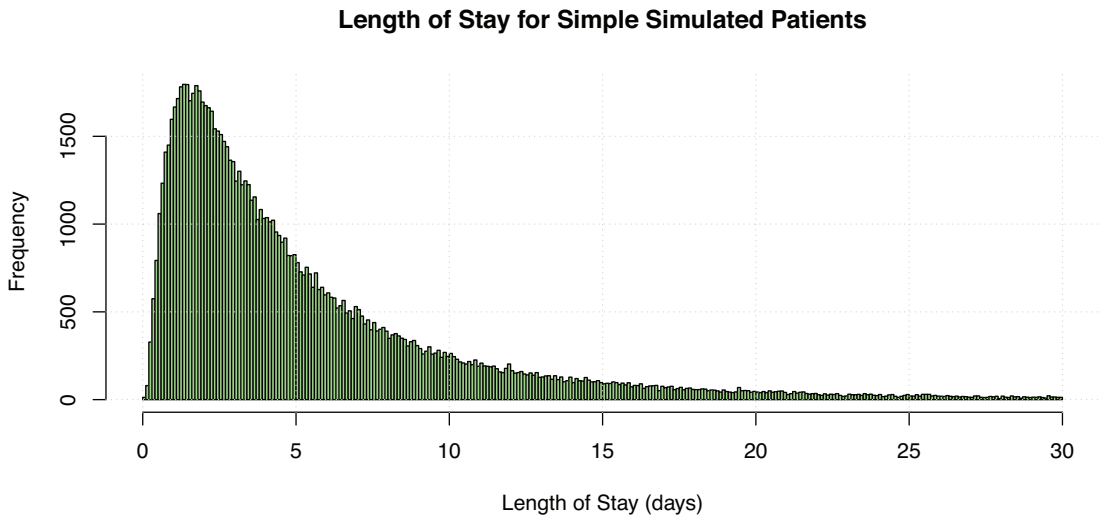

Figure 3.5: Simulated patient LOS for patients generated by on the single-compartment patient model is missing the daily spikes. The single compartment model still manages to capture the fact that LOS peaks on or near day 2, and is heavy tailed.

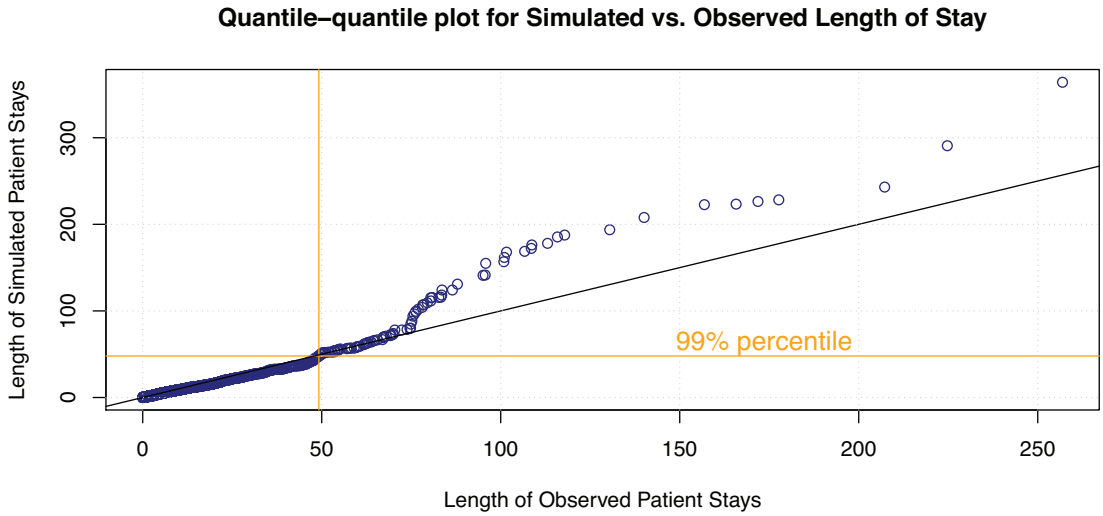

Figure 3.6: A quartile-quartile plot of observed LOS versus simulated LOS shows a very close match of the distributions for LOS values under 70 days. For LOS > 70 days, the LOS values in the simulations tend to be longer, though it should be kept in mind that, despite the large amount of real-estate occupied by this region of the plot, few patients in either data set ($< 0.5\%$) have LOS values in this range.

High Load Threshold	Recidivism Rate	LOS	Simulated 75th %ile Load
Disabled	0.1216	6.149	0.730
0.90	0.1438	6.343	0.786
0.85	0.1497	6.385	0.799
0.825	0.1557	6.447	0.809
0.80	0.1567	6.462	0.816
0.75	0.1625	6.489	0.816
0.70	0.1675	6.555	0.826
0.60	0.1698	6.601	0.831
0.00	0.1744	6.639	0.832

Figure 3.7: Our ICU high load threshold is meant to be chosen such that the ICU daily census is less than or equal to that threshold on 75% of simulated days. Setting this threshold at 0.825 yields the desired property that the threshold parameter and simulated ICU loads are nearly the same value. This value also happens to be close to the 75th percentile ICU load observed in the ADT data, which is 0.76. Note also that the recidivism rate for the chosen ICU high load threshold is close to the observed readmission rate of 0.1391, though somewhat greater.

peak at $LOS = 2$, which would not be the case for an exponential distribution.

Figure 3.6 shows a quartile-quartile plot comparing the observed LOS distribution to the simulated LOS distribution. Note that they match very closely until the LOS reaches 70 days or so, at which point the generated patients have longer stays. It would be possible to use separate distributions for short and long LOS, but given that there are so few patients with $LOS > 70$ ($< 0.5\%$), a single distribution is sufficient for our purposes. It should be noted that while the observed and simulated LOS distributions appear to be similar in many respects, a chi-square test for goodness of fit was inconclusive ($p = 0.25$).

With regard to recidivism rates, the single compartment models are not useful: the recidivism rate is trivially 0 for all sets of generated patient agendas since patients are never transferred. In the observed patient stays and under our definition of ICU recidivism, we find that 13.9% of patients transferred out of an ICU are readmitted.

Figure 3.7 shows the same statistics for a few different choices of ICU high load thresholds. The ICU loads and recidivism rates in simulated patient agendas are similar to those observed in the ADT data.

It is worth noting that the single compartment models (or even the 4-5 compartment models more commonly found in the literature) also fail to capture the fact that patients can interact with a diverse set of HCWs by moving from unit to unit. This seems to be of particular concern when studying the spread of infections within a healthcare environment.

CHAPTER 4
APPLICATIONS

4.1 Introduction

A realistic hospital simulator can be used to study a wide variety of problems including the spread of nosocomial infection, various operations research problems, such as staffing assignments, bed allocation, etc.[98, 35], as well as the many of the patient care problems mentioned in Chapter 3. In this Chapter, we showcase three such applications.

First, we will use our simulator to explore the effects of different infection control policies on the spread of mumps. Second, we look at social networks within the hospital, which might be used, e.g., when determining which HCWs should be a priority during vaccination campaigns[30, 31, 32, 28]. These first two applications are taken from the emerging field of computational epidemiology. Finally, we consider an instance of a resource allocation problem: where to place time clocks in a hospital so as to minimize a particular cost function.

