

Dartmouth College

Dartmouth Digital Commons

Dartmouth College Undergraduate Theses

Theses, Dissertations, and Graduate Essays

Spring 6-1-2021

The Discrete-Event Modeling of Administrative Claims (DEMAC) System: Dynamically modeling the U.S. healthcare delivery system with Medicare claims data to improve end-of-life care

Rachael Chacko

Rachael.S.Chacko.21@Dartmouth.edu

Follow this and additional works at: https://digitalcommons.dartmouth.edu/senior_theses



Part of the [Computer Sciences Commons](#), [Health and Medical Administration Commons](#), [Health Services Administration Commons](#), [Health Services Research Commons](#), and the [Systems Engineering Commons](#)

Recommended Citation

Chacko, Rachael, "The Discrete-Event Modeling of Administrative Claims (DEMAC) System: Dynamically modeling the U.S. healthcare delivery system with Medicare claims data to improve end-of-life care" (2021). *Dartmouth College Undergraduate Theses*. 232.
https://digitalcommons.dartmouth.edu/senior_theses/232

This Thesis (Undergraduate) is brought to you for free and open access by the Theses, Dissertations, and Graduate Essays at Dartmouth Digital Commons. It has been accepted for inclusion in Dartmouth College Undergraduate Theses by an authorized administrator of Dartmouth Digital Commons. For more information, please contact dartmouthdigitalcommons@groups.dartmouth.edu.

**The Discrete-Event Modeling of Administrative
Claims (DEMAC) System: Dynamically
modeling the U.S. healthcare delivery system
with Medicare claims data to improve
end-of-life care**

A thesis submitted to the faculty in partial fulfillment of
the requirements for the degree of Bachelor of Arts in
Computer Science

Rachael Chacko

Advisor: Inas Khayal, Ph.D.

Department of Computer Science

Dartmouth College

June 1, 2021

Contents

1	Abstract	2
2	Introduction	2
3	System Structure	5
3.1	UserSettings Class	6
3.2	Patients Class	8
3.3	Determining Form and Function	10
3.4	HealthcareSystem Class	11
4	System Output	13
4.1	Patient-Level Visualizations	14
4.2	Hospital-Level Visualizations	20
4.3	Hetero-Functional Graph Theory Toolbox Files	24
5	Discussion and Future Work	25
6	Code and Data Availability	26
7	Acknowledgments	26

1 Abstract

The shift of the U.S. healthcare delivery system from the treatment of acute conditions to chronic diseases requires a new method of healthcare system analysis to properly assess end-of-life (EOL) quality throughout the country. In this paper, we propose the Discrete-Event Modeling of Administrative Claims (DEMAC) system, which relies on a hetero-functional graph theory and discrete event-driven framework to dynamically model EOL care on multiple levels. The heat map visualizations produced by the DEMAC system enable the elucidation of not only patient-specific EOL care but also broader treatment patterns among providers and hospitals. As a whole, the DEMAC system provides visual insight into the “black box” of the U.S. healthcare delivery system that can help clinicians and hospital administrators learn where and how to improve EOL care within their institutions.

2 Introduction

Over the years, the American healthcare delivery system has evolved to incorporate numerous technological advancements designed to improve patient care, but the delivery of these innovations has also served to highlight the disparities among different institutions and patient populations. Several studies have demonstrated the variation in the quality of patient care across the U.S., particularly in end-of-life (EOL) care [1–4]. End-of-life (EOL) care is defined as the methods of treatment provided to patients near the end of their life who are often in an advanced stage of a terminal illness such as advanced cancer [5]. EOL treatments can range from palliative care at home that addresses the patient’s pain levels and well-being to life support measures that serve to extend the patient’s life, such as the use of a breathing machine. High quality EOL care is patient-centered and typically prioritizes the patient’s physical and mental comfort over extending their life with aggressive treatments such as chemotherapy and ventilator support. Such high-quality EOL care can be achieved through early discussion between the patient and physician as well as advance care planning, which establishes the patient’s directives for EOL care, such as do-not-resuscitate orders [5, 6]. Proper advance care planning has been shown to increase patients’ quality of EOL care by reducing hospitalization

and life-sustaining therapies and increasing the use of hospice and palliative care [6].

Although the methods of providing high-quality EOL care for patients have been rigorously established, several studies have demonstrated that disparities in EOL care are still visible across groups of people depending on their geographic location, socioeconomic status, race, ethnicity, and gender – factors that are also known as the social determinants of health [2, 7–10]. For instance, black patients with prostate cancer are more likely to receive high-intensity EOL care, such as more inpatient or intensive care unit (ICU) admissions or cardiopulmonary resuscitation, compared to white patients, indicating that a patient’s race, among many other factors, can have an impact on the quality of EOL care that they receive [1]. The rapidly aging U.S. population also provides an impetus to better understand the current state of EOL care. Although the elderly comprise approximately 11% of the American population, their medical care accounts for nearly one third of U.S. healthcare spending [11]. Therefore, addressing differences in the quality of EOL treatment is of utmost importance and requires a fundamental comprehension of the structure and behavior of the current U.S. healthcare delivery system.

Current methods of measuring healthcare quality across different institutions involve the application of basic descriptive metrics to describe large groups of patients [12]. However, this type of methodology provides clinicians with extensive amounts of generalized statistics instead of feasible solutions or actionable information for improving EOL care at their specific institution. Furthermore, these statistical techniques often represent healthcare delivery systems as static, rather than dynamic systems that provide different types of care over time.

Such aggregated statistical techniques were effective when patients were primarily afflicted with acute infectious diseases that required brief treatments to resolve. The shift in the predominance of acute conditions to chronic diseases over the last century resulted in multiple, lengthier hospital stays and longitudinal care delivery for more patients [13]. Although nearly 45% of the U.S. population suffer from at least one chronic disease, the healthcare system and the current methods that are used to analyze it are still fundamentally structured to treat acute, rather than chronic, diseases [14]. Thus, current statistical methods inadequately describe patients as static entities rather than dynamic beings with evolving health conditions.

With today’s static, generalized methods of healthcare system analysis, the trajectory of an individual patient’s EOL care and the care patterns of providers within a hospital are difficult

to clearly discern over a period of time. The inability to pinpoint the causes of poor EOL care within and across hospitals leads to the perpetuation of the disparities described above. With dynamic systems modeling, we can visualize a patient’s trajectory or a hospital’s resource distribution over an extended period of time rather than at one point.

To develop a dynamic method of healthcare delivery system modeling, we can leverage principles from hetero-functional graph theory (HFGT), which is used to describe the system structure, as well as discrete event dynamics with timed Petri nets [15]. HFGT models a system in terms of its form, which describes the possible resources, and its function, which describes the possible processes. HFGT allows for the distinction of different functions from one another and therefore the incorporation of multiple functions for one resource. HFGT has previously been used to analyze large systems such as smart cities and the electric power grid, and this theory can also be applied to model the healthcare delivery system [16–19]. For instance, a hospital’s form can consist of three resources: an outpatient department, an inpatient wing, and an ICU. Each resource is capable of performing several processes. As an example, the outpatient resource can support measurement and transportation processes but cannot support life-sustaining treatments, which is a process that only the ICU and inpatient resources are capable of performing. Another tool that is useful for modeling dynamic systems is timed Petri nets. Timed Petri nets permit the modeling of discrete events in which a transition fires to represent an event occurring [18, 19]. These Petri nets can model multiple events occurring at once, which is especially appropriate when comparing behavior across a hospital where several care events can occur simultaneously. With the implementation of both HFGT and discrete-event modeling concepts, we can dynamically model the healthcare delivery system to visualize a patient’s EOL care over an extended period of time.

In this paper, we propose the Discrete-Event Modeling of Administrative Claims (DEMAC) system, which is a novel tool that incorporates a systems engineering and hetero-functional graph theory framework to dynamically model the healthcare system. This project entails modeling the U.S. healthcare delivery system using Medicare administrative claims data to gain a better understanding of the distinctive patterns of EOL care for patients with poor prognosis diseases at different institutions throughout the country. The DEMAC system consists of a pipeline that takes raw claims data as an input, models the system form and function, and

produces visual output files that can provide clinicians and healthcare administrators with an insight into EOL care at their hospital. This system produces multilevel heat map visualizations that allow for the examination of EOL resource utilization for individual patients as well as care patterns across physicians and hospitals. The DEMAC system also produces system-level XML and scheduled event list files that are used as input for the HFGT Toolbox. The HFGT Toolbox implements principles from HFGT and discrete-event dynamics as described above to create simulations that visualize the discrete claim events from the data [20]. As a whole, the DEMAC system provides insight into the “black box” of the U.S. healthcare delivery system to produce behavioral signals that help clinicians and hospital administrators learn where and how to improve EOL care within their institutions.

3 System Structure

The DEMAC system is object-oriented and implemented in Python 3.7. Figure 1 depicts the functional components and general workflow of the system in detail. The three main components of the DEMAC system are the UserSettings class, which stores the file paths and relevant column names for each dataset; the Patients class, which loads the beneficiary data and creates patient- and hospital-level utilization files; and the HealthcareSystem class, which produces the heat map visualizations, utilization matrices, and HFGT Toolbox files from each utilization.

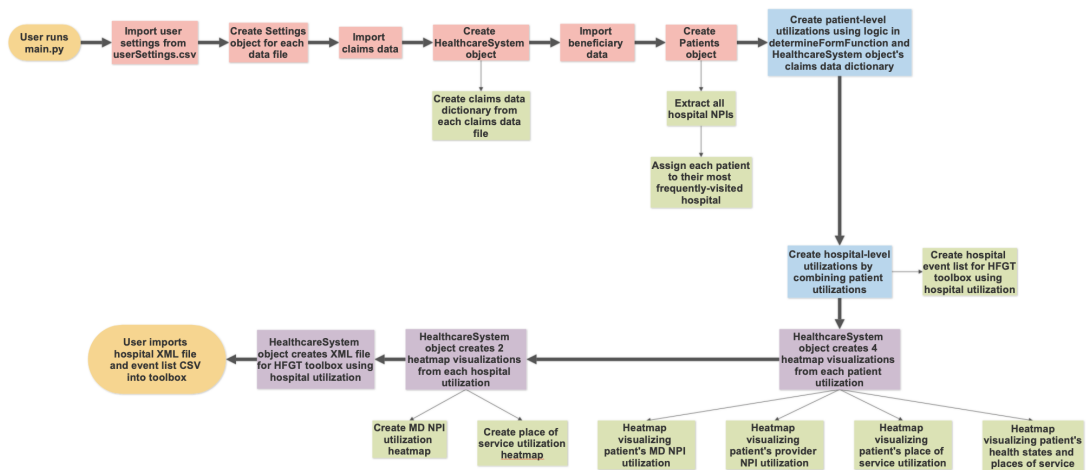


Figure 1: The functional components and workflow of the DEMAC system.

