

Process Mining to Support Discrete Event Simulation (DES) Modelling

Fatin Izzati Ramli

Thesis submitted in partial fulfillment of the requirements for the award of the Doctor of Philosophy

Division of Population Health

School of Medicine and Population Health

UNIVERSITY OF SHEFFIELD

APRIL 2025

ACKNOWLEDGEMENTS

Alhamdulillah, all praise to Allah, the Most Gracious and Most Merciful. May Peace and blessings be upon our Prophet Muhammad, his family and Companions!

I am deeply grateful for the support and encouragement that I have received throughout my PhD journey. This accomplishment would not have been possible without the guidance, prayers, and love of those who stood by me.

First and foremost, I extend my heartfelt gratitude to my parents and sister for their unconditional support. Your trust in me has been my constant source of strength and motivation.

I would like to express my sincere appreciation to my supervisor and collaborators for their patience and invaluable guidance in my research endeavours. Your mentorship has been instrumental in shaping my academic growth.

I also want to pay tribute to my late uncle, with whom we initially shared the aspiration to embark on this PhD journey. Though you are no longer with us, your encouragement has continued to inspire me throughout this process.

In memory of a dear friend who provided support and motivation for this study but is no longer with us due to the COVID-19 pandemic, your friendship and encouragement will forever be cherished. To all those who have been impacted by the COVID-19 pandemic, my thoughts and prayers are with you. It has been a challenging time for many, and I look forward to the day when we can all spend time with our loved ones once again.

This achievement is the result of the collective support and blessings of Allah, my family, friends, and mentors. I am deeply thankful for the opportunities and experiences that have shaped my academic journey.

May Allah's blessings and guidance continue to illuminate our paths as we move forward.

ABSTRACT

Effective healthcare processes are vital for delivering high-quality patient care within healthcare organisations. This doctoral research explores the integration of Process Mining (PM) into Discrete Event Simulation (DES) as a transformative approach to enhance simulation modelling.

To begin, a comprehensive review of relevant literature was conducted to thoroughly examine the historical integration of process mining in DES. This involved an in-depth analysis of methodologies and identification of limitations, establishing a robust foundation for the research.

Building on the insights gained from the literature review, an initial framework was developed. This framework aimed to facilitate the integration of process mining into DES modelling using a structured step-by-step approach.

To validate the proposed framework and gain practical insights within the healthcare domain, two case studies were conducted. These case studies not only shed light on opportunities and challenges specific to healthcare but also contributed to the iterative refinement of the framework based on the experiences gained. Furthermore, these case studies were used to compare different approaches for integrating process mining into DES (i.e. expert-driven and fully automated) versus traditional approaches for DES modelling.

Finally, in an effort to offer practical guidance, the initial step-by-step framework was refined, and a best practice checklist was developed. This checklist presents essential steps and considerations for effectively leveraging process mining to support DES modelling, providing guidance throughout the integration process.

TABLE OF CONTENT

ACKNOWLEDGEMENTS **i**

ABSTRACT **ii**

TABLE OF CONTENT **iii**

LIST OF TABLES **xi**

LIST OF FIGURES **xiii**

LIST OF ABBREVIATIONS **xv**

LIST OF PUBLICATIONS **xvi**

CHAPTER 1 INTRODUCTION **1**

1.1 Chapter Overview 1

1.2 Background 1

1.3 Introduction to Process Mining 2

1.4 Introduction to Discrete Event Simulation Modelling 3

1.5 Problem statement 4

1.6 Aim of research 5

1.6.1 Specific objectives 5

1.6.2 Research questions 5

1.6.3 Scope of the PhD study 6

1.7 Thesis layout 6

1.7.1 Chapter 1: Introduction 6

1.7.2 Chapter 2: Process Mining and DES Modelling Methodologies: Concepts and Literature Review. 7

1.7.3 Chapter 3: Initial Step-By-Step Framework 7

1.7.4 Chapter 4: Exploring Usability of Process Mining in DES Modelling Using Breast Cancer Dataset 7

1.7.5 Chapter 5: Conceptual Model And Parameters Resulting From Process Mining With The UK Kidney Dataset 8

1.7.6 Chapter 6: Comparison of Traditional Simulation Modelling Approach Vs (Fully Automated And Expert Driven) Process Mining Approaches 8

1.7.7 Chapter 7: Enhanced Final Step-By-Step Framework and Best Practice Checklist 9

1.7.8 Chapter 8: Reflections, Future Works and Conclusion 9

CHAPTER 2 PROCESS MINING AND DES MODELLING METHODOLOGIES: CONCEPTS AND LITERATURE REVIEW 10

2.1 Chapter Overview 10

2.2 Process mining Methodology 10

2.2.1 Overview of process mining 11

2.2.2 Inputs to process mining. 12

2.2.3 Steps in Process mining 13

2.2.4 Outputs from process mining 35

2.3 Discrete event simulation methodology 36

2.3.1 Overview of DES 37

2.3.2 Inputs to DES modelling 39

2.3.3 Steps involved in developing DES modelling. 40

2.3.4 Outputs of DES 42

2.4 Systematic literature review: Integrating process mining in DES modelling. 43

2.4.1 Aim of the literature review 43

2.4.2 Methods 44

2.4.3 Literature Searches 44

2.4.4 Study Selection 45

2.4.5 Data Synthesis 47

2.5 Results 47

2.5.1 Overview of Retrieved Articles 48

2.5.2 Steps in process mining 56

2.6 How process mining has been used in DES modelling 57

2.6.1 Conceptual modelling 58

2.6.2 Parameterisation 60

2.6.3 Validation 62

2.7 Discussion 67

2.8 Addressing gap in the current state of art in integrating process mining in DES modelling. 71

CHAPTER 3 INITIAL STEP-BY-STEP FRAMEWORK 73

3.1 Chapter Overview 73

3.2 A step-by-step framework from literature review 73

3.2.1 Existing Framework from Rozinat and Tamburis 73

3.2.2 Comparison of Rozinat's and Tamburis' frameworks 82

3.2.3 Discussion from Tamburis and Rozinat Framework analysis 85

3.2.4 Phase 1: Data Preparation 90

3.2.5 Phase 2: Process Mining 93

3.2.6 Phase 3: Transformation To DES modelling 99

3.2.7 Phase 4: Validation 102

3.2.8 Overall Discussion 104

CHAPTER 4 EXPLORING USABILITY OF PROCESS MINING IN DES MODELLING USING MEXICAN BREAST CANCER DATASET 105

4.1 Chapter Overview 105

4.2 Introduction to Breast Cancer 105
4.2.1 Delay in breast cancer pathways 107
4.3 Aims of case study 109
4.4 Dataset 109
4.4.1 Overview of the dataset 109
4.4.2 Variables in the dataset 110
4.5 Implementation of the Proposed Framework Using the Mexican Breast Cancer Dataset 111
4.5.1 Phase 1: Data preparation 111
4.5.2 Phase 2: Process mining 123
4.5.3 Phase 3: Converting process mining results into DES modelling. 164
4.5.4 Phase 4: Validation 170
4.6 Value of Process Mining in DES Model Development (Mexican Breast Cancer Case Study). 175
4.7 Discussion 182
4.7.1 Data processing 182
4.7.2 Conceptual modelling 183
4.7.3 Selection of algorithms 184
4.7.4 Data abstraction 185
4.7.5 Translating into DES model 186
4.7.6 Recommendation 187
4.7.7 Linking to the Kidney Case Study 188

CHAPTER 5 CONCEPTUAL MODEL AND PARAMETERS RESULTING FROM PROCESS MINING WITH THE UK KIDNEY DATASET 189

5.1 Chapter Overview 189
5.2 Background 189

5.2.1 Introduction to Peritoneal Dialysis 190
5.3 Aim of the case study 191
5.4 Dataset 192
5.4.1 Overview of the dataset 192
5.4.2 Variables in the dataset 194
5.5 Process mining 194
5.5.1 Data preparation 194
5.5.2 Process discovery 204
5.5.3 Process analyses 208
5.5.3.2 Assessment of the process model 209
5.5.4 Final process map 214
5.6 Results Interpretation and Clinical Significance 216
5.7 Value of Process Mining in Conceptual Model Development (UK Kidney Case Study) 218
5.8 Discussion 224

CHAPTER 6 COMPARISON OF TRADITIONAL SIMULATION MODELLING APPROACH VS (FULLY AUTOMATED AND EXPERT DRIVEN) PROCESS MINING APPROACHES 227

6.1 Chapter Overview 227
6.2 Introduction to Approaches 227
6.2.1 Traditional Simulation Modelling Approach 228
6.2.2 Expert Driven Process Mining Approach 230
6.2.3 Fully Automated Process Mining Approach 233
6.3 Expert Driven Process Mining Approach vs Fully Automated Process Mining Approach: Comparative Analysis using the Mexican Breast Cancer Dataset 234
6.3.1 Reiteration of Expert Driven Process Mining Approach 234

6.3.2 Fully Automated Process Mining Approach 235

6.3.3 Comparative Analysis of the Results of the Expert Driven Process Mining Approach vs Fully Automated Process Mining Approach 245

6.4 Expert Driven Process Mining Approach vs Traditional Approach to Simulation Modelling: Comparative Analysis using UK Kidney case study. 251

6.4.1 Expert-Driven Process Mining Approach 251

6.4.2 Traditional Conceptual Model by Collaborator 252

6.4.3 Comparative Analysis of the Results of the Expert Driven Process Mining Approach vs Traditional Simulation Modelling Approach 256