4.2 Agent-based Discrete-event Simulation

Algorithm 7 shows our simulation framework. We use an agent-based discrete-event simulator to study infectious diseases and social networks. We initialize by creating an empty hospital. Next, we read in our HCW and patient agendas, which are generated as described in Chapters 2 and 3, respectively. Each line in the agendas specifies an agent, a room to be occupied, and a time to enter and leave that room. The agendas are sorted by the room entrance times and read sequentially with agents

being moved around the hospital as appropriate. Our applications can then work on top of this framework, typically doing some work each time an agent leaves a room or after a some amount of time passes.

4.2.1 Epidemiological Modeling

```
input  : agendas, disease transmission parameters, etc.
output: contact graphs, disease graphs, etc.
// Initialization
// Our priority queue is ordered by the time of action of
// enqueued agenda items
queue ← ∅;
Hospital ← ∅;
// Queue up all room entrances
for each agenda in agendas do
    Enqueue(queue, EntranceTime(agenda), Entrance, agenda);
end
while queue is not empty do
    time, action, agenda ←Dequeue(queue);
    if action == Entrance then
        ProcessEntrance(Hospital, agenda);
        // Enqueue the corresponding exit event
        Enqueue(queue, ExitTime(agenda), Exit, agenda);
    end
    if action == Exit then
        // This is where contact graph or disease simulation
        // code would go
        ProcessExit(Hospital, agenda);
    end
end
```

Algorithm 7: Our framework for discrete-event simulation

It has recently become clear that the *random mixing* assumed by much of the

past infectious disease research is problematic, since it cannot explain, for example, the existence of super-spreaders that infect a disproportionately many number of other individuals. To move beyond random mixing, recent work has looked at the spread of disease on arbitrary contact networks rather than within compartments[45, 82, 81, 86, 29, 28, 117, 87], including work specifically considering nosocomial infections [120, 114, 113]. While some of these groups do consider the spread of disease on arbitrary contact networks, they are either forced to make assumptions about how individuals mix based on job role and unit assignment, or are just not looking at physical contact networks. To our knowledge this is the first effort to use agent-based discrete-event simulation to study the spread of disease within a hospital setting using real location data and on such a large scale.

Here, we study the effects of an infection control policy which quarantines symptomatic HCWs for 0, 5, and 9 days during a mumps epidemic. Our model of disease is based on the commonly used SIR model (see Hethcote for a review of the literature[54, 55]). Agents in our simulation are in one of three states: *susceptible*, *infectious*, or *recovered*. Susceptible agents have some chance of infection after coming into contact with infectious agents, thus becoming infectious themselves. Once infectious, an agent remains so for some period of time, after which recovery occurs. Agents that are recovered cannot become reinfected, and are no longer themselves infectious. Note that agents have some probability of developing symptoms some prespecified short time after becoming infectious, meaning some will remain asymptomatic. We model vaccination by setting vaccinated agents to the recovered stage

without going through the infectious stage, and we model quarantine by preventing agents from infecting other agents for the appropriate number of days.

4.2.1.1 Methods and Results

We review the mumps literature to determine appropriate ranges for the various disease parameters, including the probability of transmission given contact, the shedding curve (which we use as a proxy for the probability of being infectious after a period of time has passed), the odds of developing mumps-specific symptoms, the waiting time before symptoms develop, and the odds a particular HCW or patient has been successfully vaccinated against mumps[14, 106, 78, 94, 93, 80].

input : two agents ($agent_1$, $agent_2$) for which contact is ending, transProb
output: updated patient disease states

```
// Assumptions:
// 1) there is one shot at infection after contact ends
// 2) duration of contacts ≪ infectious period
// 3) agents ordered such that if either agent is
// infected, agent_1 is infected
if Infected(agent_1) and Susceptible(agent_2) then
    // Quarantined agents are unable to infect other agents
    if Quarantined(agent_1) then
        return;
    end
    // Shedding peaks on day two after exponential ramp up
    // and decays exponentially after day two
    r ←rand();
    if r < transProb · SheddingForDayOfInfection(agent_1) then
        Infect(agent_2);
    end
end
```

Algorithm 8: Determine whether infection should spread after the end of a simulated contact

Because it is difficult to directly observe an infection physically spreading from one individual to another, it is non-trivial to determine exact infection parameter values. To handle this uncertainly, we perform many replicates over a range of parameter values to ensure that results are robust. We also run many replicates with a range of infectious disease parameters and three different infection control policies: 1) do not quarantine symptomatic HCWs, 2) quarantine symptomatic HCWs for 5 days, and 3) quarantine symptomatic HCWs for 9 days.

Our infectious disease simulator has both HCW and patient agents, with agendas generated as described in Chapters 2 and 3 respectively. A small number of agents are randomly infected to seed the epidemic, and the agent agendas are played out. After every contact between two agents ends, the simulator decides whether or not to spread the infection based on the infection status of those two agents at that particular time as shown in Algorithm 8.

Figure 4.1 shows statistics about the spread of mumps for the three different infection control policies. Our simulations show that quarantining symptomatic HCWs significantly reduces the transmission of disease. Quarantining patients for 9 days rather than 5 days results in a further reduction in infections over the 5 day alternative. Of course, this ignores the costs associated with increased quarantine duration. It should be noted that mumps is rarely fatal, and there is a non-trivial cost associated with implementing a quarantine policy. Indeed, the Centers for Disease Control and Prevention, the Healthcare Infection Control Practices Advisory Committee, and the Academy of Pediatrics all changed their recommended mumps quarantine periods

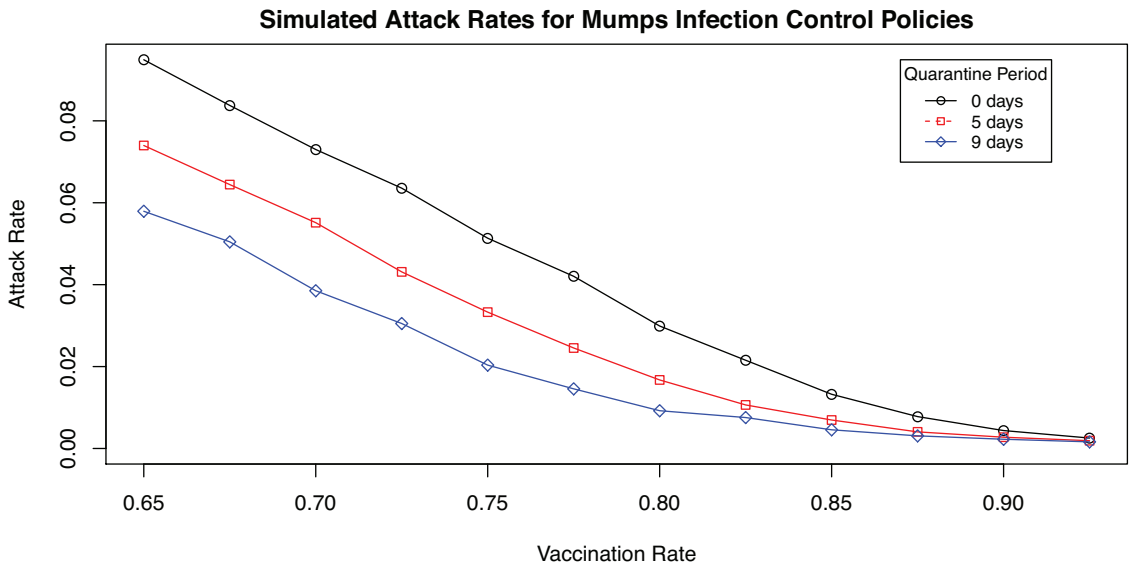

Figure 4.1: Attack rates assuming a transmission probability of 0.075% per contact show that quarantining effectively reduces the transmission of mumps in our simulations, with the 9 day quarantine reducing transmission further than the 5 day quarantine, especially for low vaccination rates / effectiveness. However, mumps is rarely fatal and implementation of the 9 day quarantine is potentially costly and burdensome.

from 9 to 5 days in 2007-2008 after a study showed that compliance was higher (in university settings) with 5 day quarantine periods [1].