3.1 UserSettings Class

The UserSettings component enables the general usability of the DEMAC system across various public and private claims data files as well as flexibility for the user to customize certain aspects of the analysis. The UserSettings component consists of a comma-separated values (CSV) template file that is completed by the user and a UserSettings class that instantiates an object containing the information from the CSV file. The DEMAC system currently supports the analysis of several Medicare files, including Medicare Provider Analysis and Review (MedPAR)/Part A, carrier/Part B, Inpatient, and Hospice claims data files. However, the UserSettings file is designed to be used with any other administrative claims data files.

The user first enters relevant information into the UserSettings CSV file required for the analysis. Such information includes the types of claims data files for analysis (such as Part A, Part B, or Hospice), the location of each claims data file, and the names of relevant fields for each file. A preview of the UserSettings CSV file is shown in Figure 2. The user can choose whether the service place level heat maps will be plotted with all the possible form-function possibilities or solely the combinations that are present in the data. The UserSettings file also permits the user to bypass the initial patient data pre-processing step by specifying the location of an existing, processed patient data file for import into the system.

After the user completes the UserSettings CSV file, a UserSettings object is instantiated, which stores each item of information from the UserSettings file into instance variables. Figure 3 demonstrates the UserSettings class diagram [21]. The claimTypesList variable stores a list of the names of claim files (e.g., “Part A”, “Hospice”, etc.) that were specified by the user for analysis. The servicePlaceDecision, characteristicsDecision, and characteristicsPath variables store the user’s choices for the two analytical options as described above. Each of the remaining instance variables is itself an object instantiated with information specific to one claims data file, in line with the system’s object-oriented design. For instance, the partBDataSettings object, instantiated from the PartBSettings class, stores as instance variables the file path of the Part B dataset and each of the relevant column names for the Part B file as specified by the user in the UserSettings CSV file. The class diagram for the PartBSettings class is depicted in Figure 4 [21]. Each of the other dataset-specific classes are structured similarly based on the column names

that they contain. This design facilitates the straightforward, generalized importing of data files and user-specific analytic options within the Patients and HealthcareSystem components at each stage of the analysis and visualization.

```

Variable,Header Name
Input Dataset Types (ex. Part A&Part B),Part A&Part B&Inpatient&Hospice
Plot with all possible PLCSRVC or just those present in data? (1 or 2),1
Use existing patient data? (Y/N),Y
Patient Characteristics Data Path,..../idata/exportedPatientData.csv
Patient Dataset Path,..../1-Data/patients.csv
Patient ID,BENE_ID
Patient State Code,State
Patient Zip Code,ZIPCODE
Patient HRR,HRR
Patient Birth/Death Date Format,%Y%m%d
Patient Birth Date,SDOB
Patient Death Date,SDOO
Patient Sex,SEX
Patient Race,RTI_RACE
Part B Dataset Path,..../1-Data/partB.csv
Part B Patient ID,BENE_ID
Part B Claim ID,CLM_ID
Part B From/Thru Date Format,%Y%m%d
Part B Claim From Date,SFROMDT
Part B Claim Thru Date,STHRUDT
Part B Provider Number,PRFNPI
Part B MD NPI,MDNPI
Part B Short Stay/Long Stay/SNF,SSLSSNF
Part B Allowed Charge,ALOWCHRG
Part B ICD Diagnosis Code 1,DGNS_CD1
Part B ICD Diagnosis Code 2,DGNS_CD2
Part B ICD Diagnosis Code 3,DGNS_CD3
Part B ICD Diagnosis Code 4,DGNS_CD4
Part B ICD Diagnosis Code 5,DGNS_CD5
Part B ICD Diagnosis Code 6,DGNS_CD6
Part B ICD Diagnosis Code 7,DGNS_CD7
Part B ICD Diagnosis Code 8,DGNS_CD8
Part B ICD Diagnosis Code 9,DGNS_CD9
Part B ICD Diagnosis Code 10,DGNS_CD10
Part B ICD Diagnosis Code 11,DGNS_CD11
Part B ICD Diagnosis Code 12,DGNS_CD12

```

Figure 2: A preview of the UserSettings file.

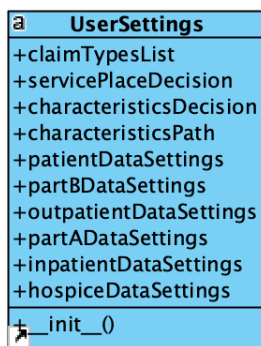


Figure 3: UserSettings class diagram.

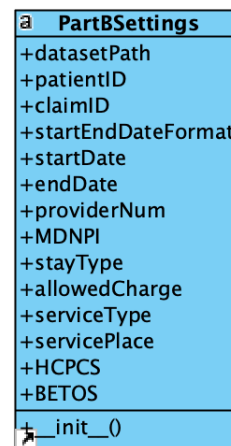


Figure 4: PartBSettings class diagram.

3.2 Patients Class

The Patients class serves to house the patient characteristics data and create the utilization files that describe a patient’s trajectory of care over the last 200 days of their life. The structure of the Patients class is visualized in the class diagram in Figure 5 [21].

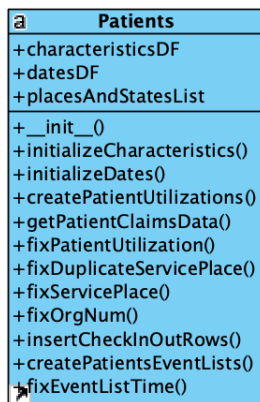
In the method `initializeCharacteristics`, the patient characteristics data is retrieved from the beneficiary file specified in the `UserSettings` file and stored in the `characteristicsDF` Pandas DataFrame instance variable. The imported characteristics data include the patient’s ID, location, birthdate, death date, sex, and race. The high-level location data include the patient’s state and zip code of residence as well as the hospital referral region (HRR), which is an indicator of the geographic area in which the patient receives their care. The HRR variable also provides insight into the rurality of the patient’s location [22]. This method also assigns each patient to their most frequently visited hospital in order to create the hospital-level visualizations later in the analytical process. The method `initializeDates` compiles the patient death dates from `characteristicsDF`, calculates the date 200 days before death that will be used as the first day of the visualizations, and saves this data within the `datesDF` DataFrame.

After the beneficiary data is imported, the Patients class method `createPatientUtilizations` condenses the patient’s claims data across each of the input claims files into one patient utilization DataFrame. Figure 6 displays the first few lines of the patient utilization of a synthetic patient with ID 101. Each line contains the patient ID, the day number of the last 200 days of life on which the claim event occurred, the data file from which the claim data was retrieved, the function of the claim event, and the form of the claim event. The form consists of the individual provider’s NPI, the hospital’s NPI, and the place of service. As an example, the first line of the sample utilization file in Figure 6 refers to a Measurement event on day 10 by provider 993 occurring in the Outpatient department of hospital 123456 that was retrieved from the Part B file. The method for determining a claim’s form and function differs based on the claims data file from which the claim originates and is discussed in more detail in the Determining System Form and Function section.

After constructing the patient utilization, the `fixPatientUtilization` method applies several fixes to the utilization DataFrame using the static helper methods `fixDuplicateServicePlace`,

fixServicePlace, and fixOrgNum to account for missing values for Hospice and Part B claims as well as duplicate claims from Part A and Inpatient files. These fixes are required to ensure that the utilization file is complete and concise before its data can be used in the analysis and visualization steps. For instance, if a patient had an ED or ICU stay during their last 200 days of life, a claim for this event would be present in the Part A and Inpatient claims files, and duplicate claims for this event would be present in the resulting utilization. To correct this issue, the fixPatientUtilization method scans the patient’s utilization DataFrame to remove the duplicate Part A rows for these events. After the utilization is complete, it is exported as a CSV file.

The Patients class method createPatientsEventLists constructs patient-level and hospital-level scheduled event lists from each of the utilization files. These event lists are fed into the HFGT Toolbox to visualize the discrete event dynamics present in the data using a Petri net [20]. The method createPatientsEventLists assigns a unique numerical index to each resource and process that are used to represent the resource-process combinations used on each day. In order to visualize more than one event on one day, the fixEventListTime method re-calculates the timing of each event firing based on how many events are present that day. The insertCheckInOutRows method adds to the event list check-in and check-out rows between resources that are required for transitions between resources. More detailed information about the event list’s structure is given in the Toolbox documentation [20].



```

patientID,dayNumber,claimType,function,MDID,orgNum,PLCSRVC
101,10,Part B,Measure,993,123456,Outp
101,10,Part B,OtherDecis,993,123456,Outp
101,15,Part B,OtherTreat,993,123456,Outp
101,15,Part B,Measure,993,123456,Outp
101,18,Part B,Measure,993,123456,Outp
101,20,Part B,PallCareDecis,24,123456,Outp
101,45,Part A,Measure,5567,123456,ED
101,60,Part A,OtherTreat,5567,123456,ED
101,97,Part A,OtherTreat,3422,123456,Inp
101,97,Part A,Transp,3422,123456,Inp
101,98,Part A,OtherTreat,3422,123456,Inp
101,98,Part A,Transp,3422,123456,Inp
101,99,Part A,Transp,3422,123456,Inp
101,100,Part B,OtherDecis,993,123456,Outp
  
```

Figure 5: Patients class diagram. Figure 6: A preview of the utilization file for patient ID 101.

3.3 Determining Form and Function

Each claims data file contains different information and therefore requires distinct methods for the determination of system form and function for each claim. The Patients class method `createPatientUtilizations` relies on the file `determineFormFunction`, which houses the dataset-specific functions that determine system form and function. The six options for form, at the highest level, are Home, Residential Facility, Outpatient, ED, Inpatient, and ICU. The nine possible functions are modeled as Transportation, Measurement, Palliative Care Decision, Advance Care Planning Decision, Other Decision, Hospice Treatment, Chemotherapy, Life-Sustaining Therapy, and Other Treatment. The following paragraphs detail the Medicare administrative claims datasets that are used to produce the visualizations, but the DEMAC system can also be used with other types of administrative claims data.

The MedPAR, or Part A, claims file contains information regarding Inpatient and Skilled Nursing Facility (SNF) stays [23]. To determine form, the Short Stay/Long Stay/SNF Indicator code provides information as to whether the claim event occurred at a Residential Facility, and the Provider Number field is used to distinguish whether the claim event occurred at Outpatient, a Residential Facility, or Inpatient [24, 25]. The function is determined from the International Classification of Diseases (ICD) diagnosis and procedure codes present in the data. More specifically, the diagnosis ICD code Z515 indicates that a Palliative Care Decision occurred, and several procedure ICD codes specify whether Measurement, Chemotherapy, Life-Sustaining Therapy, or another type of treatment occurred [26]. A Transportation event is also assigned to each day to represent the complete duration of the patient’s inpatient stay.