6.5 Fully Automated Process Mining using a Big Dataset 261

6.5.1 Challenges without expert input 261

6.6 General Comparison of the Different Approaches 262

6.6.2 Comparative Table Illustrating Key Differences 269

6.6.3 Discussion 270

CHAPTER 7 ENHANCED FINAL STEP-BY-STEP FRAMEWORK AND BEST PRACTICE CHECKLIST 272

7.1 Chapter Overview 272

7.2 Final enhanced step-by-step framework. 272

7.2.1 Framework Steps and Implementation in the Case Study 276

7.3 User Guidance to Apply the Enhanced Framework 278

7.3.1 Phase 1: Data Preparation 278

7.3.2 Phase 2: Process Mining (Conceptual Modelling) 281

7.3.3 Phase 2: Process Mining (Parameterisation) 284

7.3.4 Phase 3: Transforming to DES Model 286

7.3.5 Phase 4: Validation 287

7.4 Best practice checklist for integrating process mining in DES modelling. 288

CHAPTER 8 REFLECTIONS, FUTURE WORK, AND CONCLUSION 294

8.1 Key contributions 294

8.1.1 Developed and enhanced a step-by-step framework for integrating process mining into DES modelling. 294

8.1.2 Implemented the framework in multiple clinical case studies. 294

8.1.3 Comparison of process mining with existing traditional modelling approach. 295

8.1.4 Developed a best practice checklist for integrating process mining into DES modelling. 295

8.2 Reflections 296

8.2.1 Reflection on Objective 1: Literature review on integration of process mining in DES in past studies. 296

8.2.2 Reflection on Objective 2: Developed a Step-by-Step Framework. 297

8.2.3 Reflections on Objective 3: Perform case studies using the step-by-step framework. 298

8.2.4 Reflections On objective 4: The best practice checklist 299

8.3 Strengths 301

8.4 Limitations 301

8.5 Future work 302

8.6 Conclusion 303

REFERENCES 304

APPENDIX 1: SEARCH STRING 317

APPENDIX 2: LIST OF VARIABLES IN THE MEXICAN BREAST CANCER CASE STUDY. 320

APPENDIX 3: SOURCE CODE FOR PROCESS MINING, INCLUDING DATA PREPARATION, CONCEPTUAL MODELLING, AND PARAMETERISATION FOR THE MEXICAN BREAST CANCER DATASET. 321

APPENDIX 4: EXAMPLE OF EMPIRICAL DISTRIBUTIONS IN THE MEXICAN BREAST CANCER CASE STUDY. 325

APPENDIX 5: SOURCE CODE FOR PROCESS MINING, INCLUDING DATA PREPARATION, CONCEPTUAL MODELLING, AND PARAMETERISATION FOR THE UK KIDNEY DATASET. 328

APPENDIX 6: TRAINING SUMMARY 341

LIST OF TABLES

Table 2.1 Key terms in process mining 12

Table 2.2 Key terms used in discrete event simulation (DES) modelling 38

Table 2.3 Inclusion criteria 45

Table 2.4 Exclusion criteria 46

Table 2.5 Areas integrating process mining in DES modelling. 49

Table 2.6 Software used in past studies 54

Table 2.7 Algorithms used in past studies. 55

Table 2.8 Key Elements of Integrating Process Mining into DES Modelling 64

Table 4.1Sociodemographic characteristics 110

Table 4.2 Time Intervals and Health Service Use 110

Table 4.3 Perceived barriers 110

Table 4.4: Example of Variables and Their Corresponding Data Types 116

Table 4.5:Summary of Missing Values in Key Date Variables 117

Table 4.6 Issues relating to Mexican breast cancer dataset 119

Table 4.7 Brief information on types of subgroups based at presentation. 153

Table 4.8 Time between events for Public and Private services 155

Table 4.9 Comparing time interval results for public services. 166

Table 4.10 Comparing time interval results for private services. 166

Table 4.11 Comparison of patient outcomes in the current and new scenarios (time in system analysis) 168

Table 4.12: Example of patient, diagnostic, and treatment intervals with durations obtained from process mining and literature sources (Public Services). 173

Table 4.13: Example of patient, diagnostic, and treatment intervals with durations obtained from process mining and literature sources (Private Services). 174

Table 5.1 Demographics information 194

Table 5.2 Relevant variables 195

Table 5.3 Data cleaning 199

Table 5.4 Modality switches 202

Table 5.5 Comparison of time-to event between medical and surgical catheter insertion methods 212

Table 6.1: Steps in traditional simulation modelling approaches. 229

Table 6.2:Steps in expert-driven process mining approach. 232

Table 6.3: Steps in fully automated process mining approach. 233

Table 6.4: List of Event Conceptual Models Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study. 246

Table 6.5: Pathways Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study. 249

Table 6.6: Pathways that Differ Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study. 250

Table 6.7: List of Event Conceptual Models Generated by Expert-Driven Process Mining Approaches and Traditional Method using the UK Kidney Case Study. 257

Table 6.8: A comparative overview of the steps involved in the three approaches. 269

Table 7.1: Framework Steps and Implementation in the Case Study. 276

LIST OF FIGURES

Figure 2.1 Example of event log 13

Figure 2.2 Process discovery using event log 33

Figure 2.3 PRISMA diagram 48

Figure 2.4 Steps involved in process discovery 57

Figure 3.1: Existing frameworks by Rozinat et al. (2009) and Tamburis et al. (2020) 73

Figure 3.2:Initial Proposed Framework 89

Figure 3.3 Phase 1: Data Preparation Framework derived from literature. 91

Figure 3.4 Phase 2: Process Mining - Conceptual Modelling Framework 94

Figure 3.5 Parameterisation Framework Derived from Literature 96

Figure 3.6 Steps Involved in Transforming Process Models into DES Framework 99

Figure 3.7 Validation Framework 102

Figure 4.1 Phases of breast cancer patient pathways (adapted from Unger-Saldaña et al., 2020). 107

Figure 4.2 Different types of delay in breast cancer pathways (adapted from Unger-Saldaña et al., 2020). 108

Figure 4.3: Initial process model from symptoms discovery to treatment start. 126

Figure 4.4: Process model generated by the Alpha Miner algorithm (ProM Software). 126

Figure 4.5: Process model generated by the Heuristic Miner algorithm (ProM Software). 130

Figure 4.6: Process model generated by the Fuzzy Miner algorithm(ProM software). 133

Figure 4.7: Process model generated by the Inductive Miner algorithm (ProM Software). 136

Figure 4.8: Example of Variation in the Mexican Breast Cancer Case Study. 145

Figure 4.9: Example of event consolidation at the abstraction stage 146

Figure 4.10: Example of Process Model Abstraction at 20%, 50%, and 80% (Frequency Thresholds). 148

Figure 4.11: Example of Process Model Abstraction at 20%, 50%, and 80% (Time Thresholds). 149

Figure 4.12 Process maps for public and private healthcare services 156

Figure 4.13 Interarrival calculation 160

Figure 4.14 Calculation for activity distribution 162

Figure 4.15: Routing probability 163

Figure 4.16 Final process model from process mining integrated into DES modelling. 165
Figure 5.1 Peritoneal Dialysis 191
Figure 5.2 Example of event log for Peritoneal Dialysis 204
Figure 5.3 Initial process model for overall PD patients 207
Figure 5.4 Initial process model for PD patients with medical catheter insertion 207
Figure 5.5 Initial process model for PD patients with surgical catheter insertion 208
Figure 5.6 Final process model for PD patients with medical catheter insertion 214
Figure 5.7 Final process model for PD patients with surgical catheter insertion 215
Figure 6.1: Event Log- Sorted Chronologically from Earliest to Latest Date 236
Figure 6.2: Conceptual model using a fully automated process mining approach 237
Figure 6.3: Fully Automated Process Mining: Transforming Default Average Time to Event into a Frequency-Based Probability Distribution 239
Figure 6.4: Setting Up Interarrival Time for Fully Automated Process Mining. 240
Figure 6.5: Setting Up Routing Probability for Fully Automated Process Mining. 240
Figure 6.6 Algorithm Transparency and User Control 244
Figure 6.7: Mexican Breast Cancer Conceptual Model Generated by Fully Automated and Expert-Driven Process Mining Approaches. 245
Figure 6.8: Conceptual model presented in the paper 'Catheter Event Rates in Medical Compared to Surgical Peritoneal Dialysis Catheter Insertion' based on the traditional method 254
Figure 6.9: Conceptual Model by Expert-Driven Process Mining and Traditional Method 256
Figure 7.1:Enhanced Final Step-By-Step Framework. 273

LIST OF ABBREVIATIONS

ABM	Agent-Based Modelling
APD	Automated Peritoneal Dialysis
BPMN	Business Process Model and Notation
CAPD	Continuous Ambulatory Peritoneal Dialysis
CE	Catheter Event
CT	Computed Tomography scan
DES	Discrete Event Simulation
DWS	Disjunctive Workflow Schema
EHR	Electronic Health Records
HGMEL	General Hospital de Mexico Eduardo Liceaga
HD	Haemodialysis
ID	Identification
INCAN	Instituto Nacional de Cancerologia
KPI	Key Performance Indicator
MRI	Magnetic Resonance Imaging
PD	Peritoneal Dialysis
PDOPPS	International Peritoneal Dialysis Outcomes and Practice Study
PET	Positron Emission Tomography
PM	Process Mining
PRISMA	Systematic Reviews and Meta-Analyses
SD	System Dynamics
USD	United States Dollar
WHO	World Health Organisation

LIST OF PUBLICATIONS

Journal Paper

1. **Understanding Breast Cancer Patient Pathways and their impact on Survival in Mexico**
   Fatin Izzati Ramli[a,b] Praveen Thokala[a], Thaison Tong[a], Karla Unger-Saldaña[c]
   (Published in Journal of Cancer Policy)