4.2.2 Contact Graphs

In contrast to traditional *random mixing models*, *contact network epidemiology* [81, 86] is a relatively new area of research that investigates the spread of disease through a population based on intrinsic features of the pathogen and structural properties of a contact network (graph) that explicitly models physical interactions between individuals. The relevance and verisimilitude of contact network epidemiology depends mostly on the quality of the contact networks. Constructing reliable contact networks is a challenge since these networks model physical interactions as

opposed to online interactions.

4.2.2.1 Graph Generation

During the initialization phase of our simulator, we create a graph whose nodes are agents (HCWs and patients) and whose edge set is empty. Each time a pair of agents coexists in a room, an edge is created (or updated) with the weight set to be the amount of time spent together. The graphs generated by this process, despite being sparse (average degree = 1% of graph size), exhibit properties (i.e., having a high clustering coefficient, small diameter, a heavy tailed degree distribution, etc.) that are typical of "small world" social networks (see Table 4.1)[117, 87].

For comparison, we also report corresponding features for the Erdös-Renyi random graph model $G(n,p)$ set to yield approximately the same expected degree. More specifically, we see that our graphs have a clustering coefficient of 0.6675, which

n (num. nodes)	8,785
m (num. edges)	193,985
$\langle k \rangle$ (mean degree)	44.16
k_{max} (max. degree)	317.0
σ (std. dev. degree dist.)	47.16
σ_{rand} (std. dev. degree dist. $G(n,p)$)	7.84
cc (clust. coeff.)	0.6675
cc_{rand} (clust. coeff. $G(n,p)$)	0.0100
c (num. components)	58.0
c_{rand} (num. components $G(n,p)$)	1
n_{giant} (num. nodes giant comp.)	8,475.37 (96.47%)
m_{giant} (num. edges giant comp.)	193,945.63 (99.26%)
$diam$ (diam. giant comp.)	17.0
$\langle \ell \rangle$ (ave. path len. giant comp.)	4.423

Table 4.1: Statistics for contact graph.

is orders of magnitude larger than the clustering coefficient of the Erdös-Renyi graph

of similar size and edge-density. Our contact graphs have one giant component, consisting of more than 96% of the nodes, along with many tiny connected components. This makes it structurally very different from the comparable Erdös-Renyi graph, that has a single connected component. The giant components in our contact graphs have a very small diameter (17) and average path length (4.424) relative to their size.

This attempt at generating HCW and patient contact networks extends two of our earlier, more primitive attempts [31, 30, 96, 29]. In [31, 30, 32] we used the EMR login data directly to obtain HCW contacts by creating a contact (edge) between two HCWs who have logged in in nearby locations at roughly the same time. Those prior results serve as examples of additional potential applications of contact graphs generated from the center-based models described in Chapter 2.

4.3 Time clock placement

The UIHC is a very large tertiary care facility with about 8,000 HCWs on staff. Since HCWs clock in at the start of the shift, good clock placement can affect institutional efficiency and therefore, profitability. There are two kinds of clocks: a basic model whose functionality is limited to clocking in, and a premium model that is capable of displaying information such as the cafeteria menu for the day and use of sick days. UIHC is interested in placing the clocks near where people start the work day so they can clock in and out immediately before and after work and cut down on time spent walking to and from their work locations while clocked in. The clocks with more advanced functionality are targeted toward the nursing staff.

Roughly speaking, the UIHC is interested in placing clocks so as to minimize

average distance from a HCW to a time clock. If HCWs had fixed locations or if we had full knowledge of their start and finish locations, then this problem could be modeled as the well known *k-medians* clustering problem, which aims to minimize the average distance to the closest of the k median locations. But it turns out that the k-medians problem is a useful model even without precise location information, because we have estimated HCW spatial distributions. Algorithms for the k-median problem can work with spatial distributions rather than exact locations and minimize the average expected distance between HCWs and clocks.

We solve the problem in two separate steps. The first step is deciding where to put the time clocks, and the second step is deciding which clocks should be the basic model, and which should be the premium model.

4.3.0.2 k-means and k-medians

We first implemented a simple 2-approximation algorithm for the k-means problem [58], which aims to minimize the maximum distance between any HCW centers and the closest of the k mean locations. We find that a typical run yields a solution with at least one HCW needing to travel 30 units to reach a clock, and on average a room has a clock 11 units away.

We then implemented a 5-approximation algorithm for the k-median problem. This algorithm starts with a feasible solution and then tries to swap a clock from one room to another (presumably without a clock) until no swaps with improvement above a threshold value are found. Arya et al. [5] show the approximation ratio of the algorithm to be bounded by $OPT \cdot (3 + 2/numSwaps)$ where $numSwaps$ is

the number of simultaneous swaps allowed. With nearly 20,000 candidate locations and 80 clocks though it was not computationally feasible to use anything other than $numSwaps = 1$. By solving the k-medians problem we were typically able to get the average expected distance from a HCW to a clock to be 9.2 weighted hops. Figure 4.2 shows a fifth floor slice of a sample solution, with the lighter dots indicating rooms with low HCW activity, and the darker dots indicating high HCW activity.

4.3.0.3 UIHC: Was k appropriate?

For any clustering problem we need to be considerate of k. UIHC was also interested in determining if 80 was a sufficient number of clocks to purchase. Assuming that queuing is not an issue, Figure 4.3 suggests that 80 clocks provides adequate coverage and adding more clocks would provide little benefit in this respect. However, depending on usage habits fewer clocks could result in longer wait times. There is an opportunity for future work studying how the clocks are being used in practice and incorporating wait time costs or capacity constraints.

4.3.0.4 Allocating the premium clocks

The process of allocating the premium clocks is straightforward. For each clock, count the expected number of nurses assigned to that clock. The clocks with the highest expected number of nurses are allocated the premium model, and the rest of the clocks are the basic model. It should be noted that the allocation of premium clocks could potentially have an affect on which clocks nurses use. For example, a nurse might be willing to walk a little further to use a premium clock. Our cost function assumes nurses will use the closest clock and does not penalize long walks