The carrier, or Part B, claims file contains information regarding fee-for-service claims that were submitted by either individual or organizational medical providers [27]. In our analysis, Part B claims serve to elucidate whether a terminal patient met with their provider to discuss palliative care options or advance care planning, which indicates high-quality EOL care. To determine form, the Place of Service code is used to distinguish between claim events occurring at Home, a Residential Facility, Outpatient, the ED, or Inpatient [28]. To determine each claim’s function, the diagnosis ICD codes, a Healthcare Common Procedure Coding System (HCPCS) code, and a Berenson-Eggers Type of Service (BETOS) code are analyzed. As in the Part A file,

the diagnosis ICD code of Z515 indicates that a Palliative Care Decision occurred. The HCPCS code specifies whether an Advance Care Planning Decision was made or a treatment such as Chemotherapy or Life-Sustaining Therapy was administered [29]. The BETOS code indicates whether Transportation, Measurement, Decision, Chemotherapy, or another treatment occurred during the visit [30].

The Inpatient claims file contains data from claim events occurring in inpatient facilities, the ED, or the ICU [31]. The form is determined using the Revenue Center code, which indicates whether the event occurred at the ED, Inpatient, or ICU, and the Admission Type code, which indicates whether the event occurred at the ED [32, 33].

The Hospice claims file contains data from claims submitted by hospice providers [34]. Form is determined using the combination of two variables: the Facility Type code and the Service Classification Type code. A Facility Type code of 8 indicates that a hospice event occurred [35]. A Service Classification Type code of 1 indicates that the hospice event occurred at home, and a Service Classification Type code of 2 indicates that the hospice event occurred at a hospital [36]. Since the latter form classification value is not specific enough to distinguish the patient’s form between the three possible options of Inpatient, the ED, and the ICU, the `fixPatientUtilization` method examines the entirety of the utilization after it is constructed to determine whether the hospice event occurred at Inpatient, the ED, or the ICU. The function for each hospice claim is assigned as `Hospice Treatment`.

3.4 HealthcareSystem Class

The purposes of the `HealthcareSystem` class are to load the claims data from the locations specified in the `UserSettings` file, create the patient- and hospital-level heat map visualizations from each utilization file, and construct the XML and event list files for the HFGT Toolbox. The class structure is visualized in Figure 7 [21].

Once loaded from the file paths indicated by the user in the `UserSettings` file, the data is stored in the instance variable `claimsDataDict`, where the key is the name of the data file (e.g., “Part A”) and the value is the `DataFrame` containing the data.

Given a patient ID, the `visualizePatientUtilization` method imports a patient utilization

file created by the Patients object and creates four heat map visualizations that depict the patient's last 200 days of life. The first visualization is an MD NPI utilization heat map that displays each of the patient's providers and which type of care that they delivered. The second visualization is an organization NPI utilization heat map that depicts the patient's care in terms of all hospitals at which they received their care. Lastly, two service place-level heat maps are created that allow for the visualization of the places or hospital departments at which the patient was treated. The service place level 1 heat map abstracts the functions to the four categories of Transportation, Measurement, Decision, and Treatment, while the service place level 2 heat map displays all nine functions described above.



Figure 7: HealthcareSystem class diagram.

To create each visualization, the patient utilization is translated into a NumPy utilization matrix with 200 columns, each of which represents one day in the patient’s last 200 days of life, and as many rows as the number of possible form-function pairs in the utilization. A cell with a value of 1 signifies that the patient received care by a resource described by the row number on the day prior to death described by the column number. These utilization matrices are exported as .npy files that can be loaded for re-analysis or further analysis. Each visualization method leverages the Seaborn and PyPlot packages to create heat maps from the utilization matrices.

`visualizePatientUtilization` also invokes the class method `visualizePatientPlacesBeforeDeath`, which produces an abstracted heat map visualization of a patient’s clinical locations and health states during their last 200 days before death from the patient’s utilization file. A synthetic patient-level health state heat map is provided in the System Output section.

The `visualizeHospitalUtilization` method follows a similar procedure to that of the `visualizePatientUtilization` method in that the hospital utilization is loaded and MD NPI-level and service place-level visualizations are produced. One hospital-level utilization matrix is produced at each level by concatenating each of the patient-level utilization matrices so that the hospital utilization can be easily visualized at a later time if needed.

The `HealthcareSystem` class also produces two XML files for use in the HFGT Toolbox. The class method `createHFGTXML` constructs the XML files with the formats specified by the Toolbox documentation, which contains more detailed information on the structure of each file [20]. The LXML and XML DOM packages are used to create each attribute of each XML file and export the files. An example of each output file is provided and described in more detail in the following System Output section.

4 System Output

To demonstrate the patient- and hospital-level system output, we include synthetic examples of three hospitals with varying methods of EOL care. At Hospital 123456, patients 100, 101, and 102 generally receive very good EOL care, including early advance care planning, palliative care decision-making, and hospice care at home for their final weeks to months of life. In contrast,

patients 103, 104, and 105 at Hospital 830624 receive poor EOL care. They are rarely referred to hospice at home and instead receive aggressive chemotherapy and life-sustaining therapy until their death date, often dying in the inpatient ward or the ICU. Hospital 583350 demonstrates heterogeneous EOL care for patients 106, 107, and 108, in which some patients enter hospice early and others receive aggressive treatment until their death date. The individual trajectories for the patients will also be discussed. A preview of the output file structure for a hospital is shown in Figure 8.



Figure 8: A preview of the output file structure for a sample hospital.

4.1 Patient-Level Visualizations

From each patient-level MD NPI utilization heat map, we can clearly visualize how the patient’s EOL care differs across all of their providers as well as the hospital at which each of these providers is employed. Figure 9 shows that Patient 102 met with providers 26 and 28 at Hospital 123456 to discuss palliative care options and advance care planning before entering a long-term hospice stay at home, indicating better EOL outcomes of patients who are treated by these providers. Note that the hospice stay at home is not visualized in the MD-level heat map because the hospice stay is not associated with a provider, but this stay is depicted in the organization NPI and service place utilization heat maps. In contrast, we see in Figure 10

that Patient 105 received more aggressive EOL care – chemotherapy and life-sustaining therapy during the last 200 days of life – under provider 337 at Hospital 583350.

The patient-level organization NPI utilization heat map demonstrates the range of hospitals at which the patient received care. The x-axis of each organization NPI heat map displays all possibilities of form-function pairs at each hospital, even if they do not exist within the data, so that the heat maps can be compared between patients. This heat map is useful for providers and healthcare administrators to visualize the variety of institutions at which their patients receive care. Figure 11 shows that Patient 106 received treatment at several hospitals during their last 200 days of life.

The patient-level service place utilization heat maps abstract the patient’s care for the visualization of the departments within each hospital where the patient received their treatment. The service place level 1 visualizations aggregate the functions into the four categories of Transport, Measure, Decision, and Treatment, and the service place level 2 visualizations depict all nine possible functions. Figures 12 and 13 are service place level 1 and 2 visualizations for Patient 101, respectively. From both figures, we see that the patient had a short inpatient stay and two ED visits before being referred to hospice treatment at home for their last twenty days of life.

The patient-level health state heat map provides an abstracted, dynamic visualization of the patient’s health states and their clinical locations during their last 200 days of life. These figures can also be compared and contrasted to discern variations in EOL care among patients of different races and geographic locations. Juxtaposing the health state heat maps of Patients 100 and 108 in Figures 14 and 15, respectively, reveals the stark differences in the quality of these patients’ EOL care. Patient 100 spent the majority of their last 200 days of life at home, while Patient 108 had several extended inpatient stays and passed away in the ICU.

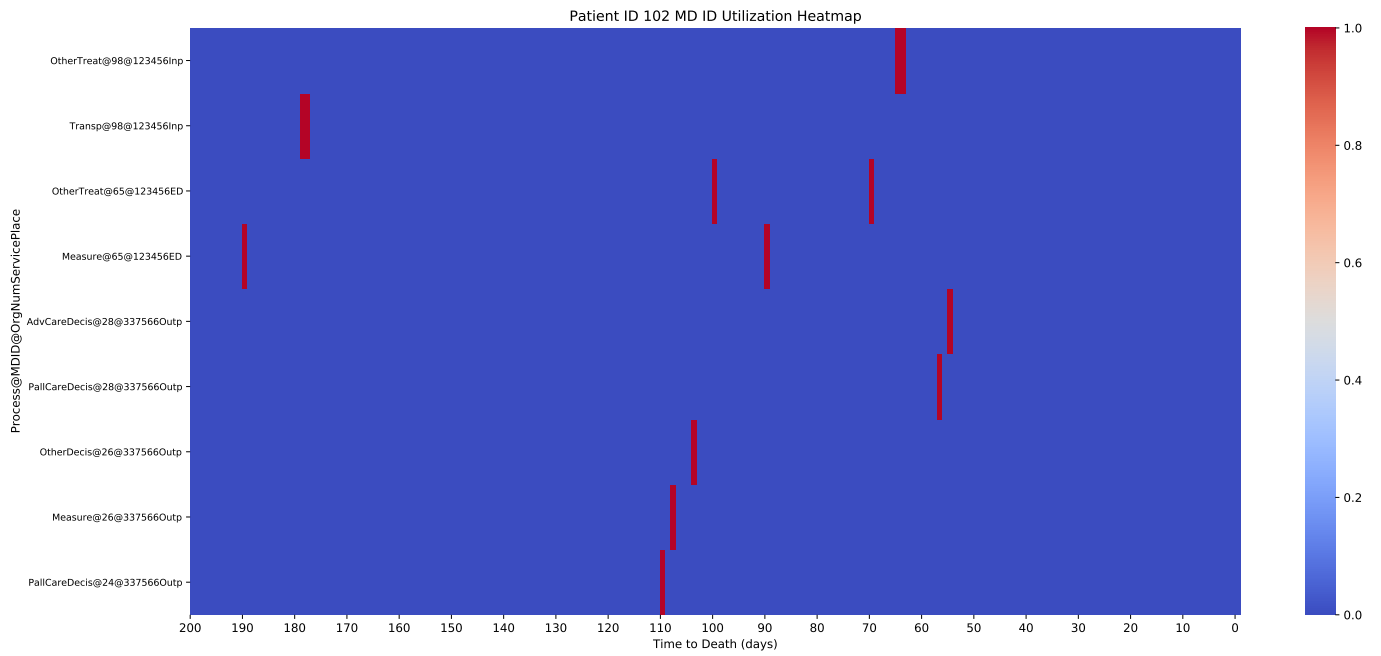


Figure 9: Patient 102 MD NPI utilization heat map.