2. **Using Process Mining to Estimate Time to Peritoneal Dialysis Initiation and Time to Infection – A Comparative Study between Patients undergoing Medical and Surgical Catheter Insertion.**
   Fatin Izzati Ramli[1,4], James Fotheringham[1,2], Mark Lambie[3], Louese Dunn[2], Simon Davies[3], Martin Wilkie[2], Thaison Tong[1], Praveen Thokala[1]
   (Completed draft)

3. **The Use of Process Mining to Support Discrete-Event Simulation: A Literature Review.**
   Fatin Izzati Ramli [1,2*], Praveen Thokala[1], Thai-Son Tong[1]
   (Completed draft)

Conference Proceedings

1. **Utilizing Process Mining with Large Synthetic Datasets: A Case Study on Chemotherapy Treatment for Breast Cancer**
   Fatin Izzati Ramli
   (Published in HELORS 2023 Proceedings: 9th International Symposium and 31st National Conference on Operational Research, University of West Attica, Athens, Greece, June 29–30 and July 1, 2023, pp. 1–4. ISBN 978-618-85079-2-0)

Conferences Abstract

1. **The use of data analytics in simulation modelling: A review**
   (Published in ORAHS 2022 Conferences Booklet)

2.Utilizing Process Mining Techniques for Chemotherapy Treatment: Simulacrum dataset
(Published in HELORS 2023 Conferences Booklet)

CHAPTER 1

INTRODUCTION

1.1 Chapter Overview

This chapter introduces the research topic, establishing the context for the thesis. It commences with a brief introduction to data analytics supporting simulation modelling, emphasizing the advantages of this integration. We subsequently present an introduction to the specific methods used in the research, focusing on integrating process mining and discrete event simulation (DES) modelling. Finally, the chapter outlines the research aim, objectives, research questions, scope, and limitations, as well as the layout of the subsequent chapters.

1.2 Background

The healthcare industry has become highly competitive, and organisations are under constant pressure to improve their processes in order to remain competitive. One widely used approach to achieve these goals is through the use of process modelling and simulation. These tools are widely regarded as essential for enhancing the effectiveness of healthcare processes (Mohammadi et al., 2018), identifying system bottlenecks (Chen, Chiang, & Storey, 2012), and eliminating inefficiencies (Prackwieser & Buchmann, 2013).

In this context, data analytics plays a vital role. It involves the application of various techniques and tools to extract valuable insights from healthcare databases (Mohammadi et al., 2018; Elgendy & Elragal, 2016; Kunc & Brien, 2019). Healthcare organisations can leverage advanced big data technologies to collect, store, and analyse large volumes of data from sources such as electronic medical records (Ajah & Nweke, 2019; Ohlhorst,

2012). This wealth of data presents significant opportunities to improve decision-making processes within the healthcare industry (Hitt & Wu, n.d.).

More specifically, simulation modelling is increasingly used in the healthcare sector. It involves creating digital representations of real-world healthcare processes or systems, enabling organisations to explore different scenarios and identify opportunities for improvement (Banks, Carson, & Nelson, 2010; Robinson, 2003).

Integrating data analytics with simulation modelling in healthcare offers substantial benefits. Most notably, data analytics facilitates the identification of hidden patterns and trends, thereby uncovering actionable insights and opportunities for process improvement (Collins et al., 2020). Furthermore, incorporating real-world data behaviour into simulation models enhances their accuracy and applicability.

Despite growing evidence supporting this integration, gaps persist in the existing literature. Specifically, there is a shortage of robust methodologies, structured strategies, and practical tools to effectively enable this integration (Greasley, 2018; Rodic, 2017). Consequently, additional research is essential to fully harness the advantages of combining data analytics and simulation modelling within the healthcare sector.

1.3 Introduction to Process Mining

As an analytical method, process mining uses timestamps from different events to construct a comprehensive process map. It relies on event logs data generated during the execution of clinical processes to extract insights that improve understanding of how processes are actually carried out (Murillas, 2019). Process mining enhances transparency by revealing the sequence of activities, the duration of each step, and the interdependencies between tasks. This, in turn, supports a deeper understanding of process dynamics and helps identify inefficiencies (Kovalchuk et al., 2018; Mans et al., 2013).

The process mining journey begins with event data collected from various sources, including electronic health records, administrative systems, and medical devices (Selig,

2017). This data is then compiled into a structured event log that captures the sequence of tasks, timestamps, and the individuals involved at each stage of a specific procedure.

One of the key strengths of process mining is its ability to uncover the actual ('as-is') state of processes. Process models are generated using visualisation tools that illustrate the real flow of activities, decision points, and variations in performance. By comparing these real-world models with ideal or expected workflows, healthcare professionals can identify deviations, bottlenecks, and inefficiencies that may hinder optimal care delivery (Martin et al., 2015; Ito et al., 2020; Gatta et al., 2020).

Through in-depth analysis, process mining helps uncover patterns and trends that might otherwise go unnoticed. For example, it can highlight prolonged waiting times for certain procedures or treatments, enabling hospitals to manage resources more effectively and improve patient care. Additionally, process mining supports the identification of targeted interventions to enhance service delivery. These improvements can lead to shorter waiting times, reduced administrative burdens, and ultimately, increased patient satisfaction and better health outcomes.

1.4 Introduction to Discrete Event Simulation Modelling

Discrete event simulation (DES) modelling is a technique used to develop computer-based representations of real-world processes, which can then be analysed and optimised. It enables healthcare professionals to model the flow of patients, resources, and activities within complex systems, allowing them to identify bottlenecks, evaluate alternative strategies, and make informed decisions to enhance patient care and improve operational efficiency (Buss & Rowaei, 2010; Brailsford & Hilton, 2001).

In clinical settings, discrete event simulation (DES) models capture the changes that occur at specific time points as patients move through different phases of care. These models rely on data such as patient arrival patterns, treatment durations, resource availability, and the rules governing process interactions (Hamrock et al., 2013). By running multiple replications of the simulation, healthcare professionals can observe how various factors influence overall system behaviour and performance outcomes.

Discrete event simulation (DES) modelling offers numerous advantages in clinical settings. It enables healthcare providers to explore 'what-if' scenarios without disrupting real-world operations, providing insights into patient flow dynamics and identifying potential areas for improvement (Karnon et al., 2012). A common application of DES in healthcare is optimising patient flow. For instance, by simulating processes such as patient admission, diagnosis, treatment, and discharge, healthcare administrators can assess the impact on key outcomes such as waiting times and patient throughput and make informed decisions about process redesign. Moreover, DES is a valuable tool for analysing the effects of proposed changes before they are implemented in practice (Lord et al., 2013; Lim et al., 2019). For example, if a hospital plans to modify its patient waiting time policies, simulation modelling can forecast how such changes might affect patient care and operational efficiency.

1.5 Problem statement

Discrete event simulation (DES) modelling, a valuable tool for analysing complex systems and processes, can be effectively complemented by process mining techniques.

Creating an accurate simulation model from scratch can be time-consuming. Process Mining can accelerate this process by automatically discovering process flows and patterns from event logs. These discovered 'conceptual' models can serve as initialization basis for the simulation model, reducing the effort required to build the simulation from the ground up. DES models are often created based on expert opinions, and limited data. Process mining, on the other hand, uses actual event logs from process executions to provide a more accurate representation of how processes are really carried out. This real-world data-driven approach ensures that the model reflects the actual variations and deviations that occur in practice.

Moreover, DES often requires assumptions or estimates for input parameters such as arrival rates and service times. These assumptions can introduce uncertainties and inaccuracies into the simulation model. Process mining can provide these parameters from the real-world event data to validate or refine these assumptions, resulting in more accurate simulation outcomes.

Additionally, validating complex DES models can be challenging due to their intricate nature. Process mining can serve as an external validation tool by comparing simulation results to historical data, ensuring that the model aligns with observed behaviours.

1.6 Aim of research

The aim of this thesis is to explore how process mining can be integrated to support DES (discrete-event simulation) modelling. To achieve this goal, we have identified several specific objectives that will guide our research.

1.6.1 Specific objectives

- To study the integration of process mining in DES in the past through a comprehensive review of relevant literature.
- To develop a step-by-step framework for integrating process mining into DES modelling.
- To perform case studies (i.e. practical applications of the integrated process mining and DES modelling) using the proposed framework to identify opportunities and challenges. The case studies will also inform the refinement of the framework. Also, the results from the process mining approach will be compared with traditional simulation modelling.
- To create a best practice checklist that can guide the use of process mining to support DES modelling.

1.6.2 Research questions

- How have previous studies used process mining in DES modelling?
- What are the key steps for integrating process mining in DES modelling?
- What are the issues arising in practical implementation of integrating process mining in DES modelling?

- How can the proposed framework be improved based on the experiences gained from the case studies?
- What are the essential best practices for using process mining in DES modelling?

1.6.3 Scope of the PhD study

In this PhD study, the scope of integrating process mining in DES modelling is limited to DES modelling without incorporating constraints, as the datasets used in the case studies lack resource capacity information. Therefore, capacity modelling is not within the scope of this research. Additionally, this study specifically focused on process discovery within process mining, and whilst process mining can include other aspects (e.g. conformance checking and performance enhancement), these were not included within the scope as they were not considered relevant to achieving the objectives of this research.

1.7 Thesis layout

This section presents an outline of the chapters comprising this thesis. It provides a brief description of the content and structure of each chapter, highlighting the key topics addressed in the research.

1.7.1 Chapter 1: Introduction

The current chapter (Chapter 1) starts with a brief introduction to the fundamental concepts of process mining and discrete event simulation modelling. Then, the problem statement is presented, identifying the specific area of focus and the challenges to be addressed. The aim of the research is then outlined, emphasizing the objectives and research questions that the study aims to address, along with the scope of the study.