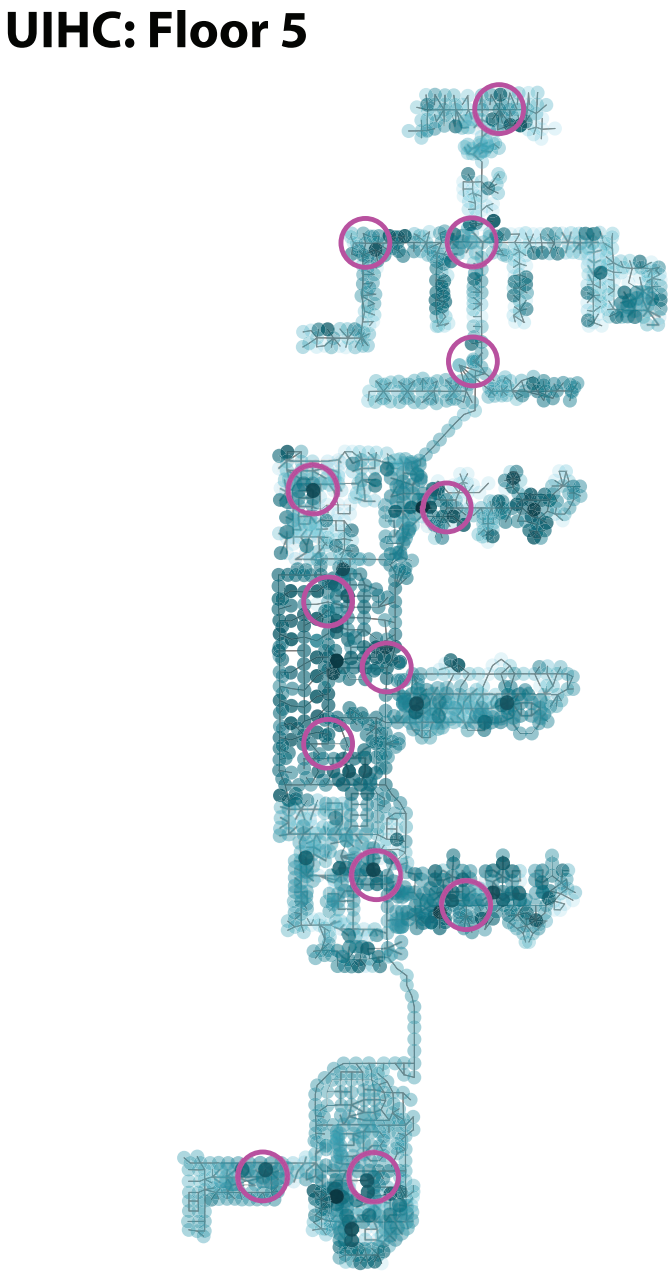

Figure 4.2: Time clocks placed on floor five of the UIHC facility by the weighted k-median algorithm. Dots represent expected levels of activity and circles represent placed time clocks. Each HCW can expect to travel about nine weighted hops to reach the nearest time clock.

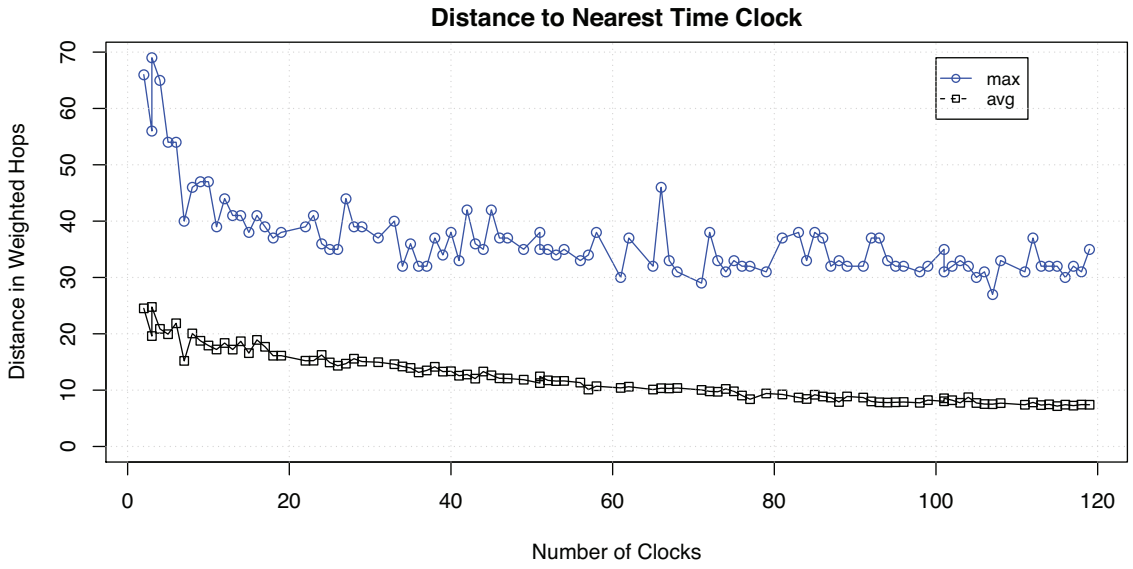

Figure 4.3: The maximum and average distance a HCW needs to travel to reach a clock versus the number of clocks k.

to premium clocks in particular.

CHAPTER 5
CONCLUSIONS

We have built and described a realistic simulator that can be used to answer a wide variety of questions about infection prevention, resource allocation, and patient care. Our simulation framework requires three primary inputs; architectural, health-care worker, and patient data. In Chapter 1 we discussed potential architectural data sources, our spatial model, and the construction of our hospital graph. In Chapter 2 we discussed sources of HCW location data, our model of HCW spatial distributions, and the process for generating maximum likelihood models including an extension to better fit observed data for HCWs with multiple foci. In Chapter 3, we discussed our patient flow models and our process for generating patient agendas. Chapter 4 showcased three sample applications for our simulation and HCW and patient models: an infectious disease simulator, a contact network generator, and identifying location to place time clocks within the hospital to minimize some cost function.

In designing our simulator, we were able to minimize assumptions about how HCWs and patients move, effectively avoiding the "random mixing" assumption common to many infectious disease simulators. Our HCW models are also compact, flexible enough to be used for a wide variety of job roles, training is computationally feasible even for very large hospitals, and are generative. We translated techniques from location-aware search into the hospital environment, developed data structures for use in efficiently processing millions of location data points in tens of thousands of rooms for thousands of HCWs, and improved the performance of the algorithm for

identifying optimal single-center HCW models. We extended our models to allow multiple centers, proved that the proposed multi-center log-likelihood functions are not unimodal implying gradient ascent methods are not guaranteed to find the optimal solution, and introduced heuristics for training multi-center models that outperform existing heuristics both in required computation time and in fitting the observed data.

We leveraged our good working relationship with UIHC to get access to unusually fine-grained healthcare data. We were provided access to architectural blueprints, 19 million EMR HCW logins over a period of 22 months, and data for $104,543$ inpatient visits $307,692$ patient transfers over a period of 41 months. To the best of our knowledge, this is the first agent-level hospital-wide simulator based on fine-grained location and interaction data for healthcare workers and patients.

The EMR login data and ADT data, despite being rather noisy, seem to have enough "signal" to be able to provide robust estimates of spatial distributions of HCWs and patient flows in a hospital environment. Moreover, EMR login records are routinely available to nearly every modern healthcare facility. The estimated spatial distributions match our expectations quite nicely. As far as we know, this is the first model of HCW spatial distributions in the literature.

5.1 Future Work

Due to the significant and growing costs of healthcare delivery and its implications for quality of life, it seems likely that the body of work concerned with modeling and simulation within a healthcare environment will continue to attract a lot of attention. HCW spatial modeling in particular is a relatively unexplored area,

in which this work attempts to lay some groundwork. Because of its novelty, there are a number of ways in which these HCW modeling efforts could be streamlined or extended.

Training HCW models requires some notion of a metric space in which HCW activity takes place, in our case a graph theoretic model. Constructing such a spatial model is non-trivial, and could be improved in various ways. For example, to avoid the laborious task of manually building a hospital graph at other facilities, we have been exploring automated extraction of graphs from CAD files through the use of navigation mesh generation [83]. There is also the possibility of using pedometers, radar, or other techniques and technologies to automatically generate a graph without CAD files.