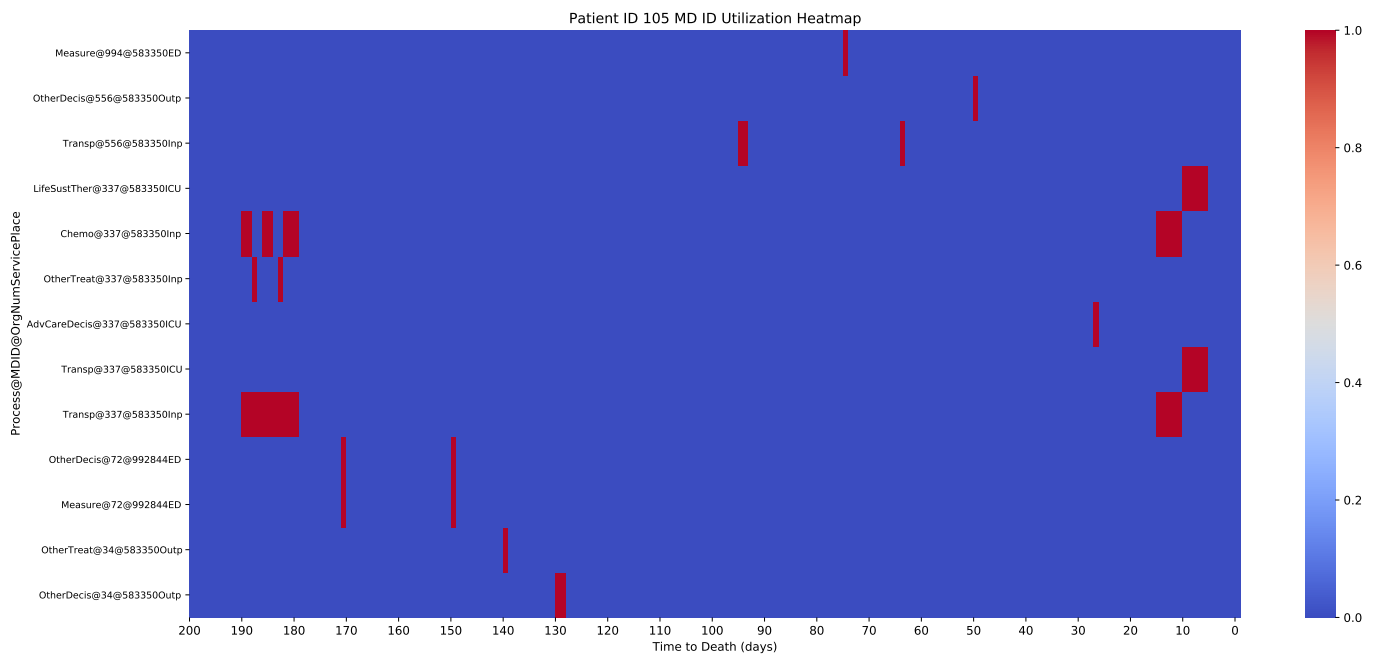


Figure 10: Patient 105 MD NPI utilization heat map.

Patient ID 106 Org Num Utilization Heatmap

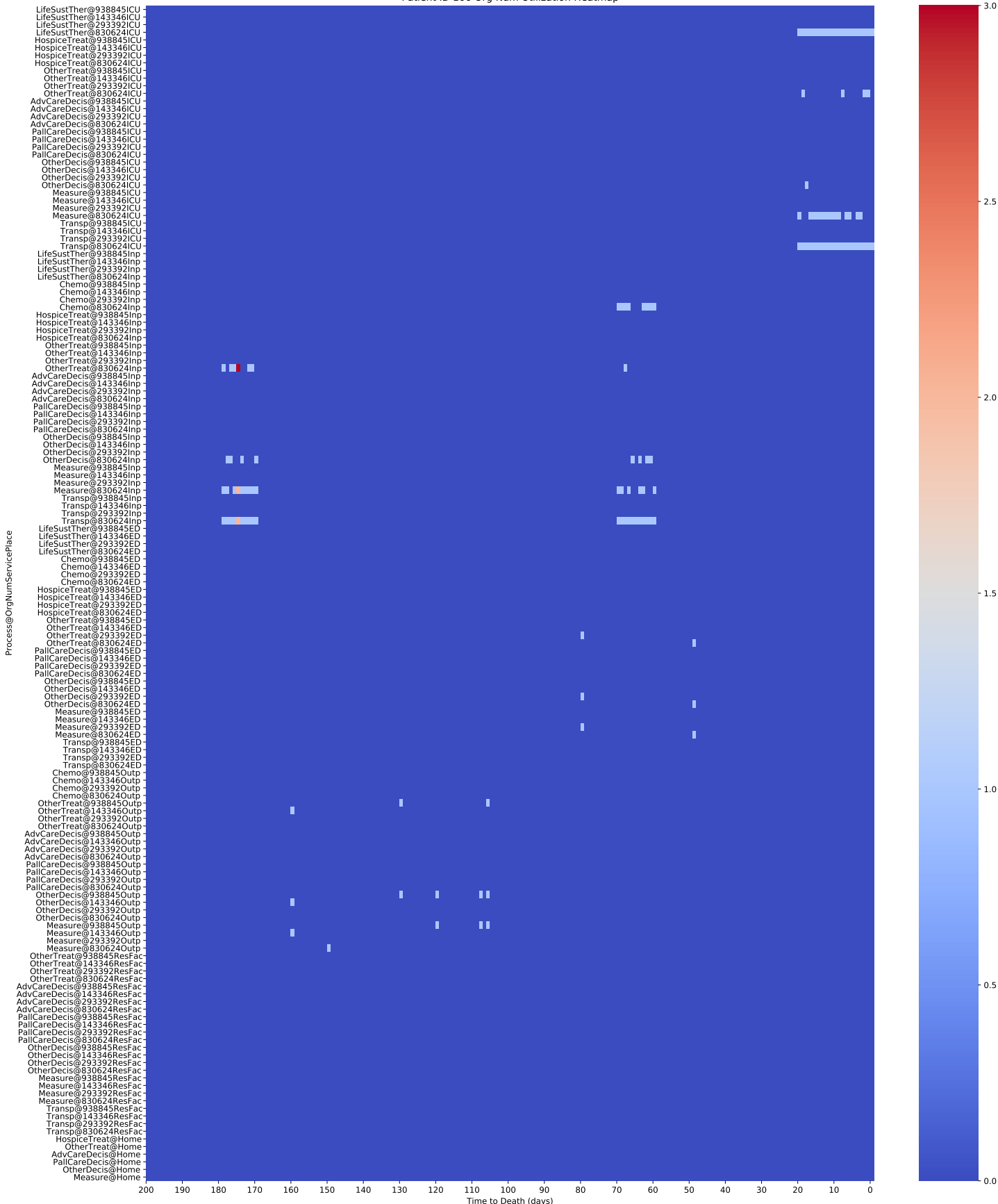


Figure 11: Patient 106 organization NPI utilization heat map.

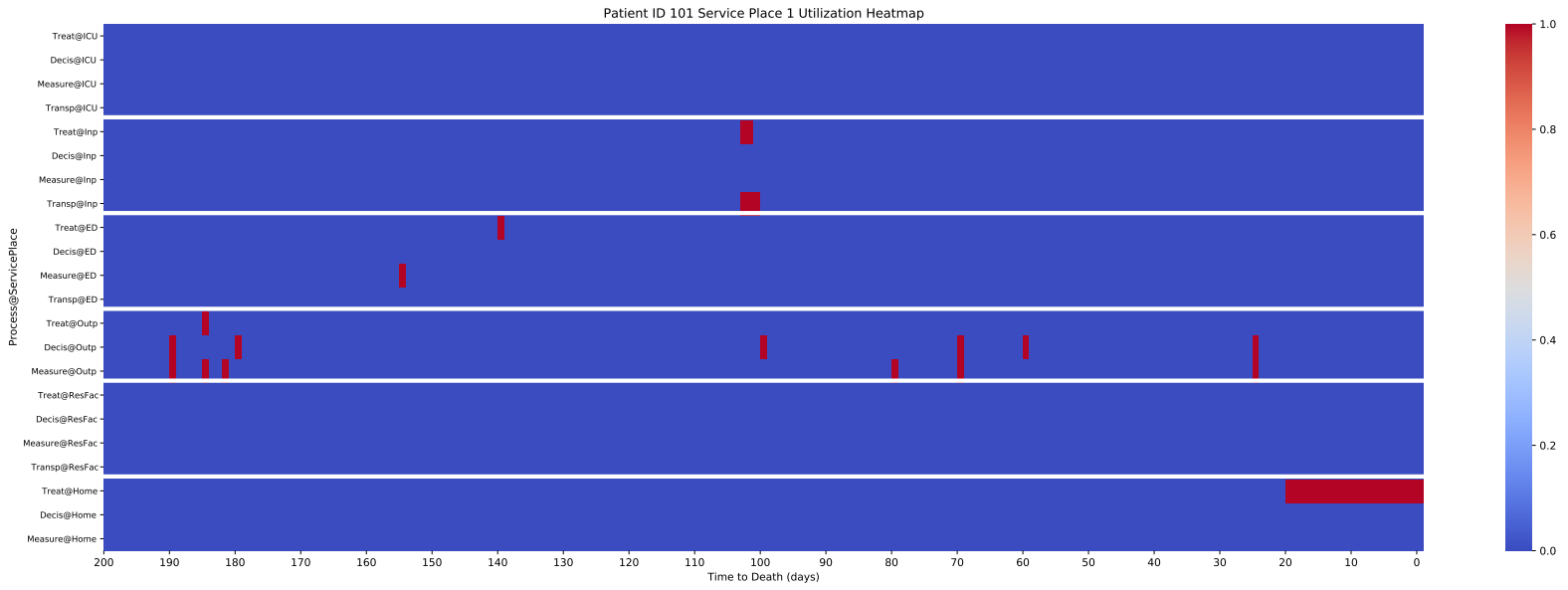


Figure 12: Patient 101 service place level 1 heat map.