1.7.2 Chapter 2: Process Mining and DES Modelling Methodologies: Concepts and Literature Review.

Chapter 2 of this thesis explores the methodologies of process mining and discrete event simulation (DES). Each method is described in detail, including essential terminologies, inputs, outputs, and the steps involved. Notably, significant attention is given to the integration of process mining and DES modelling, a pivotal aspect aligned with the study's objectives.

Additionally, the chapter includes an extensive literature review on studies integrating process mining in DES modelling. Our data extraction primarily focused on conceptual modelling, parameterisation, and validation within this integration. We present and discuss the findings and insights derived from this literature review. Furthermore, we identify gaps in existing research that warrant further investigation and discuss these in the concluding section of this chapter.

1.7.3 Chapter 3: Initial Step-By-Step Framework

Chapter 3 provides a comprehensive guide to the methodologies and structured steps involved in integrating process mining and discrete event simulation (DES) modelling. The chapter begins with an overview, providing a foundation for a detailed exploration of the ‘initial’ step-by-step framework derived from a thorough literature review. It then describes the framework, outlining the four essential phases for integrating process mining and DES modelling: data preparation, process mining, process model transformation to DES modelling, and validation.

1.7.4 Chapter 4: Exploring Usability of Process Mining in DES Modelling Using Breast Cancer Dataset

Chapter 4 of this thesis focuses on the integration of process mining in discrete event simulation (DES) modelling using a Mexican breast cancer dataset from Mexico. Process mining was used to understand the journey of breast cancer patients from symptom discovery through the start of treatment, and to investigate for differences in pathways and time delays between patients who went to public services versus those went to private services. The final process maps were integrated into a DES model to simulate patient

movement and to perform scenario analyses estimating the impact of reducing delays on diagnostic intervals and time to treatment initiation in breast cancer pathways.

In this chapter, we provide an in-depth exploration of process mining to support conceptual modelling, parameterisation, and validation of the simulation model. We discuss the steps taken to develop the simulation model, including the identification and selection of relevant process mining techniques for data extraction and analysis. We also address the challenges associated with data quality and the measures taken to ensure the accuracy and reliability of the model.

1.7.5 Chapter 5: Conceptual Model And Parameters Resulting From Process Mining With The UK Kidney Dataset

In this chapter 5, we use the proposed step-by-step framework on a peritoneal dialysis dataset to derive the conceptual model and model parameters. Process maps provide a visual representation of the sequential steps and activities involved in the PD treatment process, facilitating a better understanding of patient pathways and the overall flow of the treatment. The dataset enables an examination of the differences between the medical catheter insertion technique and the surgical catheter insertion technique, in particular the time taken to initiate PD dialysis and the time to infection. Analysing infection rates and durations for each group allows for gaining insights into the potential impact of the insertion technique on the occurrence and timing of infections, thereby informing clinical decision-making.

In this chapter, we provide an in-depth exploration of process mining to support conceptual modelling and to estimate model parameters. Subsequently, we compared the process model generated through process mining with the existing conceptual model, as described in in chapter 6.

1.7.6 Chapter 6: Comparison of Traditional Simulation Modelling Approach Vs (Fully Automated And Expert Driven) Process Mining Approaches

In chapter 6, we perform a comparative analysis of the traditional approach to develop DES models to our (expert-driven) process mining approach, as well as to a fully automated process mining approach. The comparative assessment encompasses results from chapters 4 and 5, involving conceptual models, pathways, and parameters. Chapters

4 and 5 describe expert-driven process mining approach, which involves fusion of process mining techniques and expert opinions, achieved by manually integrating process mining into DES modelling through Disco software and Simul8. We compare outputs (i.e. conceptual models, pathways, and parameters) from chapter 4 with those created through fully automated process mining in Simul8 without any expert input. We compare the conceptual model and parameters from chapter 5 with the conceptual model developed using traditional approach (without process mining). Following this, we present a comparative analysis between the different approaches in terms of the time/resources, skills required and the acceptability of each of the approaches.

1.7.7 Chapter 7: Enhanced Final Step-By-Step Framework and Best Practice Checklist

Chapter 7 presents the final framework derived from the literature review and further enhanced by the findings from two case studies. Furthermore, the chapter includes a best practice checklist that serves as a reference for future researchers interested in integrating process mining in DES modelling. The checklist outlines the essential steps and considerations to ensure a successful integration process. Lastly, the chapter presents a discussion on the key elements of the framework, highlighting the findings and insights gained from the literature and case studies. The discussion provides an analysis of the strengths, limitations, and potential areas for improvement of the framework.

1.7.8 Chapter 8: Reflections, Future Works and Conclusion

Chapter 8 serves as a reflection on the achievements made in this PhD study. The chapter begins by revisiting the objectives set at the beginning of the study and discussing the findings and outcomes obtained throughout the research. Moreover, this section discusses the key contributions of this research. Furthermore, the chapter identifies potential future work and areas for further research. It discusses potential extensions of the current study and emerging trends in the field that warrant further investigation. Finally, the chapter concludes by summarizing the key findings, contributions, and implications of the research.

CHAPTER 2
PROCESS MINING AND DES MODELLING METHODOLOGIES: CONCEPTS AND LITERATURE REVIEW

2.1 Chapter Overview

This chapter lays the groundwork by introducing the methodologies of Process Mining and Discrete Event Simulation (DES) Modelling. Additionally, a literature review reveals research gaps that relate to integrating these methodologies.

2.2 Process mining Methodology

This section offers a detailed description of the process mining methodology. We start by describing the basics of process mining, its purpose, advantages, and limitations. Second, we define the key terminologies associated with process mining, including events, traces, and process models. In order to conduct the process mining analysis, inputs are necessary, such as event logs, and any other information related to the process under analysis.

Process mining methodology is composed of several steps including data cleaning and transformation and the use of algorithms to obtain process knowledge from the event logs. We describe these in detail and show how to prepare them for analysis. Lastly, we discuss the outputs of the process mining methodology such as process models, performance measures, and process improvement recommendations. Understanding these outputs is crucial to interpreting and applying the results of the process mining analysis effectively.

Given that the main objective of the research was to investigate the integration of process mining in DES modelling, the focus was primarily on process discovery. Conformance checking and enhancement, which are separate aspects of process mining that involve assessing compliance and improving processes, are not within the scope of the PhD. In order to align with the research objectives, the decision was made to concentrate solely on process discovery and not delve into conformance checking and enhancement in this particular study. As such, these aspects are not discussed in detail in the thesis.

2.2.1 Overview of process mining

Process mining is a process methodology that allows for extracting useful information and knowledge from event logs, which are the records of events that have actually taken place in the system, including the timing, the person involved, and the data processed (W. van der Aalst, 2016; W. M. P. van der Aalst, 2011; W. van der Aalst et al., 2004). The primary objective of process mining is to uncover information that is not immediately apparent from the event logs.

Process mining involves three primary types: process discovery, conformance checking, and process improvement. The selection of the process mining type is based on the specific problem to be solved and the characteristics of the process. All three types of process mining are aimed at different elements of the process and employ various tools and techniques of data analysis.

Discovery is focused on understanding the process structure and sequence of activities. It is used to generate a process model from event logs or other sources of data (W. van der Aalst, 2016; A. Rozinat et al., 2009). Conformance is concerned with the recognition of discrepancies from the actual behaviour of a process defined in the process model. It is used to ensure that the process is being performed as per the plan and to propose changes to the model wherever any deviations are noticed (A. Rozinat et al., 2010; W. M. P. van der Aalst, 2011). The enhancement is process-specific and aims to find process-related problems and areas for improvement. This is used to improve the process and modify the process model or behaviour to enhance performance and results (W. van der Aalst, 2016).

This would ensure that better results are obtained because the different types of process mining, their objectives, and approaches would be understood and hence one would be able to select the most appropriate methods. As described earlier, the focus of the thesis is on process discovery and, as such, conformance checking and enhancement are not within the scope of the PhD. The key terminology in process mining is described in Table 2.1.

Table 2.1 Key terms in process mining

Terminology	Definition
Case	A unique execution of a process, represented by a sequence of events in a specific order for a particular entity or object.
Event	A specific occurrence or activity in a process, captured in the event log. It includes information such as timestamp, activity name, resource, and related data.
Trace	Another term for a case. It represents a sequence of events that occur in a particular order for a specific entity or object.
Resources	Individuals or entities involved in the execution of a process, such as employees, machines, or systems. Resources perform activities and can be assigned to specific events or tasks in the event log.
Process Discovery	The automated generation of process models from event logs. It involves analysing event data to uncover the underlying process flow and relationships between activities. The goal is to create a visual representation of how processes are executed.

2.2.2 Inputs to process mining.

Process mining involves the analysis of event data or logs that record the sequence of activities and associated data for a given process. These logs can be sourced from various platforms such as databases, enterprise systems, or even application logs, and must be complete and unbiased in order to produce accurate analysis and process models.

An event log is a structured collection of all activities that occurred in a specific process over a given period of time. It contains detailed information such as the start and end times of activities, the users involved, and the relevant properties such as type of activity

or resource allocation. Each event record in the log represents a specific activity that occurred at a particular time (Turner et al., 2012).

In the healthcare system, event logs are a collection of events that have taken place in relation to patient care. These records detail actions or procedures, including diagnostic tests, drug distribution, or healthcare provider encounters (Fernandez-Llatas et al., 2021). Event logs are found in different systems such as EHRs, clinical information systems, or other healthcare databases. The information in the event log can be used to build process models, analyse healthcare processes, and find areas for improvement.