HCW modeling requires large amounts of HCW data either collected directly or inferred from other data sources. Our research group has been exploring the use of inexpensive, wireless sensor networks to directly collect fine-grained location and contact information in near real-time [56, 95, 60]. We have recently deployed a network of these "motes" in an intensive care unit at UIHC. While this dataset is confined to a single unit, it contains much more fine-grained data than the EMR data used in this paper. This dataset can be used to validate our spatial distribution models, overcome limitations in our EMR data, and to generate spatial distribution models based on more fine-grained data.

Being relatively new territory, there are numerous directions for further study of HCW models themselves. For example, we have begun exploring how other proba-

bility decay functions (e.g., exponential) might be incorporated into our model. There are also potentially numerous opportunities for refining the multi-center models and model training. It may be possible to develop better heuristics, and there is also the opportunity to explore whether or not some HCWs might have more than two natural centers of activity. Future work should also focus on incorporating clinical data into HCW and patient modeling and agenda generation. For example, the attractiveness of patient rooms for HCWs should depend on the presence and diagnosis of patients, and the transition probabilities and LOS in patient models should depend on patient diagnosis and staffing levels. Additionally, HCW movement is not likely to actually be a random walk. Future work should explore adding higher level strategy or memory to HCW movement, perhaps through the use of hidden Markov models.

The results of our sample applications themselves also suggest further topics for study. Intensive care recidivism is widely acknowledged to be a problem, but little is known about its causes or prevention. Our work suggests that overloaded ICUs contribute to the problem, and we encourage further exploration along these lines. Our mumps infection control policy simulations suggest that the recent policy recommendation of a 5 day quarantine period rather than 9 days might lead to larger mumps outbreaks. This depends to a large extent on how infectious individuals are 7-11 days after infection. While it is difficult to pinpoint infectious disease parameters, more data on shedding levels after the onset of symptoms would potentially shed more light on this issue. The shedding curve we used to inform our model seems to indicate small, but still significant infectiousness levels after one week. Because

mumps is rarely fatal, this is a question of reducing costs and management burden. While it may well be the case that 5 days is the most cost-effective quarantine period, there seems to be an opportunity for further simulation or econometric analysis.

Beyond the applications highlighted in this work, there are a wealth of opportunities for improving hospital operations and patient care based on a realistic hospital simulator. Reviews of hospital simulator literature by Jun et al. and England point to dozens of applications of a realistic hospital simulator including optimizing bed assignment, sizing facilities, scheduling surgeries and admissions, assisting with medical decision making, etc. [70, 40]. Operational issues are becoming increasingly important particularly in emergency departments across the country, which have seen a nationwide decline in capacity, while simultaneously having to deal with increasing numbers of visitors [34, 8, 35]. Our simulator could also be extended to estimate the effects of increased geriatric patient loads over the next decade, and could be used to test explicit policies for choosing which patients to discharge when ICUs become full [19].

In addition to these more general operational problems discussed in the hospital simulation literature, other potential applications rely more specifically on our fine-grained location data. We present two such examples: (1) Is patient care dependent on the distance from the patient bedroom to the service to which that patient is assigned? Anecdotal evidence suggests that patients far from their physicians tend to be seen less frequently and later in the day than other patients, and may expect to spend more time on average in the hospital. (2) Do adverse events, e.g., patient falls

or medication errors, cluster spatially or temporally? Our models and simulation are particularly suited to addressing such questions after linking the appropriate datasets.

It should also be noted that this research has implications outside of healthcare. Indeed, the work was inspired by a framework developed to characterize the centers and dispersions of search engine queries. In general, this framework seems useful in any resource location problem where the resources are consumed by a population whose locations are uncertain. For example, using this framework one could answer questions such as "Where should the Iowa City Police Department increase its presence to combat the rise in downtown violence?".

5.1.0.4.1 Acknowledgments

We thank Lee Carmen, Amy Liang, Chris Smith, and Joseph Wagner for their assistance with processing UIHC data. We thank Stephanie Holley RN BSN CIC, Tom Persoon, Hendrick Shultz M.D., and Chris Smith for their input as "behavioral consultants." We thank Alicia Gerke M.D. and Keven Doerschug M.D. for taking the time to explain intensive care unit operations. We thank the 1st ACM International Health Informatics Symposium reviewers for their helpful feedback. We would also like to acknowledge Jason Fries, Matt Stults, Matt Hootman, Gaurav Kanade, Shobha Kazinka, Justin Hoffman, James Paton, and Brandyn Kusenda for their help manually building the hospital graph from UIHC CAD printouts.

Support for this research was provided by a University of Iowa College of Medicine Translational Research Pilot Grant (Philip M Polgreen, Sriram V Pemmaraju and Alberto M Segre), by a National Institutes of Health Young Investigator

Award NIH-K01-AI075089 (Philip M Polgreen), and by the National Institutes of Health grant NIH-R21-AI081164 (Alberto M Segre, Sriram V Pemmaraju, and Philip M Polgreen).

APPENDIX SELECTED PROOFS AND DERIVATIONS

Proof Sketch of Non-concavity of Multi-center Log-likelihood Function

(1) We consider a simple 5 room facility with 2 HCWs. (2) We choose not-trivial distances between the rooms. (3) We assume HCW_A has one login in room 1 and one login in room 2, while HCW_B has one login in room 3. (4) We let room 4 and 5 be centers in a multi-center model of HCW_A. (5) We find the log-likelihood function for such a model. (6) We find the Hessian of the log-likelihood function, and the eigenvector of the Hessian. (7) We fix three of the four decay parameters, and run a local search to maximize one of the eigenvalues. (8) We show that one of the eigenvalues is positive for some choice of decay parameters. This implies that the Hessian is not negative semi-definite, which in turn implies that the log-likelihood function is non-concave. While the full output is too long to reproduce in its entirety, the following Mathematica commands will reproduce our results:

```
(* Define the log-likelihood function *)
f[B1_, y1_, B2_, y2_, d11_, d12_, d13_, d21_, d22_, d23_]:=
Log[B1 * d11^ - y1  + B2 * d21^ - y2] + Log[B1 * d12^ - y1 + B2 * d22^ - y2]+
Log[1 - B1 * d13^ - y1 - B2 * d23^ - y2]
(* Assign some (arbitrarily chosen) distances from each center to each room *)
ll = f[B1, y1, B2, y2, 2, 5, 6, 3, 4, 5]
(* Find the Hessian *)
H = D[ll, {{B1, y1, B2, y2}, 2}]
```

```
(* Find the eigenvalues of the Hessian *)

Eig = Eigenvalues[H]

(* Try to find a set a decay parameters such that one eigenvalue is positive.

Note that this local maximization step is not gauranteed

to find such a set of decay parameters. *)

NMaximize[{Extract[Eig/.y1 → 0.1, 3], B1 > 0.1, B1 < 0.9}, {B1}]

NMaximize[{Extract[Eig/.y1 → 0.1, 4], B1 > 0.1, B1 < 0.9}, {B1}]

NMaximize[{Extract[Eig/.y2 → 0.1, 1], B2 > 0.1, B2 < 0.9}, {B2}]

NMaximize[{Extract[Eig/.y2 → 0.1, 2], B2 > 0.1, B2 < 0.9}, {B2}]

NMaximize[{Extract[Eig, 1], B1 > 0.1, B1 < 0.9, y1 > 0.1, y2  > 0.1, B2 > 0.1,

B2 < 0.9}, {B1, y1, B2, y2}]

(* These two sets of decay parameter settings result in two different

eigenvalues taking a psotive value. *)

Eigenvalues[H/.{B1 → 0.9,  y1 → 0.1,  B2 → 0.9, y2 → 0.1}]

Eigenvalues[H/.{B1 → 0.2,  y1 → 2.9,  B2 → 0.5, y2 → 0.3}]
```

REFERENCES

[1] Updated recommendations for isolation of persons with mumps. *The Centers for Disease Control and Prevention Morbidity and Mortality Weekly Report*, 57:1103–1105, October 2008.