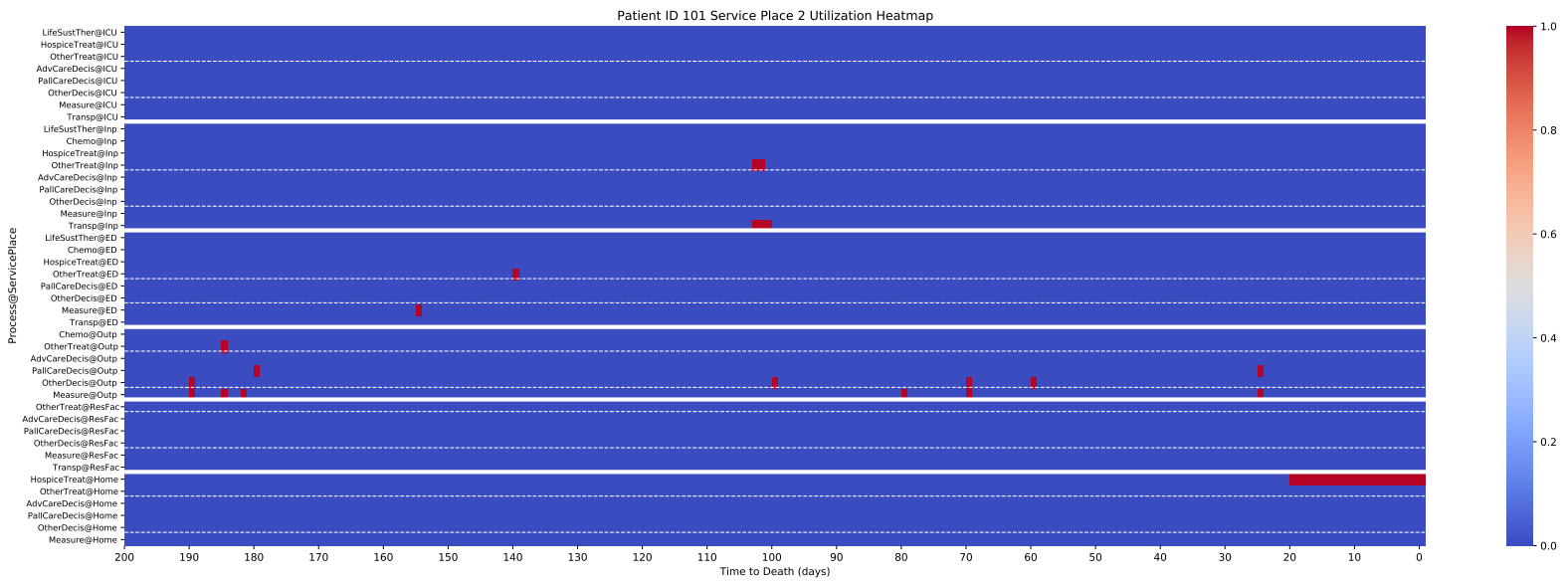


Figure 13: Patient 101 service place level 2 heat map.

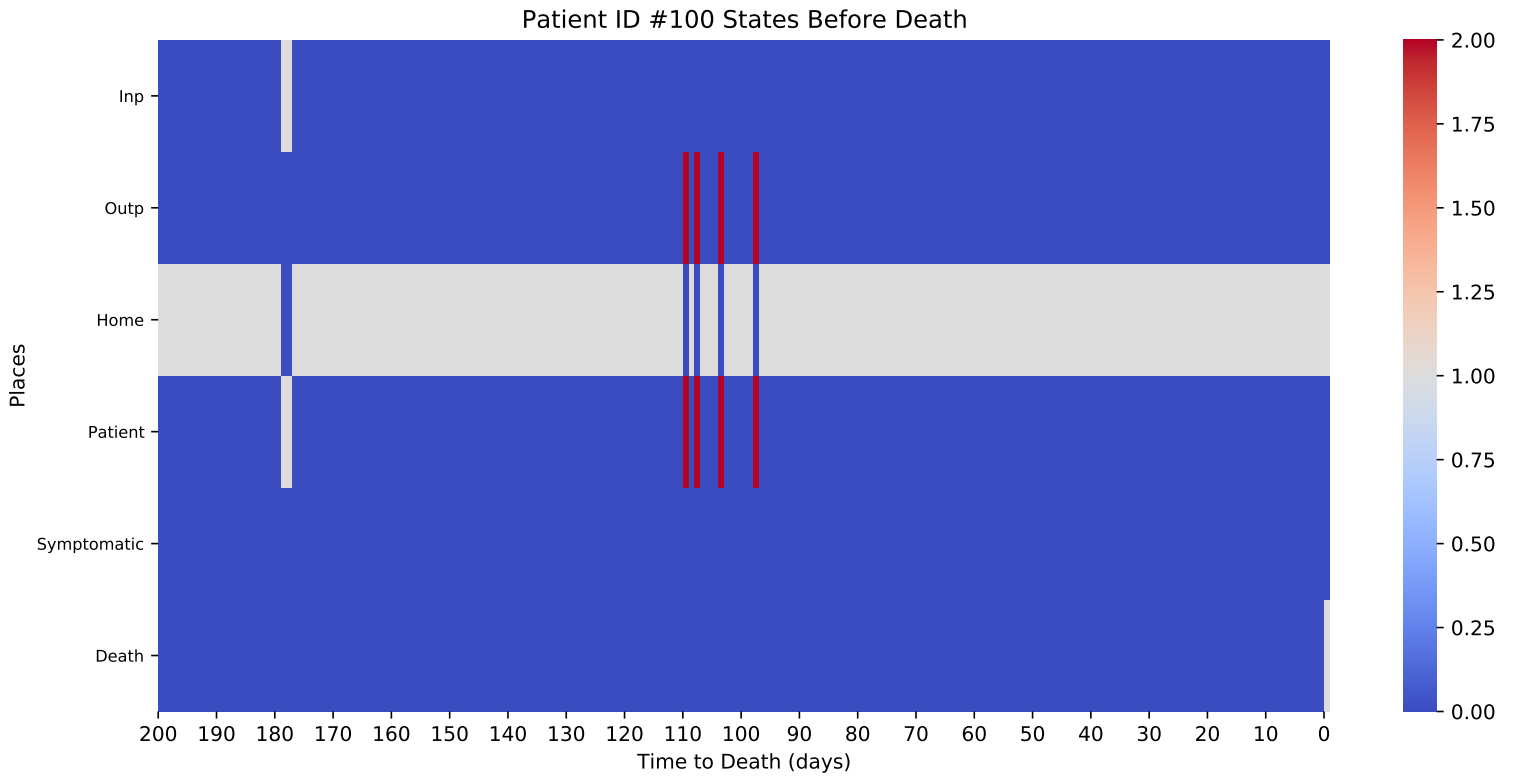


Figure 14: Patient 100 health state heat map.

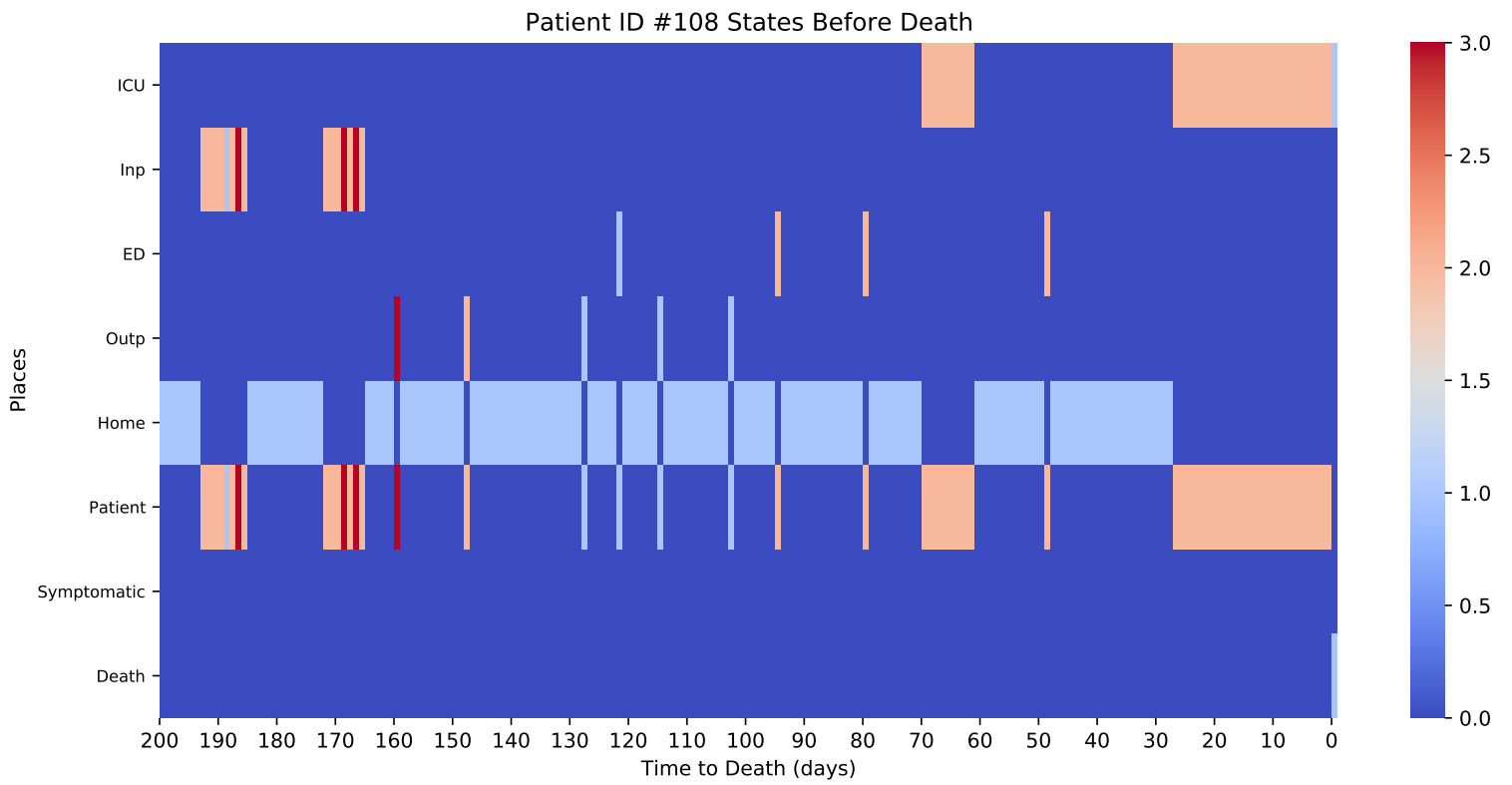


Figure 15: Patient 108 health state heat map.

4.2 Hospital-Level Visualizations

The hospital-level MD NPI utilization heat maps demonstrate clear patterns of care among clinicians and allow for the comparison and contrast of their treatment patterns between hospitals. Each cell of each hospital heat map is normalized by the number of patients assigned to the hospital, as seen in the color bar on the right side of the image.