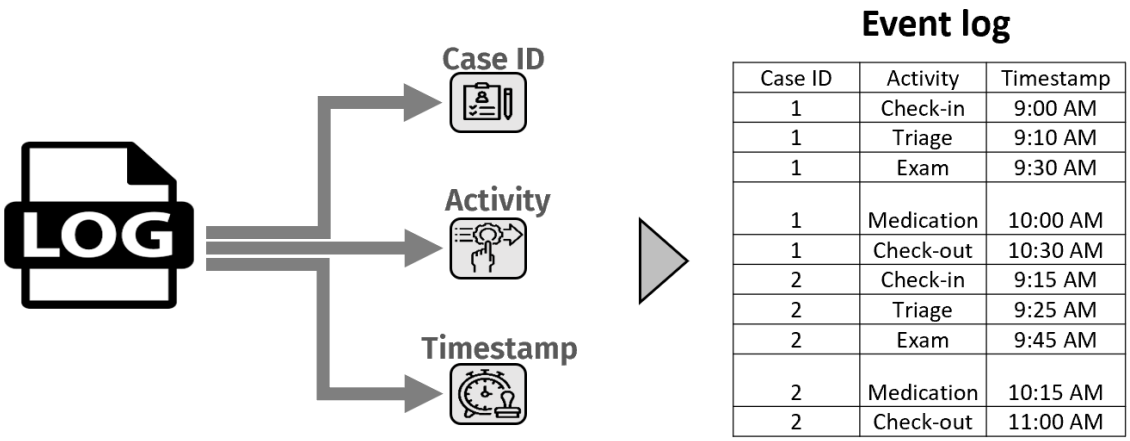


Figure 2.1 Example of event log

2.2.3 Steps in Process mining

This chapter provides an overview of the process mining steps, including data preparation (also known as pre-processing), process discovery, conformance checking, and process improvement. Although the particular methods of conformance checking and process improvement were not implemented in this PhD study, a general overview is presented to provide a comprehensive understanding of the concept of process mining as a whole.

2.2.3.1 Pre-processing

Data pre-processing is an essential step of process mining aimed at preparing collected data for further analysis. This is because the data may be unclean, inaccurate, or even missing information. It is important because the data may contain errors, noise, or irrelevant information. The goal of this step is to ensure that the data is cleaned, filtered,

and transformed before being input into the process mining algorithms to obtain accurate and reliable results.

The data pre-processing step includes its own set of sub-steps, which are data cleaning, data transformation, data integration, and data reduction. Data cleaning refers to the process of removing inaccuracies and errors (Cho et al., 2019; Zhou et al., 2014). For example, a dataset of medical records may include entries with incomplete information such as age or blood pressure readings. In such cases, these missing values can be addressed using imputation techniques like mean imputation or regression imputation, which involve estimating the missing values based on other available data.

Data transformation is a crucial step that involves converting the collected data into a standard format that is suitable for analysis. This may also include data type conversions, rescaling of the attributes, as well as normalization to guarantee uniformity within the dataset. This ensures that the results of analysis are accurate and reliable.

In addition, data integration is the process of combining data from different sources or databases into one consolidated dataset, enabling comprehensive analysis and informed decision-making. On the other hand, data reduction focuses on reducing either the size or complexity of a dataset while preserving important details in it, thus leading to improved efficiency and decreased redundancy. Both data integration and data reduction play vital roles in data management and analysis, enhancing data quality and facilitating valuable insights from large and complex datasets.

2.2.3.2 Process discovery

The process discovery phase in process mining involves the generation of an automated process model using event logs. These event logs contain valuable data about the actions performed within the process, including details such as the time, activities, and utilized resources. Leveraging this data, process mining tools generate process models that visually represent the sequence of actions and their relationships.

Algorithms

Process mining algorithms are used to create process models automatically from event logs. There are several algorithms for process discovery under process mining, and they have their strengths and limitations. The common algorithms used in process mining are Alpha, Heuristic Miner, Inductive Miner, Fuzzy Miner and Genetic Miner(W. Van Der Aalst, 2016). There is a need to select the most suitable algorithm for the specific process being analysed since each of these algorithms uses a different approach with its own level of complexity.

2.2.3.2.1 Alpha Miner

The Alpha Miner is a foundational algorithm in process mining designed to discover process models, typically represented as Petri nets, from event logs (van der Aalst, Weijters, & Maruster, 2004). It focuses on uncovering the control-flow perspective by analysing the order and relationships between activities in a process. Using key rules such as direct follows, causality, parallelism, and choice, the Alpha Miner identifies patterns like sequences (strict order between activities), parallel activities (tasks that can occur concurrently), decision points (alternative pathways), and loops (repetitive behaviours). The algorithm is particularly suitable for structured processes with clear, well-defined workflows, as it relies on the consistency and completeness of event logs. However, the Alpha Miner has limitations, including its sensitivity to incomplete logs, inability to handle complex loops effectively, and challenges in discovering unstructured processes.

Key Rules in Alpha Miner

The Alpha Miner algorithm identifies relationships between activities in an event log using a set of key rules. These rules include direct succession, causality, parallelism, and unrelatedness. These rules form the foundation for constructing process models, such as Petri nets, that capture the control-flow perspective of a process.

1. **Direct Succession**: This rule identifies when one activity directly follows another in a trace. For example, in a trace $A \rightarrow B \rightarrow C$, $A \rightarrow B$ and $B \rightarrow C$ are direct

successions. Direct succession is the most basic relationship and serves as the foundation for identifying more complex patterns like causality and parallelism.

2. **Causality**: Causality describes a directional dependency between activities. If $A \rightarrow B$ is observed in the event log but $B \rightarrow A$ is not, A is said to causally lead to B. For instance, in a clinical workflow, if *Triage* → *Diagnosis* is always observed but *Diagnosis* → *Triage* is not, the causality *Triage*→*Diagnosis* is established. This ensures that the flow of activities in the process model reflects the correct order of dependencies.

3. **Parallelism**: Parallelism occurs when two activities can happen concurrently or in no specific order. This is detected when both $A \rightarrow B$ and $B \rightarrow A$ appear in different traces, indicating that the activities are independent. For example, in a clinical setting, *Blood Test* and *Imaging* may be conducted in parallel, as their order varies in different traces (*Consultation* → *Blood Test* → *Imaging* vs. *Consultation* → *Imaging* → *Blood Test*).

4. **Unrelatedness**: Two activities are considered unrelated if no direct succession, causality, or parallelism exists between them. This means the activities occur in separate parts of the process or belong to entirely different workflows. For example, in healthcare, *Prescription* → *Medication Administration* and *Patient Education* → *Discharge* might be unrelated if these pairs belong to distinct clinical pathways.

Construction of process model in Petri Net

A Petri Net is a graphical and mathematical tool used to model and analyse processes, particularly those involving concurrency, synchronization, and dependencies. It provides a structured way to represent how activities in a system interact and progress over time.

At its core, a Petri Net consists of three main components: places, transitions, and tokens. Places (represented as circles) indicate conditions or states within the system, while transitions (represented as rectangles) represent activities or events that change those states. Tokens, which reside in places, signify the availability of resources or conditions needed to enable an activity.

When a transition is "enabled" meaning all required input places contain tokens, it fires, consuming tokens from the input places and generating tokens in the output places. This movement of tokens simulates the dynamic behaviour of a process, capturing various execution flows:

- **Sequential execution**, where one activity follows another in order.
- **Parallelism**, allowing multiple activities to occur simultaneously.
- **Decision-making**, where different pathways can be taken based on conditions.
- **Synchronization**, requiring multiple inputs before an action can proceed.

For example, in a healthcare workflow, a Petri Net can model the journey of a patient through different medical procedures. A place like *"Patient Registered"* holds a token once registration is complete. This token enables a transition, such as *"Blood Test"*. Once the blood test is conducted, the token moves to a new place, *"Blood Test Results Ready"*, indicating that the next step (e.g., a doctor's consultation) can occur.

2.2.3.2.2 Heuristic Miner

Heuristics Miner builds process models by considering not only the direct sequence of events but also the frequency of those sequences. It looks for the most frequent patterns in the event log and models them as the main process flows, while also allowing for some level of noise and exceptions (Weijters, van der Aalst, & Alves de Medeiros, 2006). The primary strength of Heuristics Miner lies in its frequency-based approach. By focusing on the most commonly observed paths within an event log, it builds a process model that captures the main workflows while filtering out low-frequency or rare events as noise. This allows Heuristics Miner to create simplified, yet accurate, representations of processes that might otherwise be too complex or "messy" to visualise effectively.

Key rules in Heuristics Miner

1. **Direct Succession**: The Heuristic Miner scans the event log and records pairs of activities where one activity consistently follows another. This direct succession is essential for understanding the order in which activities occur. If activity A often precedes activity B, the Heuristic Miner will recognise and capture this relationship, symbolized as A → B. This succession relationship helps build a process map showing how activities transition from one to the next. For example, *Patient Admission → Initial Consultation* the Direct Succession relationship is between "Patient Admission" and "Initial Consultation," indicating that, for most patients, an initial consultation directly follows admission.
2. **Frequency-Based Relationships** in the Heuristics Miner algorithm allow it to identify and prioritize relationships between activities based on how frequently they occur in the process. Unlike the Alpha Miner algorithm, which doesn't consider the frequency of events, Heuristics Miner assigns weights to each relationship. This helps to emphasize more common sequences and downplay less frequent or rare ones, creating a process model that reflects the primary pathways.
3. **Thresholding:** Heuristic Miner applies thresholding by filtering out paths that occur below a defined frequency threshold. This process removes infrequent pathways while retaining more significant ones, ensuring that only the most relevant patterns are included in the overall process model.
4. **Dealing with Loops:** The Heuristic Miner is capable of handling certain types of loops effectively, particularly those that involve long-distance patterns, while it struggles with short, closely repeating loops.
5. **Parallelism:** Heuristic Miner detects parallelism more reliably than Alpha Miner, which struggles with incomplete logs and complexities. Alpha Miner identifies parallelism when two activities appear in different orders across traces. For example, if A precedes B in one trace (A→ B) and B precedes A in another (B→ A), it concludes they can occur in parallel. While effective for clean logs, this method fails when sequences are missing, or noise introduces spurious dependencies.