[2] J G Anderson. Evaluation in health informatics: computer simulation. *Computers in Biology and Medicine*, 32(3):151–164, 2002.

[3] R Anderson and R May. Infectious diseases of humans. *citeulike.org*, Jan 1991.

[4] J Arnaud. Automated control of internet services. *Proceedings of the Fifth International Workshop on Feedback Control Implementation and Design in Computing Systems and Networks*, pages 7–12, 2010.

[5] V Arya, N Garg, R Khandekar, A Meyerson, K Munagala, and V Pandit. Local search heuristic for k-median and facility location problems. *STOC '01: Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 21–29, Jul 2001.

[6] N H Augustin, M A Mugglestone, and S T Buckland. An autologistic model for the spatial distribution of wildlife. *Journal of Applied Ecology*, 33(2):339–347, 1996.

[7] L Backstrom, J Kleinberg, R Kumar, and J Novak. Spatial variation in search engine queries. *International World Wide Web Conference: Proceeding of the 17th international conference on World Wide Web*, pages 357–366, Jan 2008.

[8] A Bagust. Dynamics of bed use in accommodating emergency admissions: stochastic simulation model. *Bmj*, 319(7203):155, 1999.

[9] S Bansal, B Grenfell, and L Meyers. When individual behaviour matters: homogeneous and network models in epidemiology. *Journal of The Royal Society Interface*, Jan 2007.

[10] T Bardell, J F Legare, K J Buth, G M Hirsch, and I S Ali. Icu readmission after cardiac surgery. *European journal of cardio-thoracic surgery*, 23(3):354–359, 2003.

[11] P Berkhin. A survey of clustering data mining techniques. *Grouping Multidimensional Data*, Jan 2006.

[12] D Bernoulli. An attempt at a new analysis of the mortality caused by smallpox and of the advantages of inoculation to prevent it.

[13] S Blower and D Bernoulli. An attempt at a new analysis of the mortality caused by smallpox and of the advantages of inoculation to prevent it. 1766. *Rev Med Virol*, 14(5):275–88, Jan 2004.

[14] A L Bonebrake. Effects of mumps outbreak in hospital, chicago, illinois, usa, 2006. *Emerging Infectious Diseases*, 16(3):1–7, Mar 2010.

[15] D Brockmann, L Hufnagel, and T Geisel. The scaling laws of human travel. *Nature*, 439(7075):462–465, 2006.

[16] M Brown. Modelling the spatial distribution of suburban crime. *Economic Geography*, Jan 1982.

[17] R Ceglowski, L Churilov, and J Wasserthiel. Combining data mining and discrete event simulation for a value-added view of a hospital emergency department. *Journal of the Operational Research Society*, 58(2):246–254, 2006.

[18] D B Chalfin, S Trzeciak, A Likourezos, B M Baumann, and R P Dellinger. Impact of delayed transfer of critically ill patients from the emergency department to the intensive care unit. *Critical care medicine*, 35(6):1477, 2007.

[19] Carri W Chan. Maximizing throughput of hospital intensive care units with patient readmissions. pages 1–41, Feb 2011.

[20] S Chib and E. Greenberg. Understanding the metropolis hastings algorithm. *American Statistical Journal*, 49:327–335, 1995.

[21] C A Chrusch, K P Olafson, P M McMillan, D E Roberts, and P R Gray. High occupancy increases the risk of early death or readmission after transfer from intensive care. *Critical care medicine*, 37(10):2753, 2009.

[22] L Citrome, L Green, and R Fost. Length of stay and recidivism on a psychiatric intensive care unit. *Hospital & community psychiatry*, Jan 1994.

[23] W Cohn, F Sellke, C Sirois, and A Lisbon. Surgical icu recidivism after cardiac operations. *Chest*, Jan 1999.

[24] M Comas, X Castells, L Hoffmeister, R Román, F Cots, J Mar, S GutiérrezMoreno, and M Espallargues. Discreteevent simulation applied to analysis of waiting lists. evaluation of a prioritization system for cataract surgery. *Value in Health*, 11(7):1203–1213, 2008.

[25] L G Connelly and A E Bair. Discrete event simulation of emergency department activity: A platform for systemlevel operations research. *Academic Emergency Medicine*, 11(11):1177–1185, 2004.

[26] D Cordes, V Haughton, J D Carew, K Arfanakis, and K Maravilla. Hierarchical clustering to measure connectivity in fmri resting-state data. *Magnetic resonance imaging*, 20(4):305–317, 2002.

[27] M Côté and W Stein. An erlang-based stochastic model for patient flow. *Omega*, Jan 2000.

[28] D E Curtis, C S Hlady, G Kanade, P M Polgreen, S Pemmaraju, and A M Segre. Hospital contact networks and the prevention of nosocomial infections. *Working Paper*.

[29] D E Curtis, C S Hlady, S Pemmaraju, P M Polgreen, and A M Segre. Modeling and estimating the spatial distribution of healthcare workers. *Proceedings of the 1st ACM International Health Informatics Symposium*, Nov 2010.

[30] D E Curtis, C S Hlady, S V Pemmaraju, A M Segre, and P M Polgreen. Social network influence on vaccination uptake among healthcare workers. In *5th Decennial International Conference on Healthcare-Associated Infections*, Atlanta, GA, USA, March 2010.

[31] D E Curtis, G Kanade, S V Pemmaraju, P M Polgreen, and A M Segre. Analysis of hospital health-care worker contact networks. In *5th UK Social Networks Conference*, July 2009.

[32] D E Curtis, S Pemmaraju, L A Polgreen, P M Polgreen, and A M Segre. Peer effects and influenza vaccination among healthcare workers. In *3rd Biennial Conference of the American Society of Health Economists*, June 2010.

[33] B Demianyk, D Sandison, B Libbey, R Guderian, RD McLeod, MR Eskicioglu, MR Friesen, K Ferens, and S Mukhi. Technologies to generate contact graphs for personal social networks. *e-Health Networking Applications and Services (Healthcom), 2010 12th IEEE International Conference on*, pages 15–22.

[34] R W Derlet and J R Richards. Overcrowding in the nation's emergency departments: complex causes and disturbing effects. *Annals of emergency medicine*, 35(1):63–68, 2000.

[35] M A Draeger. An emergency department simulation model used to evaluate alternative nurse staffing and patient population scenarios. *Proceedings of the 24th conference on Winter simulation*, pages 1057–1064, 1992.

[36] C Duguay and F Chetouane. Modeling and improving emergency department systems using discrete event simulation. *Simulation*, 83(4):311, 2007.

[37] T C Edwards, Jr, E T Deshler, D Foster, and G G Moisen. Adequacy of wildlife habitat relation models for estimating spatial distributions of terrestrial vertebrates. *Conservation Biology*, 10(1):263–270, 1996.

[38] E El-Darzi, C Vasilakis, T Chaussalet, and P H Millard. A simulation modelling approach to evaluating length of stay, occupancy, emptiness and bed blocking in a hospital geriatric department. *Health care management science*, 1(2):143–149, 1998.

[39] M Elliott. Readmission to intensive care: a review of the literature. *Australian Critical Care*, Jan 2006.