For example, as demonstrated in Figure 16, providers 223, 878, and 935 at Hospital 830624 tend to treat patients using more aggressive methods such as chemotherapy and life-sustaining therapy during their last few days of life. In contrast, we can observe in Figure 17 that providers 24, 26, and 28 at Hospital 123456 meet with patients for palliative care decisions and advance care planning and are likely to recommend long-term hospice care at home. The heat map for Hospital 583350 demonstrates a mix of EOL care for its patients in Figure 18. Provider 303 provides better EOL care to their patients by making decisions related to palliative care and advance care planning, which leads to home-based hospice referrals in the last few days of life. In contrast, provider 337 tends to administer more aggressive modes of treatment, such as chemotherapy and life-sustaining therapy. Such provider-specific figures can aid hospital administrators in distinguishing which providers may require more or specific training to improve their EOL care patterns for their terminal patients.

Comparing and contrasting the service place-level heat maps further emphasizes the differences in EOL care between the hospitals. In line with the pattern of poor EOL care at Hospital 830624 demonstrated in other parts of the analysis, the level 1 service place visualization shown in Figure 20 illustrates that many patients receive aggressive treatment during their last few days of life and pass away in the ICU. The patients at Hospital 123456 receive more favorable EOL care in terms of early palliative care decision-making, advance care planning, and hospice referral, as visualized in Figure 19.

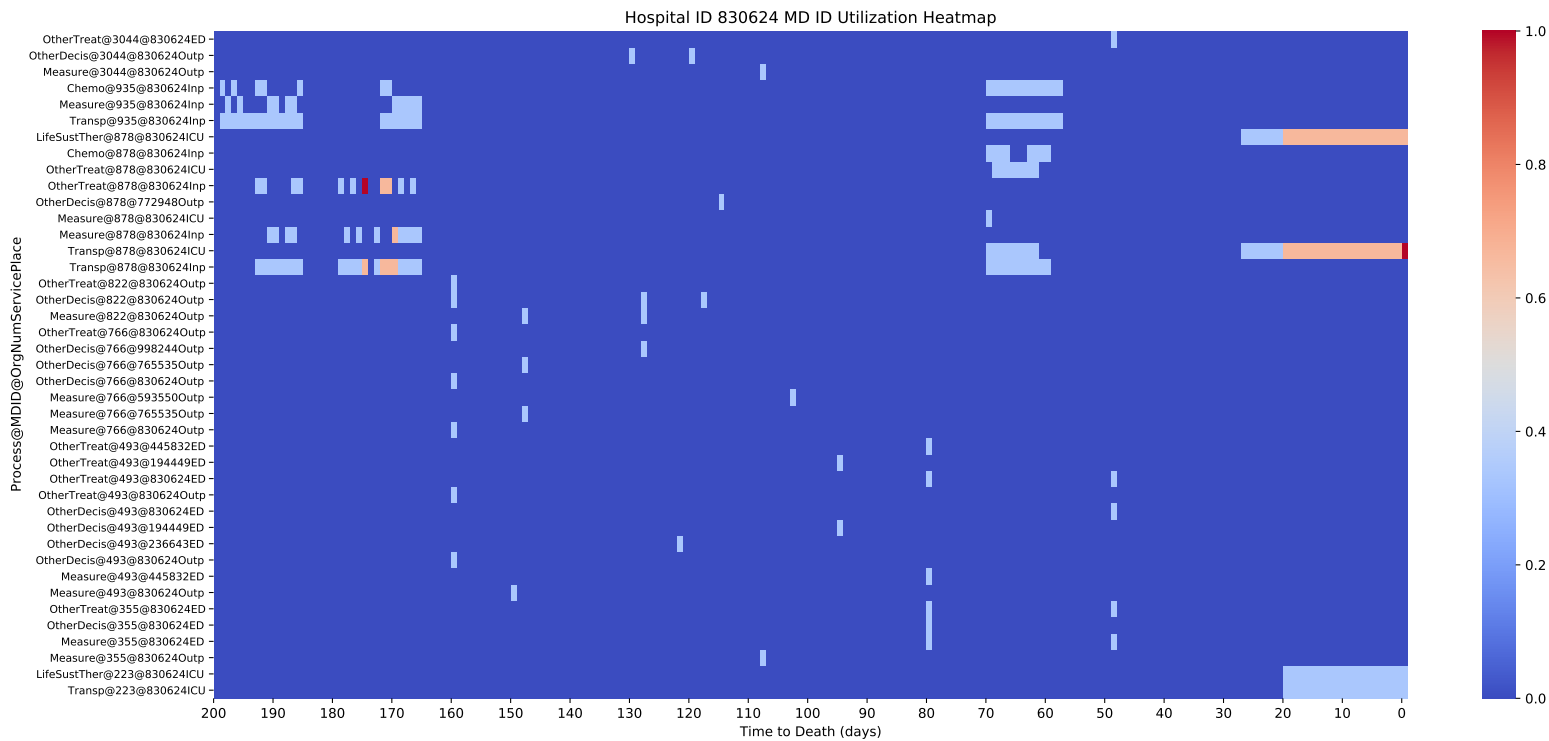


Figure 16: Hospital 830624 MD NPI utilization heat map.

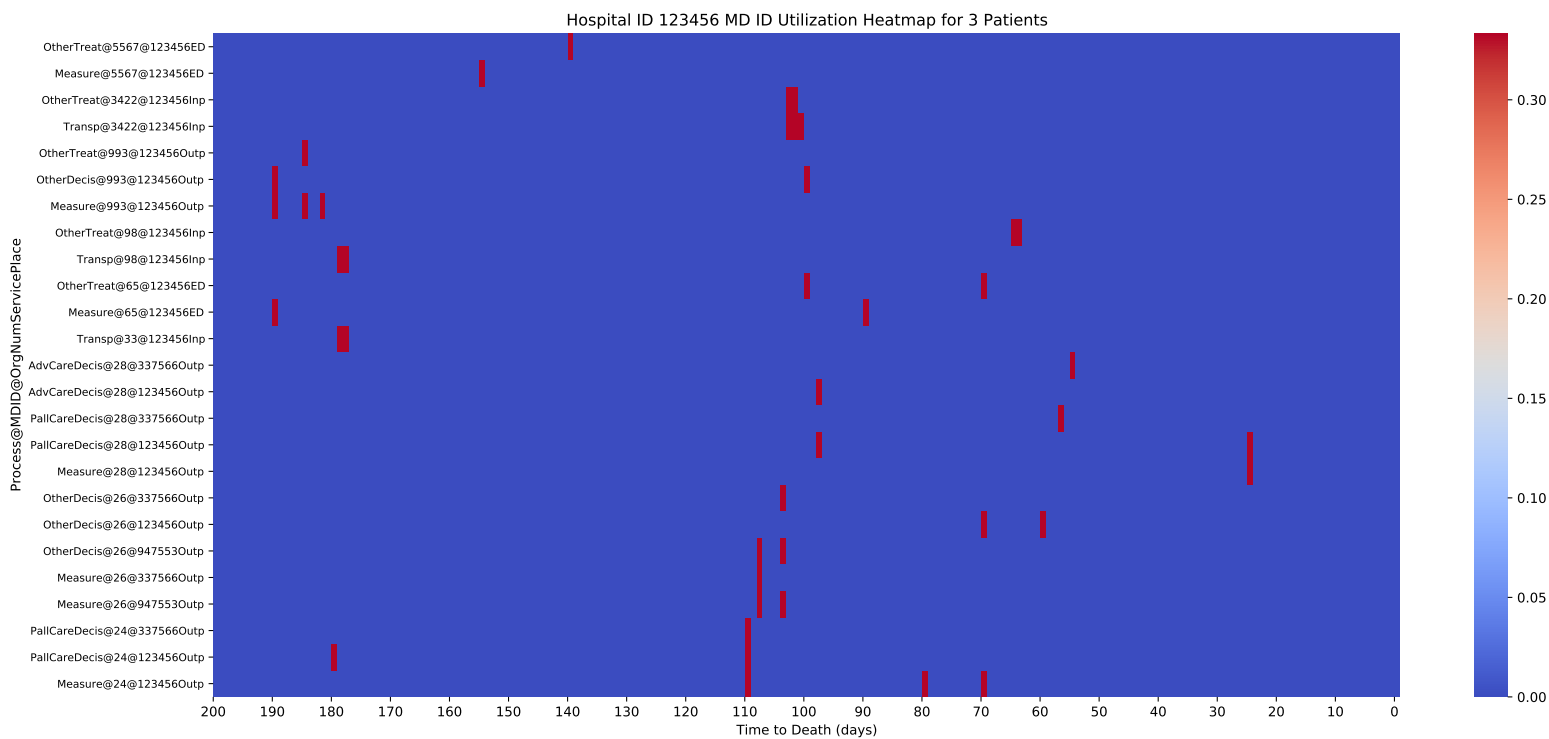


Figure 17: Hospital 123456 MD NPI utilization heat map.