Construction of process model in Causal-Net (C-Net)

The Heuristic Miner algorithm uses Causal-Net (C-Net) models to represent process flows. Causal-Nets are a specific type of process modelling technique designed to capture causal relationships, parallel activities, and repetitive patterns. Unlike Petri Nets, which are limited in representing certain types of parallelism (specifically, they struggle to model OR-parallelism), C-Nets are more flexible and can model different kinds of parallel behaviours. However, even with C-Nets, the existing Heuristic Miner can only model XOR (exclusive) and AND (parallel) splits and joins, which limits its flexibility in representing complex business processes where OR relationships are needed.

Causal Nets (C-nets) are a graph-based modelling framework designed specifically for process discovery. They provide a clear representation of the causal dependencies between activities in a process, overcoming some limitations of traditional models such as Petri nets and Business Process Model and Notation (BPMN). Unlike these models, C-nets focus solely on the direct relationships between activities without the need for intermediate constructs like places in Petri nets. This makes them more concise and tailored for analysing process behaviours extracted from event logs.

In a C-net, activities are represented as nodes, while causal relationships between activities are depicted as directed arcs. Each activity has associated input and output bindings that define its preconditions and postconditions. Input bindings describe the sets of activities that must have occurred for an activity to begin, while output bindings describe the possible sets of activities that may follow upon its completion. For instance, an activity a might have input bindings, indicating that it is a start activity, and output bindings such as { {b, d}, {c, d} }, meaning the process may proceed to b and d or c and d. This representation provides an intuitive way to model both sequential and parallel behaviour in processes.

- Nodes: Represent activities in the process.
- Arcs: Represent causal dependencies between activities.
- Bindings: Each activity has:
    1. A set of input bindings (conditions for the activity to start).

2. A set of output bindings (conditions resulting after the activity finishes).

Characteristics of C-nets:

Input and Output Bindings.

Causal Nets (C-nets) use input and output bindings to define activity dependencies in a process. Input bindings specify the conditions required for an activity to be enabled by listing all preceding activities that must occur. Output bindings define the possible activities that can follow once an activity is completed, allowing for branching in the process. For example, if activity A has output bindings {b, d} and {c, d}, it means that after A, the process can either proceed to B and D or C and D. Similarly, if activity E requires both B and D to occur first, its input binding is {b, d}.

Routing Logic

Causal Nets (C-nets) differ from Petri nets by not using explicit places to model routing logic. Instead, the flow of the process is determined by the input and output bindings of activities. These bindings inherently encode the routing behaviour, such as choices and concurrency, without requiring additional structural elements.

For example, an XOR-split is represented when an activity has multiple independent output bindings, allowing the process to proceed to one of several possible paths. Similarly, an AND-split is modelled when the output binding specifies that multiple activities must occur simultaneously after an activity completes. This approach simplifies the visualisation of routing logic while still capturing the process behaviour. By embedding routing decisions directly within the bindings, C-nets provide a more compact representation compared to Petri nets. However, this can also make the routing logic less explicit, which might require additional effort to interpret for complex processes.

Start and End Activities

In Causal Nets (C-nets), the start activity (e.g., a) has no input bindings, meaning it initiates the process without any prior activities. The end activity (e.g., z) has no output

bindings, marking the conclusion of the process with no subsequent activities. These features clearly define the beginning and end of the workflow.

Bindings as Sets

In Causal Nets (C-nets), bindings are represented as sets of sets, defining the conditions required for an activity to be executed. Each set within a binding specifies a combination of activities that must occur for the next activity to begin. For example, the input binding for activity e might be { {b, d}, {c, d} }. This indicates that e can only start if one of the two combinations has occurred: either b and d together or c and d together. This structure allows C-nets to capture complex dependencies and pathways in process models, providing flexibility in representing alternate combinations of preceding activities.

2.2.3.2.3 Fuzzy Miner

The Fuzzy Miner is a process mining algorithm designed to handle complex, "fuzzy" process models often found in real-world event logs with high variability and noise. Traditional process mining algorithms, like Alpha Miner or Heuristics Miner, aim to generate exact process models that capture each recorded activity and pathway in detail. However, when applied to highly variable processes, these algorithms can produce overly complex models, often called "spaghetti models," which are difficult to interpret and analyse (Günther & van der Aalst, 2007).

To address this, the Fuzzy Miner takes a different approach. It uses significance and correlation metrics to simplify and abstract the model, focusing on the most important activities and paths while filtering out less significant ones.

Fuzzy Miner Mechanism: Significance and Correlation Metrics

Fuzzy Miner builds its footprint by analysing two main metrics for activities and their connections:

- **Significance**: This metric measures how often an activity occurs or how frequent a specific path between two activities is. High-significance

activities or paths are considered important, while low-significance elements are candidates for removal or abstraction.

- **Correlation**: This measures how strongly activities are related. If two activities frequently occur together (direct succession), their correlation is high. Strongly correlated activities are more likely to appear as connected in the process model.

The Fuzzy Miner algorithm generates a simplified process footprint by analysing two primary metrics for activities and their connections: significance and correlation. These metrics allow Fuzzy Miner to highlight essential paths and relationships in a process while minimizing or removing less important elements, making it easier for analysts to focus on the main structure of complex workflows.

The significance metric measures how often an activity occurs individually or how frequently a specific path appears between two activities. In essence, significance highlights activities and connections that play a prominent role in the process. High-significance activities are those that recur frequently and form the backbone of the workflow. These activities or paths are considered essential and are retained in the model to ensure a complete view of the primary process. On the other hand, activities or connections with low significance are deemed less critical to the overall flow and are candidates for abstraction or removal. For example, in a hospital setting, activities like "Patient Registration" or "Initial Triage" may have high significance because they occur with every patient, whereas rare or specialized procedures may have low significance and may be hidden in the final footprint to simplify the visualisation.

The correlation metric assesses the strength of the relationship between activities, specifically how often they occur together in sequence, or "direct succession." If two activities have a high correlation, it suggests that they frequently follow each other in the process, indicating a strong dependency or common flow between them. These highly correlated activities are then likely to be represented as connected in the process model. For instance, in a healthcare workflow, if "Initial Triage" is often followed by "Doctor Consultation," there would be a high correlation between these two activities, and Fuzzy Miner would likely display a direct connection between them in the model. Conversely, if two activities rarely occur in sequence, they would have a low correlation and may not

be represented as connected in the footprint, reducing clutter and emphasizing only the critical connections.

Together, the significance and correlation metrics enable Fuzzy Miner to create a footprint that is both streamlined and informative. By focusing on high-significance activities and paths and emphasizing strong correlations, the algorithm captures the primary process flows and core relationships. This approach makes it easier for users to identify important patterns and optimise workflows without being overwhelmed by the noise of infrequent or weakly related activities. In complex environments, such as healthcare, this helps stakeholders focus on the central patient pathways and dependencies, supporting effective process analysis and decision-making.

The Fuzzy Miner captures pathways directly from the event log without enforcing strict, rule-based patterns like AND, XOR, or OR. Instead, it relies on correlation and significance metrics to identify and highlight the most important activities and pathways. the Fuzzy Miner can represent patterns that resemble AND, OR, and XOR behaviours, but it does so in a flexible way based on significance and correlation, rather than using strict, rule-based interpretations of these patterns as other algorithms (like Alpha Miner) do.

Fuzzy Miner takes the pathways from the event log "as is," meaning it does not impose a strict logical structure. It observes how frequently and strongly activities are related to one another, rather than enforcing predefined relationships or control-flow structures.

Significance (frequency of activities and paths) and correlation (strength of relationships between activities) provide a flexible way to analyse complex processes that not follow strict, rule-based patterns.

Unlike Alpha Miner and Heuristics Miner, which strictly enforce structures like AND, XOR, and OR, Fuzzy Miner’s reliance on significance and correlation allows it to handle “spaghetti-like” processes by filtering out rare and less important pathways, thus preventing the model from becoming overly complex. This adaptability enables the Fuzzy Miner to simplify processes in a way that rule-based methods can’t, making it effective for visualizing highly variable processes.

Flexible and Adaptive Visualisation, by adjusting the significance and correlation thresholds, Fuzzy Miner allows for a customizable level of detail, making it easier to simplify or focus on specific areas without needing rigid rules to define each pathway.

Abstraction in the Fuzzy Miner

The abstraction mechanism in Fuzzy Miner's footprint allows analysts to view a process at different levels of detail based on their objectives, making it an adaptive tool for simplifying complex workflows. This mechanism is especially useful in scenarios where large volumes of activities and connections would otherwise create overwhelming visual clutter

Aggregation is one of the main techniques Fuzzy Miner uses for abstraction is collapsing activities. When multiple activities are closely related and occur in consistent patterns within the process, Fuzzy Miner can combine these into a single high-level activity node. This method significantly reduces the number of individual nodes in the footprint, making it easier for analysts to understand the overall process without being distracted by minor details. For example, in a hospital discharge process, activities like "Prepare Discharge Summary," "Finalize Medication Instructions," and "Issue Discharge Forms" might be collapsed into a single "Discharge Processing" node. By grouping these steps, Fuzzy Miner allows analysts to observe the general flow of patient discharge without diving into each sub-step, making the process model more concise and focused.