[40] W England and S D Roberts. Applications of computer simulation in health care. *Proceedings of the 10th conference on Winter simulation-Volume 2*, pages 665–677, 1978.

[41] A Esogbue and A Singh. A stochastic model for an optimal priority bed distribution problem in a hospital ward. *Operations Research*, Jan 1976.

[42] M Ester, H Kriegel, J Sander, and X Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proc. KDD*, Jan 1996.

[43] T Evans et al. The world health report 2004. 2004.

[44] S Eubank, H Guclu, VS Anil Kumar, M V Marathe, A Srinivasan, Z Toroczkai, and N Wang. Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988):180–184, 2004.

[45] L K Fitzpatrick, J A Hardacker, W Heirendt, T Agerton, A Streicher, H Melnyk, R Ridzon, S Valway, and I Onorato. A preventable outbreak of tuberculosis investigated through an intricate social network. *Clinical Infectious Diseases*, pages 1801–1806, 2001.

[46] U Frick, J Rehm, S Krischker, and C Cording. Length of stay in a german psychiatric hospital as a function of patient and organizational characteristics–a multilevel analysis. *International Journal of Methods in Psychiatric Research*, 8(3):146–161, 1999.

[47] D Griffith. Modelling urban population density in a multi-centered city. *Journal of urban economics*, Jan 1981.

[48] P J Gruenewald, B Freisthler, L Remer, E A LaScala, and A Treno. Ecological models of alcohol outlets and violent assaults: crime potentials and geospatial analysis. *ADDICTION-ABINGDON-*, 101(5):666, 2006.

[49] S Guha and S Khuller. Greedy strikes back: Improved facility location algorithms. *Journal of Algorithms*, 31(1):228–248, 1999.

[50] R Haley, D Culver, and J White. The nationwide nosocomial infection rate. *American journal of epidemiology*, Jan 1985.

[51] J Han, M Kamber, and A Tung. Spatial clustering methods in data mining: A survey. *Geographic Data Mining and Knowledge Discovery*, Jan 2001.

[52] A Haskose, BG Kingsman, and D Worthington. Modelling flow and jobbing shops as a queueing network for workload control. *International Journal of Production Economics*, 78(3):271–285, 2002.

[53] Y Heo, R Choudhary, S Bafna, A Hendrich, and M P Chow. A modeling approach for estimating the impact of spatial configuration on nurses' movement. *Proceedings of the 7th International Space Syntax Symposium*, 2009.

[54] H Hethcote. The mathematics of infectious diseases. *SIAM REVIEW*, Jan 2000.

[55] H Hethcote, M Zhien, and L Shengbing. Effects of quarantine in six endemic models for infectious diseases. *Mathematical biosciences*, 180(1-2):141–160, 2002.

[56] C S Hlady, D E Curtis, M A Severson, J Fries, S Pemmaraju, A M Segre, T Herman, and P M Polgreen. A near-real-time method for discovering healthcare worker social networks via wireless devices. In *47th Annual Meeting of the Infectious Disease Society of America*, October 2009.

[57] C S Hlady, T Tassier, A M Segre, T Herman, S V Pemmaraju, and P M Polgreen. Comparing the length of isolation periods to prevent the nosocomial spread of mumps (poster). *International Meeting on Emerging Diseases and Surveillance*, Feb 2009.

[58] D Hochbaum and D Shmoys. A unified approach to approximation algorithms for bottleneck problems. *Journal of the Association for Computing Machinery*, 33:533–550, Jan 1986.

[59] N R Hoot, L J LeBlanc, I Jones, S R Levin, C Zhou, C S Gadd, and D Aron-

sky. Forecasting emergency department crowding: a discrete event simulation. *Annals of emergency medicine*, 52(2):116–125, 2008.

[60] T Hornbeck, D E Curtis, T Herman, G Thomas, A M Segre, and P M Polgreen. Contact patterns for hcws: Not everyone is the average. In *21st Annual Scientific Meeting of the Society for Healthcare Epidemiology of America*, Atlanta, GA, USA, April 2011.

[61] L Isella, M Romano, A Barrat, C Cattuto, V Colizza, W Van den Broeck, F Gesualdo, E Pandolfi, L Ravà, and C Rizzo. Close encounters in a pediatric ward: measuring face-to-face proximity and mixing patterns with wearable sensors. *PloS one*, 6(2):e17144, 2011.

[62] M Isken and B Rajagopalan. Data mining to support simulation modeling of patient flow in hospitals. *Journal of medical systems*, Jan 2002.

[63] S B Issenberg, W C McGaghie, I R Hart, J W Mayer, J M Felner, E R Petrusa, R A Waugh, D D Brown, R R Safford, and I H Gessner. Simulation technology for health care professional skills training and assessment. *JAMA: the journal of the American Medical Association*, 282(9):861, 1999.

[64] T J Iwashyna, A A Kramer, and J M Kahn. Intensive care unit occupancy and patient outcomes. *Critical care medicine*, 37(5):1545, 2009.

[65] S H Jacobson, S N Hall, and J R Swisher. Discrete-event simulation of health care systems. *Patient Flow: Reducing Delay in Healthcare Delivery*, pages 211–252, 2006.

[66] A K Jha, D Doolan, D Grandt, T Scott, and D W Bates. The use of health information technology in seven nations. *International Journal of Medical Informatics*, 77(12):848–854, 2008.

[67] A K Jha, T G Ferris, K Donelan, C DesRoches, A Shields, S Rosenbaum, and D Blumenthal. How common are electronic health records in the united states? a summary of the evidence. *Health Affairs*, 25(6):w496, 2006.

[68] S Johnson. Hierarchical clustering schemes. *Psychometrika*, Jan 1967.

[69] E Jones, T Oliphant, P Peterson, et al. SciPy: Open source scientific tools for Python, 2001–.

[70] J B Jun, S H Jacobson, and J R Swisher. Application of discrete-event simulation in health care clinics: a survey. *Journal of the Operational Research Society*, 50(2):109–123, 1999.

[71] G Karypis, E Han, and V Kumar. Chameleon: Hierarchical clustering using dynamic modeling. *Computer*, Jan 2002.

[72] W O Kermack and A G McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 115:700–721, Dec 1927.

[73] R M Klevens, J R Edwards, C L Richards, T C Horan, R P Gaynes, D A Pollock, and D M Cardo. Estimating health care-associated infections and deaths in us hospitals, 2002. *Public Health Reports*, 122(2):160, 2007.

[74] A Kogan, J Cohen, and E Raanani. Readmission to the intensive care unit after" fast-track" cardiac surgery: risk factors and outcomes. *The Annals of Thoracic Surgery*, Jan 2003.

[75] A Komashie and A Mousavi. Modeling emergency departments using discrete event simulation techniques. *Proceedings of the 37th conference on Winter simulation*, pages 2681–2685, 2005.

[76] A P Kumar and R Kapur. Discrete simulation application-scheduling staff for the emergency room. *Proceedings of the 21st conference on Winter simulation*, pages 1112–1120, 1989.

[77] A Marell, J P Ball, and A Hofgaard. Foraging and movement paths of female reindeer: insights from fractal analysis, correlated random walks, and lévy flights. *Canadian Journal of Zoology*, 80(5):854–865, 2002.

[78] M Marin, P Quinlisk, T Shimabukuro, C Sawhney, C Brown, and CW LeBaron. Mumps vaccination coverage and vaccine effectiveness in a large outbreak among college students–iowa, 2006. *Vaccine*, 26(29-30):3601–3607, 2008.