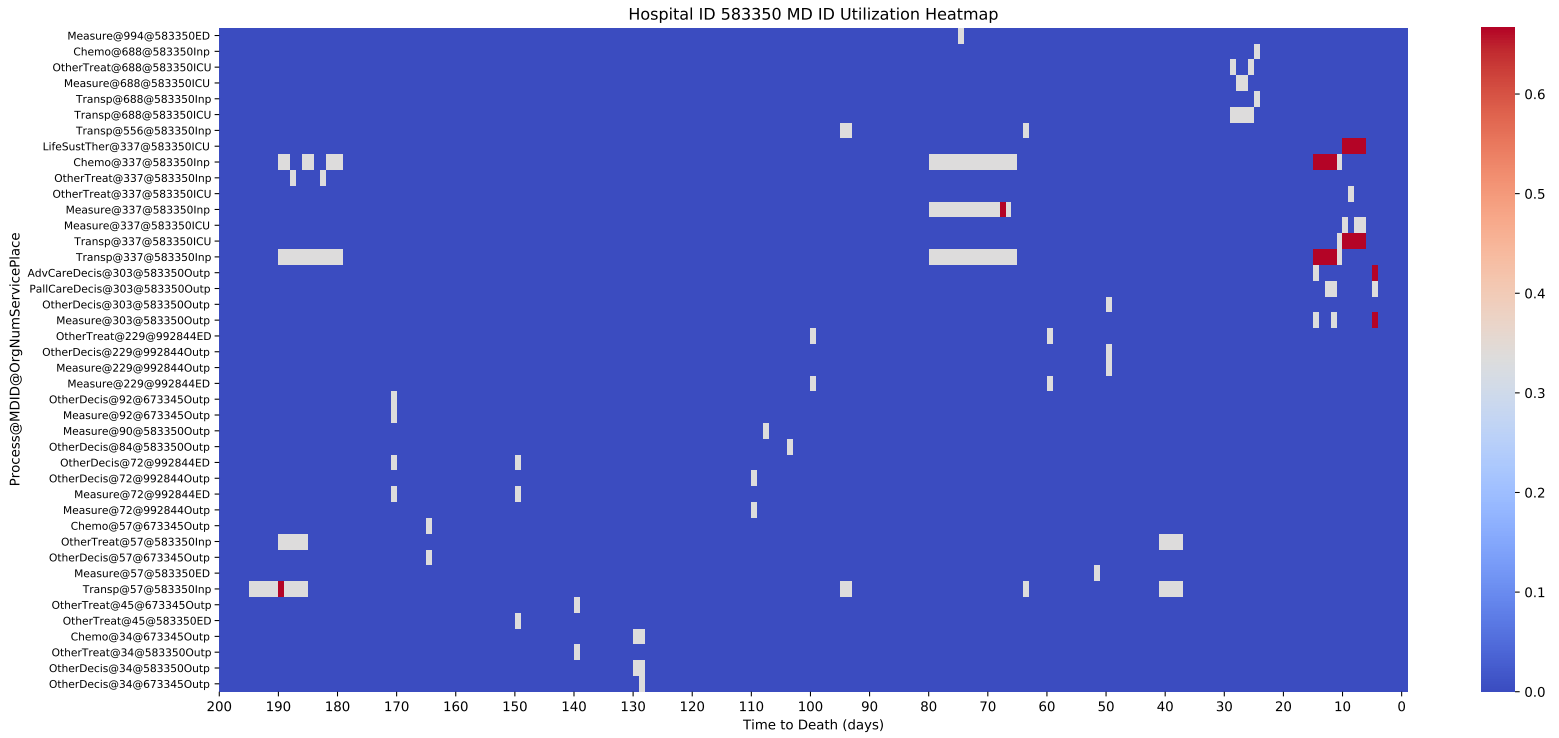


Figure 18: Hospital 583350 MD NPI utilization heat map.

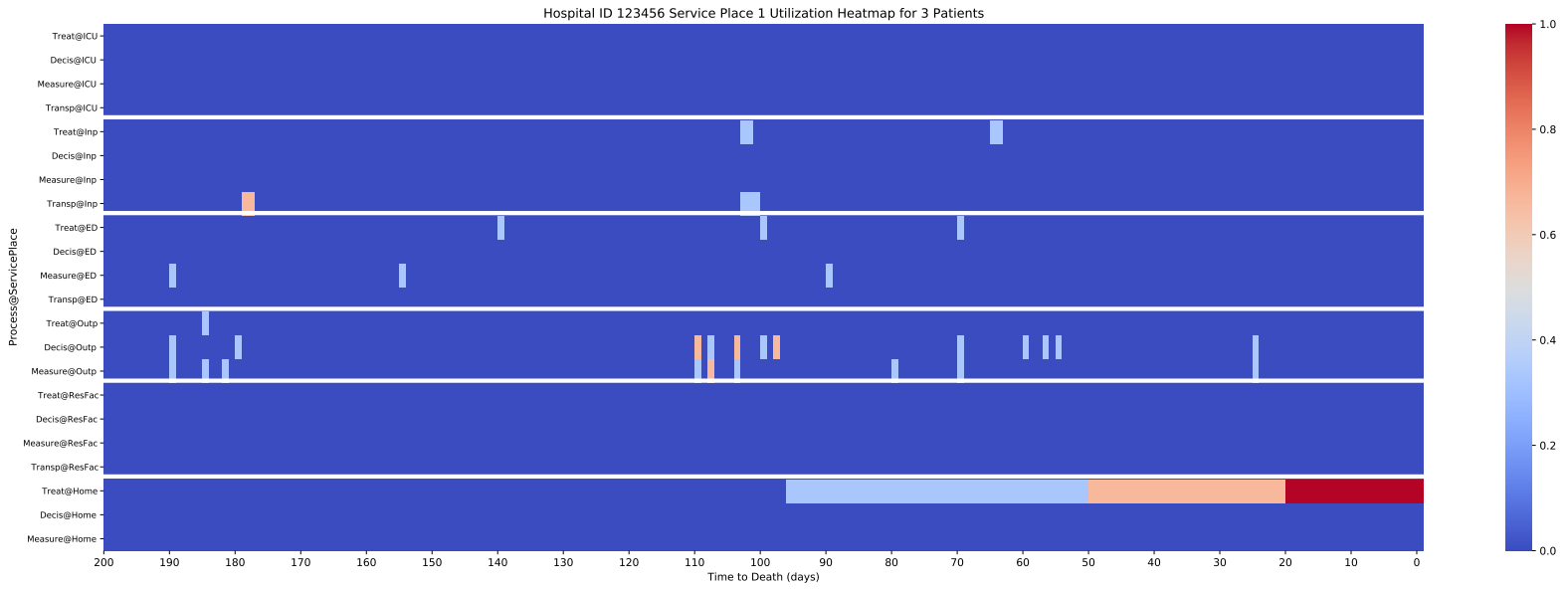


Figure 19: Hospital 123456 Service Place 1 heat map.

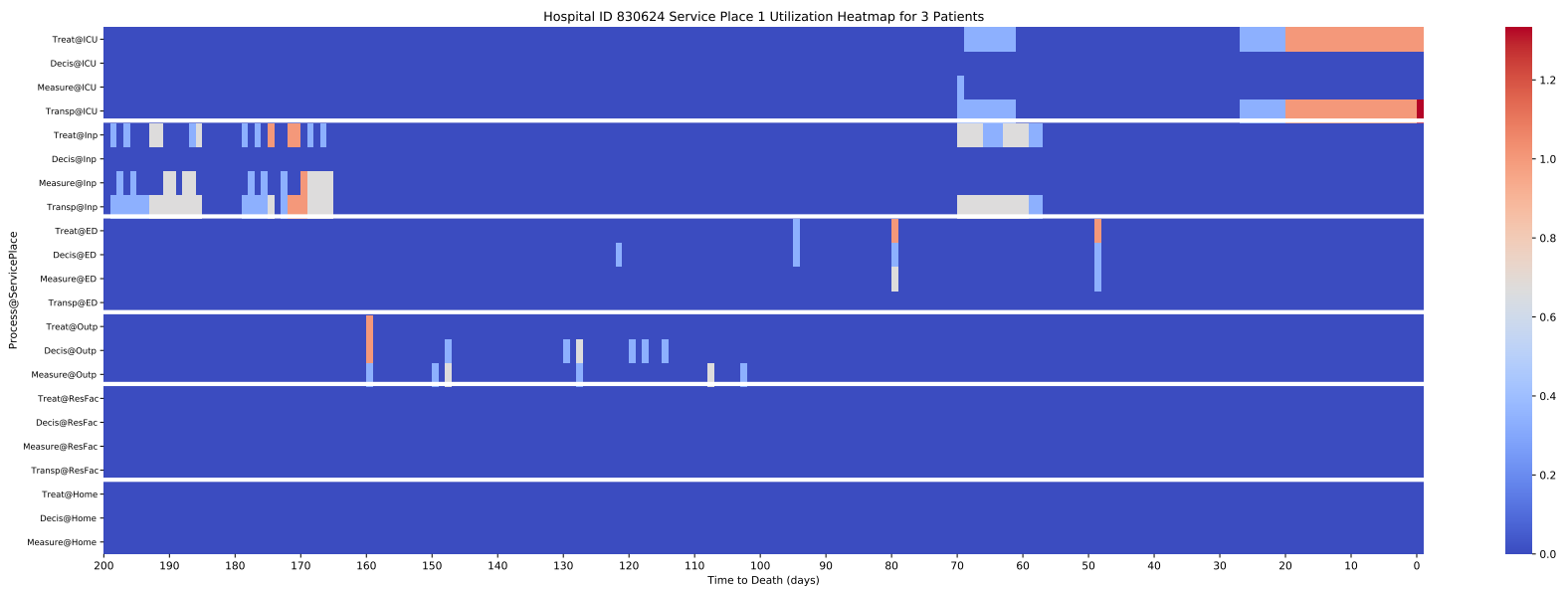


Figure 20: Hospital 830624 Service Place 1 heat map.

4.3 Hetero-Functional Graph Theory Toolbox Files

The Patients and HealthcareSystem classes produce several files that can be processed by the HFGT Toolbox system, which provides the discrete event simulation of the claims data. A more detailed description of the structure of the input files required for the Toolbox can be found within the documentation [20].

As necessitated by the Toolbox, two input XML files are constructed per hospital. The first XML file is required to construct the HFGT mathematical models, and the second XML file allows for the visualization of the system in a Petri net using the GUI-based Petri net simulator of the Toolbox. The first XML file contains each resource modeled as a Machine attribute with GPS coordinates, non-Transportation functions modeled as MethodxFom attributes, and Transportation functions modeled as MethodxPort attributes. The second XML file contains the same attributes as the first file but requires the addition of several other attributes for Petri net visualization, such as the number of initialization tokens, the GPS offset, and the process duration time. Figure 21 provides a preview of the first type of XML file.