Additionally, Fuzzy Miner employs abstract relationships to reduce visual complexity. Rather than displaying every possible connection between activities, which can clutter the model, Fuzzy Miner uses correlation-based filtering to show only the most relevant and frequently occurring paths. Connections that are rare or insignificant to the main process flow are either displayed with weak or dashed lines, signalling their low importance, or omitted altogether if they fall below a certain threshold. This selective visualisation ensures that the primary, high-frequency pathways are emphasized, helping analysts focus on core connections and dependencies within the process. For instance, in a patient flow model, the frequent transition from "Triage" to "Consultation" might be shown as a strong, bold line, while a rare path, like a direct route from "Triage" to "Specialist Procedure," could appear faintly or be excluded entirely. By abstracting these

relationships, Fuzzy Miner offers a clear and streamlined view that prioritizes the most important process connections, facilitating more effective analysis and decision-making.

Another technique Fuzzy Miner uses to enhance clarity is emphasis. In a process model, certain activities or pathways are often more critical due to their frequency or significance within the workflow. Fuzzy Miner visually emphasizes these key elements by adjusting attributes like line thickness, colour, and contrast. Important activities and frequently travelled paths are depicted with bold or brightly coloured lines, making them stand out in the model. For example, in a patient journey map, the transition from "Emergency Admission" to "Initial Triage" might be shown as a thicker line, underscoring its central role in the emergency department workflow. Emphasis allows analysts to identify and focus on high-impact areas within the process quickly, which can help prioritize areas for improvement or identify potential bottlenecks.

Fuzzy Miner also supports customization, allowing analysts to tailor the level of detail in the model based on their specific objectives or the context of the analysis. By adjusting thresholds for significance and correlation, users can control how much detail is displayed in the process model. This is similar to choosing different map scales; a highway map shows only major routes, while a city map shows local streets. In process mining, customization means an analyst could use a high threshold to view only the core activities and main flows, ideal for executive-level summaries. Alternatively, they could lower the threshold to reveal more detailed paths and rare activities if they are conducting an in-depth analysis of specific cases. For instance, in a hospital setting, one might choose to focus on the typical patient flow paths with a high threshold or lower it to examine rare events, such as emergency escalations, when analysing response effectiveness.

Tunable Simplification

The tunable simplification feature in Fuzzy Miner allows analysts to adjust the level of detail in the process footprint by modifying the significance and correlation thresholds. These thresholds determine which activities and connections appear in the model based on their frequency and relational strength, making it possible to customize the visualisation according to specific analytical needs. This flexibility helps analysts focus on either high-level summaries or detailed process insights, depending on the objectives of the analysis.

- **Higher Thresholds**: Setting higher thresholds for significance and correlation results in a simplified footprint that displays only the most frequent and strongly correlated activities. When these thresholds are high, rare or weakly connected activities and paths are filtered out, leaving a streamlined model that highlights the main steps and primary pathways in the process. This approach is particularly valuable when analysts need a quick overview of the process, such as during initial assessments or when presenting findings to stakeholders who may only need to see the primary process flow. For example, in a hospital emergency department process, high thresholds might reveal only the core steps like "Patient Registration," "Triage," and "Doctor Consultation," ignoring less frequent activities to provide a clear, high-level view of patient flow.

- **Lower Thresholds**: Conversely, lowering the thresholds allows more activities and paths to be included in the footprint, bringing to light detailed and less frequent behaviours within the process. When thresholds are set lower, rare connections and low-frequency activities become visible, enabling analysts to identify and explore variations and exceptions in the workflow. This level of detail can be crucial when investigating specific issues, such as bottlenecks, inefficiencies, or unique cases that deviate from the standard flow. In a healthcare setting, for instance, lowering the thresholds might reveal rare paths like "Specialist Consultation" following "Triage," which could be relevant for understanding complex cases or identifying areas where additional resources may be required.

Construction of Directly-Follows Graphs

Fuzzy miner output process model in Directly-Follows Graph (DFG) representation. DFG is a fundamental representation used in process mining to visualise the relationships between activities in a process. The DFG uses nodes to represent activities and directed edges to represent the flow between activities that directly follow one another.

The creation of a DFG begins with an event log, which is a collection of traces. Each trace is a series of events that represent how a particular process instance unfolds. To make the graph meaningful and manageable, three filtering thresholds are used:

1. **Trace frequency threshold**: Determines the minimum number of times a trace must occur to be included in the DFG.

2. **Activity frequency threshold**: Sets the minimum number of times an activity must appear to be shown as a node in the graph.

3. **Transition frequency threshold**: Specifies the minimum number of direct successions between two activities needed to include an edge between them.

A Directly-Follows Graph (DFG) begins with filtering traces based on a specified trace frequency threshold. Traces sequences of activities representing individual process instances are retained only if they meet this threshold. This approach ensures that the graph captures both dominant patterns and rare but critical process behaviours, rather than focusing solely on the most common sequences.

The next stage involves filtering activities using an activity frequency threshold. Activities occurring less frequently than the threshold are excluded to maintain visual clarity. However, the relevance and strategic importance of infrequent activities should be taken into account, as some rare actions may hold significant process value. Striking this balance helps preserve the comprehensiveness of the graph without introducing excessive complexity.

Once filtering is complete, the remaining activities are represented as nodes. Directed edges are added between nodes if the number of direct successions between the activities meets the defined transition frequency threshold. This step emphasizes the most meaningful transitions while ensuring that important but less frequent pathways are still acknowledged.

Additional elements are then included to enhance the graph. Frequencies of activities appear on the nodes, while timing information such as the average or maximum time between activities is added to the edges. These details contribute to a richer, more insightful view of the process, capturing both dominant flows and nuanced variations in behaviour and timing.

2.2.3.2.4 Inductive Miner

The Inductive Miner is an algorithm in process mining used for generating process models from event logs. Its unique advantage lies in its ability to create structured, sound models that represent real-world processes accurately, even when the data contains noise or infrequent patterns. This makes it particularly useful in complex environments where other algorithms may struggle to balance precision and simplicity. The algorithm follows an inductive approach, breaking down the process into smaller components that can be analysed individually before combining them to form a comprehensive model. This recursive decomposition allows the Inductive Miner to capture concurrency, loop structures, and other control-flow patterns effectively. It is also scalable, meaning it can manage large datasets without compromising performance.

Compared to other algorithms like the Alpha Miner or Heuristics Miner, the Inductive Miner emphasizes model quality by ensuring the resulting process models are both minimalistic and capable of generalizing from the data. This quality makes it suitable for applications where a clear, understandable process model is essential, such as in business process optimisation and compliance monitoring.

Key rules in Heuristics Miner

The Inductive Miner (IM) framework operates as a recursive algorithm designed to create sound, block-structured models from event logs (Leemans, Fahland, & van der Aalst, 2013). This based on the framework:

1. Base Case Check

The first step in the IM framework is to check if the event log meets a *base case*. A base case represents simple patterns in the data that can be directly transformed into a small process model without needing further decomposition. Common base cases include simple sequences or single activities.

Imagine an event log representing a single step in a patient's journey, such as a blood pressure check. This event is simple and doesn't require any further division or complex modelling. In this case, the IM would recognise this as a base case and directly create a process model node representing the blood pressure check activity.

2. Finding a Cut

If no base case is found, the IM framework attempts to identify a *cut* in the log. A cut is a way to split the log into partitions based on the relationships between activities, like sequence, parallelism, or choice. The cut divides the event log into distinct groups of activities that can be handled separately. Each partition is associated with a specific operator, such as:

- **Sequence**: Activities that follow one another in order.
- **Parallelism**: Activities that can occur concurrently.
- **Choice**: Activities where one of several options is taken.

In a hospital setting, a patient may undergo multiple diagnostic tests, such as an X-ray, MRI, and blood test, which can be done in parallel depending on equipment and staff availability. The IM algorithm would detect a parallel cut, recognizing that these activities do not follow a strict sequence and can occur concurrently. The resulting process model would show these diagnostic tests as parallel tasks.

3. Log Splitting

After identifying a cut, the log is split into smaller *sublogs* according to the cut's partition. Each sublog represents a subset of activities within the process. The IM framework then recursively analyses each sublog to further refine the process model.

Consider a more complex scenario where a patient's journey includes initial assessment, diagnostics, and treatment. After the cut step, the log might be divided into three sublogs:

- **Assessment** (e.g., registration and initial health checks),
- **Diagnostics** (e.g., blood tests and imaging),
- **Treatment** (e.g., medication administration and physical therapy).

The IM framework would then handle each of these sublogs separately, applying the same steps recursively to model each segment of the patient's journey. Moreover. By

breaking down the event log into smaller sublogs, the Inductive Miner can analyse each component (or "module") of the process independently. This modular approach helps in understanding and representing different segments of a process with clarity. For example, in a patient journey, separating the log into sublogs for "Assessment," "Diagnostics," and "Treatment" allows each stage to be analysed on its own, making it easier to create a detailed model.

4. Fall Through

If neither a base case nor a cut can be identified, the IM framework applies a *fall through* mechanism. This step uses heuristics to construct part of the process model, ensuring that the algorithm can continue. Fall through is the default option when no clear structure can be imposed on the data, allowing the framework to proceed even with complex, unstructured event logs.

Suppose a patient journey log includes highly variable or infrequent events, like emergency procedures (e.g., unexpected surgery due to complications). In such cases, the IM algorithm might not identify a base case or a clean cut because these activities don’t fit predictable patterns. Instead, the fall-through mechanism would add these events as individual branches in the process model, maintaining them as distinct steps without trying to group them.

Construction of process model in Process Tree

At the end of its recursive analysis, the Inductive Miner outputs a process tree. A process tree is a hierarchical representation used to model the flow of activities within a process. It provides a structured view of how tasks or activities are related to each other and how they are executed in a process. A process tree uses four primary operators to dictate the relationships and execution order of tasks: exclusive choice (XOR), parallel (AND), sequence (SEQ), and loop (LOOP). These operators help to define how activities in a process should behave and interact with each other (Bernard & Andritsos, 2019).