[79] A Marshall, C Vasilakis, and E El-Darzi. Length of stay-based patient flow models: recent developments and future directions. *Health care management science*, Jan 2005.

[80] M B Meyer. An epidemiologic study of mumps; its spread in schools and families. *American Journal of Epidemiology*, 75(2):259, 1962.

[81] L Meyers. Contact network epidemiology: Bond percolation applied to infectious disease prediction and control. *Bulletin: American Mathematical Society*, 44:63–86, Jan 2007.

[82] L Meyers, M Newman, and M Martin. Applying network theory to epidemics:

control measures for mycoplasma pneumoniae outbreaks. *Emerging Infectious Diseases*, Jan 2003.

[83] D Miles. Crowds in a polygon soup: Next-gen path planning. In *20th Game Developers Conference*, March 2006.

[84] H Mo and S White. An analytic model for the spatial clustering of dark matter haloes. *Arxiv preprint astro-ph*, Jan 1995.

[85] J A Nelder and R Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

[86] M Newman. Spread of epidemic disease on networks. *Physical Review E*, Jan 2002.

[87] M E J Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[88] G Nishi, R Suh, M Wilson, and S Cunneen. Analysis of causes and prevention of early readmission to surgical intensive care. *The American Surgeon*, Jan 2003.

[89] O Olguin, P A Gloor, and A Pentland. Wearable sensors for pervasive healthcare management. *Pervasive Computing Technologies for Healthcare, 2009. PervasiveHealth 2009. 3rd International Conference on*, pages 1–4, 2009.

[90] World Health Organization. Summary of probable sars cases with onset of illness from 1 november 2002 to 31 july 2003. http://www.who.int/csr/sars/country/table$2004_0 4_2 1$/*en/index.html*.

[91] S H Owen and M S Daskin. Strategic facility location: A review. *European Journal of Operational Research*, 111(3):423–447, 1998.

[92] J Peiris, Y Guan, and K Y Yuen. Severe acute respiratory syndrome. *Nature medicine*, Jan 2004.

[93] R N Philip, K R Reinhard, and D B Lackman. Observations on a mumps epidemic in a "virgin" population. 1958. *American journal of epidemiology*, 142(3):233, 1995.

[94] P M Polgreen, L C Bohnett, J E Cavanaugh, S B Gingerich, L E Desjardin, M L Harris, M P Quinlisk, and M A Pentella. The duration of mumps virus shedding after the onset of symptoms. *Clinical Infectious Diseases*, 46(9):1447–1449, May 2008.

[95] P M Polgreen, C S Hlady, M A Severson, A M Segre, and T Herman. A wireless and portable system for monitoring hand hygiene compliance. in press.

[96] P M Polgreen, T Tassier, S Pemmaraju, and A M Segre. Using social networks to prioritize vaccination strategies for healthcare workers. *Infection Control and Hospital Epidemiology*, 31:893–900, September 2010.

[97] RG Priest, N Fineberg, S Merson, and T Kurian. Length of stay of acute psychiatric inpatients: an exponential model. *Acta Psychiatrica Scandinavica*, 92(4):315–317, 1995.

[98] P Punnakitikashem, J M Rosenberger, and D Buckley Behan. Stochastic programming for nurse assignment. *Computational Optimization and Applications*, 40(3):321–349, 2008.

[99] M Raunak, L Osterweil, A Wise, L Clarke, and P Henneman. Simulating patient flow through an emergency department using process-driven discrete event simulation. *Proceedings of the 2009 ICSE Workshop on Software Engineering in Health Care*, pages 73–83, 2009.

[100] C Richardson. Stochastic simulation of daily precipitation, temperature, and solar radiation. *Water Resources Research*, Jan 1981.

[101] U E Ruttimann and M M Pollack. Variability in duration of stay in pediatric intensive care units: a multiinstitutional study. *The Journal of pediatrics*, 128(1):35–44, 1996.

[102] P Sartwell. The incubation period and the dynamics of infectious disease. *American Journal of Epidemiology*, 83(2):204, 1966.

[103] D Schilling and V Jayaraman. A review of covering problems in facility location. *fisher.osu.edu*, Jan 1992.

[104] A A Sharov, A M Liebhold, and E A Roberts. Spatial variation among counts of gypsy moths (lepidoptera: Lymantriidae) in pheromone-baited traps at the expanding front. *Environmental Entomology*, 25:1312–1320, 1997.

[105] D B Shmoys, É Tardos, and K Aardal. Approximation algorithms for facility location problems (extended abstract). *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 265–274, 1997.

[106] R Simpson. Infectiousness of communicable diseases in the household (measles, chickenpox, and mumps). *Lancet*, Jan 1952.

[107] J E Stahl, D Rattner, R Wiklund, J Lester, M Beinfeld, and G S Gazelle. Reorganizing the system of care surrounding laparoscopic surgery: a cost-effectiveness analysis using discrete-event simulation. *Medical decision making*, 24(5):461, 2004.

[108] M J Strauss, J P LoGerfo, J A Yeltatzie, N Temkin, and L D Hudson. Rationing of intensive care unit services: an everyday occurrence. *Survey of Anesthesiology*, 31(1):38, 1987.

[109] J M Swaminathan, S F Smith, and N M Sadeh. Modeling supply chain dynamics: A multiagent approach. *Decision Sciences*, 29(3):607–632, 1998.

[110] J R Swisher and S H Jacobson. Evaluating the design of a family practice healthcare clinic using discrete-event simulation. *Health care management science*, 5(2):75–88, 2002.

[111] T R Talbot, S F Bradley, S E Cosgrove, C Ruef, J D Siegel, and D J Weber. Influenza vaccination of healthcare workers and vaccine allocation for healthcare workers during vaccine shortages. *Infection Control and Hospital Epidemiology*, 26(11):882–890, 2005.

[112] G Taylor and S McClean. Stochastic models of geriatric patient bed occupancy behaviour. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, Jan 2000.

[113] L Temime, L Kardas-Sloma, L Opatowski, C Brun-Buisson, P Y Boëlle, and D Guillemot. Nososim: an agent-based model of nosocomial pathogens circulation in hospitals. *Procedia Computer Science*, 1(1):2245–2252, 2010.

[114] L Temime, L Opatowski, Y Pannet, C Brun-Buisson, P Y Boëlle, and D Guillemot. Peripatetic health-care workers as potential superspreaders. *Proceedings of the National Academy of Sciences*, 106(43):18420, 2009.

[115] C Toregas, R Swain, C ReVelle, and L Bergman. The location of emergency service facilities. *Operations Research*, 19(6):1363–1373, 1971.

[116] T Ueno and N Masuda. Controlling nosocomial infection based on structure of hospital social networks. *Journal of Theoretical Biology*, Jan 2008.

[117] D J Watts and S H Strogatz. Collective dynamics of'small-world' networks. *Nature*, 393:440–442, Jan 1998.

[118] C Weissman. Analyzing intensive care unit length of stay data: Problems and possible solutions. *Critical care medicine*, Jan 1997.

[119] E West, D Barron, J Dowsett, and J Newton. Hierarchies and cliques in the social networks of health care professionals: implications for the design of dissemination strategies. *Social Science & Medicine*, Jan 1999.

[120] I Yu, T Wong, Y Chiu, N Lee, and Y Li. Temporal-spatial analysis of severe acute respiratory syndrome among hospital inpatients. *hub.hku.hk*.

[121] Y Zhao, G Karypis, and U Fayyad. Hierarchical clustering algorithms for document datasets. *Data Mining and Knowledge Discovery*, 10(2):141–168, 2005.