The Patients object constructs patient-level and hospital-level scheduled event lists from each of the claim events present in the utilization files. These event lists are used by the Toolbox to visualize the flow of patient care through the healthcare delivery system via a Petri net. Each line of the event list contains the patient ID, the day number on which the claim event occurred, the numerical index representing the resource utilized on that day, and the numerical index representing the process that occurred on that day. Figure 22 shows a preview of a scheduled event list produced by the system.

```
<?xml version="1.0" ?>
<LFES dataState="raw" name="Hospital ID 234567" type="Healthcare">
  <Operand name="individual"/>
  <Operand name="patient"/>
  <Machine gpsX="0" gpsY="0" name="Home">
    <MethodxFom name="Patient ages" operand="individual" output="individual" status="true"/>
    <MethodxPort dest="Home" name="Transport" operand="individual" origin="Home" output="individual" ref="Vehicle" status="true"/>
  </Machine>
  <Machine gpsX="0" gpsY="8" name="ResFac">
    <MethodxFom name="Transp" operand="patient" output="patient" status="true"/>
    <MethodxFom name="OtherTreat" operand="patient" output="patient" status="true"/>
    <MethodxPort dest="ResFac" name="Transport" operand="patient" origin="ResFac" output="patient" ref="Ambulance" status="true"/>
  </Machine>
  <Machine gpsX="12" gpsY="8" name="Outp">
    <MethodxFom name="AdvCareDecis" operand="patient" output="patient" status="true"/>
    <MethodxPort dest="Outp" name="Transport" operand="patient" origin="Outp" output="patient" ref="Ambulance" status="true"/>
  </Machine>
</LFES>
```

Figure 21: A preview of an XML file generated for a sample hospital.

```

idxToken,tStart,idxResource,idxProcess
idx,min,idx,idx
XXXXXXXX101,1.0,1,4
XXXXXXXX101,171.0,4,8
XXXXXXXX101,171.0,2,2
XXXXXXXX101,171.0,2,4
XXXXXXXX101,171.0,2,3
XXXXXXXX101,171.0,4,8
XXXXXXXX101,171.25,2,2
XXXXXXXX101,171.5,2,4
XXXXXXXX101,171.75,2,3
XXXXXXXX101,189.0,5,21
XXXXXXXX101,189.0,3,2
XXXXXXXX101,189.0,3,6
XXXXXXXX101,189.0,5,21
XXXXXXXX101,189.0,3,3
XXXXXXXX101,189.25,3,2
XXXXXXXX101,189.5,3,6
XXXXXXXX101,189.75,3,3
XXXXXXXX101,198.0,5,23
XXXXXXXX101,198.0,2,5
XXXXXXXX101,198.0,2,2
XXXXXXXX101,198.0,5,23
XXXXXXXX101,198.333,2,2
XXXXXXXX101,198.666,2,5
XXXXXXXX101,200,1,1

```

Figure 22: A preview of an event list generated for a sample hospital.

5 Discussion and Future Work

The DEMAC system provides dynamic visualization and analysis of EOL care for individual patients as well as larger hospital systems. There are several possible routes for extending the DEMAC system’s analytical capabilities. The system currently supports the grouping of patients by their most frequently visited hospital. The ability to cluster patients by race, gender, or rurality would allow for the visualization and comparison of disparities in healthcare delivery across these different groups. We also plan to expand the system to include analysis of Home Health Agency, SNF, and Outpatient Medicare claims files.

Another important feature that would supplement the current DEMAC system analysis is the incorporation of treatment expense data into the modeling process. The rising costs of EOL care in the U.S. signal an urgent need to elucidate the patterns of spending within the healthcare delivery system [11]. A cost-driven analysis via the DEMAC system would be useful for healthcare administrators to pinpoint the sources of the most expensive EOL treatments, which are likely to be aggressive therapies such as chemotherapy and life-sustaining treatment.

Lastly, the DEMAC system can be extended to analyze aspects of the American healthcare delivery system outside of EOL care. For instance, the DEMAC system could be adapted to visualize the dynamic healthcare resource utilization of low-income patients using the Medicaid claims files. Furthermore, given the heterogeneous outcomes of patients with advanced cases of COVID-19, the DEMAC system could potentially be used to analyze the differences in EOL care for COVID-19 patients across the country.

6 Code and Data Availability

We intend to publish the code and sample data for this thesis on GitHub within the next few months.

7 Acknowledgments

I would like to thank my advisor Dr. Inas Khayal for her guidance and feedback throughout the design, development, and publication stages of this thesis project. I am also grateful to my family and friends for their continued support.

References

- [1] Firas Abdollah et al. “Racial disparities in end-of-life care among patients with prostate cancer: a population-based study”. In: *Journal of the National Comprehensive Cancer Network* 13.9 (2015), pp. 1131–1138.
- [2] Melissa DA Carlson et al. “Geographic access to hospice in the United States”. In: *Journal of palliative medicine* 13.11 (2010), pp. 1331–1338.
- [3] Amber E Barnato et al. “Advance care planning norms may contribute to hospital variation in end-of-life ICU use: a simulation study”. In: *Medical decision making* 34.4 (2014), pp. 473–484.

- [4] Amber E Barnato et al. “Norms of decision making in the ICU: a case study of two academic medical centers at the extremes of end-of-life treatment intensity”. In: *Intensive care medicine* 38.11 (2012), pp. 1886–1896.
- [5] Aria A Razmaria. “End-of-life care”. In: *Jama* 316.1 (2016), pp. 115–115.
- [6] Arianne Brinkman-Stoppelenburg, Judith AC Rietjens, and Agnes Van der Heide. “The effects of advance care planning on end-of-life care: a systematic review”. In: *Palliative medicine* 28.8 (2014), pp. 1000–1025.
- [7] Alina Baciu et al. “The state of health disparities in the United States”. In: *Communities in action: pathways to health equity*. National Academies Press (US), 2017.
- [8] Tessa Jones et al. “Advance Care Planning, Palliative Care, and End-of-life Care Interventions for Racial and Ethnic Underrepresented Groups: A Systematic Review”. In: *Journal of Pain and Symptom Management* (2021).
- [9] S McGowan et al. “A Systematic Review of Racial Disparities in the Intensive Care Unit”. In: *TP19. TP019 Quality, Process, and Outcomes in Acute and Critical Care* (2021), A1691–A1691.
- [10] Jonas Nilsson et al. “End-of-life care: where do cancer patients want to die? A systematic review”. In: *Asia-Pacific Journal of Clinical Oncology* 13.6 (2017), pp. 356–364.
- [11] Paul E Marik. “The cost of inappropriate care at the end of life: implications for an aging population”. In: *American Journal of Hospice and Palliative Medicine*® 32.7 (2015), pp. 703–708.
- [12] Haya R Rubin, Peter Pronovost, and Gregory B Diette. “The advantages and disadvantages of process-based measures of health care quality”. In: *International Journal for Quality in Health Care* 13.6 (2001), pp. 469–474.
- [13] Diane Whitmore Schanzenbach, Ryan Nunn, and Lauren Bauer. “The changing landscape of American life expectancy”. In: *Washington, DC: The Hamilton Project, June* (2016).
- [14] Wullianallur Raghupathi and Viju Raghupathi. “An empirical study of chronic diseases in the United States: a visual analytics approach to public health”. In: *International journal of environmental research and public health* 15.3 (2018), p. 431.

- [15] Wester C. H. Schoonenberg, Inas S. Khayal, and Amro M. Farid. “Hetero-functional Graph Theory”. en. In: *A Hetero-functional Graph Theory for Modeling Interdependent Smart City Infrastructure*. Cham: Springer International Publishing, 2019, pp. 37–93. ISBN: 9783319993003. DOI: 10.1007/978-3-319-99301-0_4. URL: http://link.springer.com/10.1007/978-3-319-99301-0_4 (visited on 05/26/2021).
- [16] Wester CH Schoonenberg, Inas S Khayal, and Amro M Farid. *A Hetero-functional Graph Theory for Modeling Interdependent Smart City Infrastructure*. Springer, 2019.
- [17] Dakota J Thompson, Wester CH Schoonenberg, and Amro M Farid. “Hetero-functional Graph Resilience of the Future American Electric Grid”. In: *arXiv preprint arXiv:2006.10678* (2020).
- [18] Inas S Khayal and Amro M Farid. “A Dynamic System Model for Personalized Healthcare Delivery and Managed Individual Health Outcomes”. In: *arXiv preprint arXiv:1910.09104* (2019).
- [19] Inas S Khayal. “Dynamic Modeling of Complex Healthcare Systems Using Big Data to Describe and Visualize Healthcare Utilization”. In: *2020 IEEE International Systems Conference (SysCon)*. IEEE, pp. 1–8.
- [20] Prabhat Hegde et al. “The hetero-functional graph theory toolbox”. In: *arXiv preprint arXiv:2005.10006* (2020).
- [21] Visual Paradigm International. *Visual Paradigm*. Version 16.3. 2020. URL: <https://https://www.visual-paradigm.com/>.
- [22] The Centers for Medicare and Medicaid Services (CMS). *Medicare Data for the Geographic Variation Public Use File: A Methodological Overview*. URL: https://www.cms.gov/Research-Statistics-Data-and-Systems/Statistics-Trends-and-Reports/Medicare-Geographic-Variation/Downloads/Geo_Var_PUF_Methods_Paper.pdf.
- [23] Research Data Assistance Center (ResDAC). *MEDPAR*. URL: <https://resdac.org/cms-data/files/medpar>.

- [24] Research Data Assistance Center (ResDAC). *MEDPAR Short Stay/Long Stay/SNF Indicator Code*. URL: <https://resdac.org/cms-data/variables/medpar-short-staylong-staysnf-indicator-code>.
- [25] Research Data Assistance Center (ResDAC). *MEDPAR Provider Number*. URL: <https://resdac.org/cms-data/variables/medpar-provider-number>.
- [26] The Centers for Medicare and Medicaid Services (CMS). *2021 ICD-10-PCS*. 2021. URL: <https://www.cms.gov/medicare/icd-10/2021-icd-10-pcs/>.
- [27] Research Data Assistance Center (ResDAC). *Carrier (Fee-for-Service)*. URL: <https://resdac.org/cms-data/files/carrier-ffs>.
- [28] Research Data Assistance Center (ResDAC). *Place of Service Code Set*. URL: https://www.cms.gov/Medicare/Coding/place-of-service-codes/Place_of_Service_Code_Set.
- [29] Research Data Assistance Center (ResDAC). *Healthcare Common Procedure Coding System (HCPCS) Code (FFS)*. URL: <https://resdac.org/cms-data/variables/healthcare-common-procedure-coding-system-hcpcs-code-ffs>.
- [30] Research Data Assistance Center (ResDAC). *Line Berenson-Eggers Type of Service (BETOS) Code*. URL: <https://resdac.org/cms-data/variables/line-berenson-eggers-type-service-betos-code>.
- [31] Research Data Assistance Center (ResDAC). *Inpatient (Fee-for-Service)*. URL: <https://resdac.org/cms-data/files/ip-ffs>.
- [32] Research Data Assistance Center (ResDAC). *Revenue Center Code (FFS)*. URL: <https://resdac.org/cms-data/variables/revenue-center-code-ffs>.
- [33] Research Data Assistance Center (ResDAC). *Claim Inpatient Admission Type Code (FFS)*. URL: <https://resdac.org/cms-data/variables/claim-inpatient-admission-type-code-ffs>.
- [34] Research Data Assistance Center (ResDAC). *Hospice (Fee-for-Service)*. URL: <https://resdac.org/cms-data/files/hospice-ffs>.

- [35] Research Data Assistance Center (ResDAC). *Claim Facility Type Code (FFS)*. URL: <https://resdac.org/cms-data/variables/claim-facility-type-code-ffs>.
- [36] Research Data Assistance Center (ResDAC). *Claim Service Classification Type Code (FFS)*. URL: <https://resdac.org/cms-data/variables/claim-service-classification-type-code-ffs>.