Exclusive Choice (XOR): The exclusive choice operator (xor) expresses that only one of the branches can be executed. This operator is used to model decision points in the process, where only one of several possible paths will be taken depending on conditions or choices.

Parallel (AND): The parallel operator (and) indicates that all branches should be executed simultaneously, in any order. This operator is used to represent tasks that can happen concurrently or in parallel.

Sequence (SEQ): The sequence operator (seq) enforces an ordered execution of tasks. Activities connected by this operator must be executed from left to right, meaning the first task must be completed before the next one begins.

Loop (LOOP): The loop operator represents more complex execution behaviour, where a set of activities can be repeated multiple times. The loop operator typically has an entry condition (when to start the loop) and an exit condition (when to stop the loop). This operator is crucial for modelling iterative processes where certain activities might need to be revisited.

The root node of a process tree is the top-level node that represents the entire process. It serves as the entry point or starting point of the process model. From this root node, the process can branch into multiple paths, each representing a different sequence or set of activities that may occur depending on the flow of the process. The root node is essential in providing a centralized starting point for understanding the process, from which various paths can diverge, reflecting the possible outcomes or decisions within the process (van der Aalst et al., 2006).

The branches of the process tree define the flow of activities in the process. Each branch represents a possible path that can be taken, either as a sequence of tasks or as a set of alternatives that depend on specific conditions. These branches can be thought of as representing the different execution flows that the process might follow. The branches allow the process tree to capture the complexity of decision points, parallel tasks, and sequences, offering a clear visual representation of how tasks are related and how they flow from one activity to the next (van der Aalst et al., 2006).

The depth and hierarchy of the process tree structure provide flexibility in modelling processes at various levels of detail. The hierarchical nature of the tree allows for different levels of abstraction, meaning that the process tree can be used to represent both high-level workflows, which capture the general steps of the process, and low-level details, which focus on the specific tasks within each step. This structure makes the process tree a versatile tool for understanding both the overall structure of a process and the finer

details that contribute to its execution. The depth and hierarchy help in organizing the process in a way that can be easily navigated and understood by different stakeholders, from those interested in a broad overview to those who need to focus on specific tasks(van der Aalst et al., 2006).

2.2.3.3 Process analysis

After the process model is generated in process mining, process discovery involves analysis to identify patterns and parameters in the process. This analysis can uncover common bottlenecks between particular steps or specific resources being left unused. First, the process model is compared to what would be an ideal process or it may also be considered as a desired state of the total scenario in order to detect variance and opportunities for improvements e.g., by reducing steps that are not necessary or finding more efficient ways to perform certain activities (De Weerdt, 2012).

The process of this discovery is an iterative one because the resulting process model may then be subject to change and further refinement based on feedback from stakeholders or additional data analysis. The final process model is selected based on its ability to fit the observed behaviour in the event logs, which is evaluated using fitness requirements. The selection process involves considering four main criteria: fitness, precision, generalization, and simplicity.

Fitness ensures that the discovered model permits all behaviours observed in the event logs, while precision ensures that the model does not include behaviours that are absent in the logs. Generalization requires that the model be able to replicate behaviours not present in the event logs but likely to occur in the future. Lastly, simplicity criteria require that the discovered model be simplified and optimised.

Taking into account these criteria, a model can be chosen that adequately describes the behaviour that is observed in event logs and at the same time has sufficient generality, precision, and simplicity for practical purposes. This makes the final process model to be a reliable representation of the analysed process.

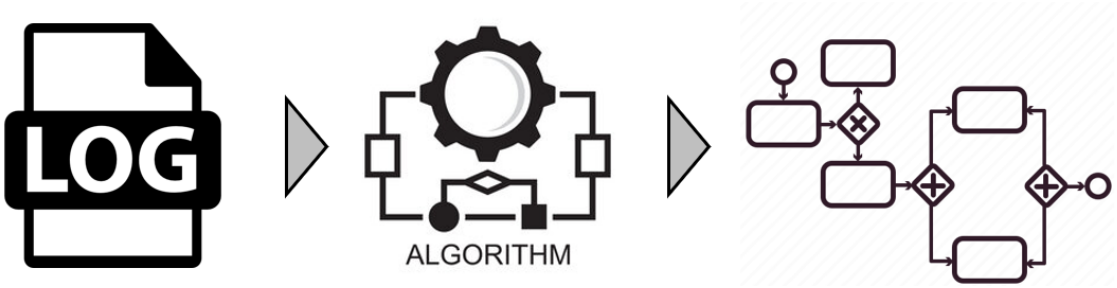


Figure 2.2 Process discovery using event log.

2.2.3.4 Conformance checking

One of the important aspects in process mining is conformance checking, which refers to comparing observed process executions with a predefined process model and checking for differences. The purpose of conformance checking is to determine the similarity between actual process execution and ideal behaviour as specified by a process model. Thus, through the analysis of following the established pattern, we can detect inefficiencies or deviations and other problems associated with processes that may affect their work (A.Rozinat et al., 2010). There are several different approaches to conformance checking in process mining, including:

Control-flow conformance checking: This approach considers the sequence in which activities are performed in the process model compared to how they are observed to be performed. This is achieved by measuring the activity sequences, durations and frequencies in the process model against those observed during execution.

Data conformance checking: Data conformance checking is a data-centric approach that focuses on the data generated and consumed by activities in a process. It involves comparing the data obtained from the process model and the data created during observed execution for consistency and correlation. The main objective is to ensure that the data produced by the observed execution matches the expected data outlined in the process model.

Performance conformance checking: This approach focuses on the process performance measures such as activity lead time, cycle time, and throughput. The aim is to measure those metrics between the process model and actual execution with a view to

identifying possible performance problems. This analysis allows organisations to find any discrepancies and inefficiencies in the process execution that can potentially influence its general performance.

2.2.3.5 Process enhancement

Process enhancement involves the identification and overcoming of challenges associated with the process in order to improve the overall performance of the process itself. Through the use of data-driven analytics, challenges and opportunities for improvement in the process are established. The process of enhancement generally involves multiple essential steps. Initially, it is crucial to detect process-related concerns that hinder the process's performance. This can be achieved through process mining methods like control-flow, data, and performance conformance checks(W. Van Der Aalst, 2016). By examining the data generated by the process, issues like delays or quality problems can be recognised.

After the problems have been identified, the next step is to determine what is causing them. This can involve analysing data and process models for identifiable patterns or trends, conducting interviews or surveys to gather further information, or using any other appropriate method to gain a complete understanding of the problems.

Using root cause analysis, a plan for an improvement can be developed. This plan could include changes to the process model, allocation of resources or the behaviour of people or teams involved in the process. After the plan is developed, it can be implemented by providing education to people about new processes, changing software or systems or modifying physical aspects of the process layout.

After the adoption of these changes, it is important to closely monitor the process in order to determine if such improvements have been effective. This may include the use of process mining approaches to monitor performance on an ongoing basis and spot any potential issues. However, the improvement plan might need to be updated and additional changes may need to be incorporated in it due to any new issues that come up or due modifications applied through the course of the process.

In the healthcare sector, process improvement might include using process mining methods to recognise and resolve problems with patient care delays such as long waits for tests or procedures. Through the analysis of systemic reasons for such delays, a customized improvement plan can be created that may consist of actions targeting staffing deficiencies or optimizing scheduling to increase efficiency.

2.2.4 Outputs from process mining

According to research by Van der Aalst et al. (20120, process mining provides a clearer and more detailed comprehension of processes as opposed to traditional modelling approaches. Process visualisation is one of the useful outcomes of process mining and it creates process maps and flowcharts in visual representation. With these visualisations, it is much easier to understand complicated processes because they show the order and interconnections of activities within a process. Process visualisation can be used by stakeholders, as stated by Rozinat et al. (2009), to gain insights into the process and identify areas for improvement.

Performance metrics are another important product of process mining. Such metrics include activity time, cycle time and throughput that give quantitative measures of process performance. These measures make it possible to compare the variants of process and identify areas for improvement. For instance, if process mining finds activities that are associated with delays, recommendations for changes in the flow can be provided.

In the context of this PhD study, the outputs from process mining can inform the development of a Discrete Event Simulation (DES) model. The output, comprising conceptual modelling and derived parameters such as time-to-event, provides valuable information for creating a more accurate DES model. Conceptual modelling entails creating a high-level representation of the process flow, identifying key activities and events. Derived parameters offer quantitative measures of process performance, such as activity duration, arrival time, probability distribution and routing probability.

With the use of process mining outputs, it will be possible to build the DES model with a higher degree of accuracy, which will reduce any potential biases and inaccuracies. The DES model does not rely on assumptions or estimates but is based on real process data received from event logs.

Furthermore, the outputs from process mining can also serve to validate the accuracy of the DES model. By testing the DES model against results from process mining, its alignment with the actual process behaviour can be assessed. Any disparities or deviations between the model and actual behaviour can be investigated and addressed, leading to a more dependable and accurate model.

2.3 Discrete event simulation methodology

This section provides an overview of Discrete Event Simulation (DES) modelling. It covers key terminologies, input requirements, steps involved, and outputs. This PhD study has a specific focus on DES (i.e. time-to-event) modelling as the primary research area, rather than resource/capacity modelling or other aspects within the wider scope of DES.

In the modelling of DES, there are different types of modelling approaches that can be used each focusing on different areas and serving a specific purpose. One of the types is time-to-event modelling that focuses on timing and sequences of events taking place within a system. This perspective focuses on the temporal facets of processes including duration, waiting time and sequencing. Time-to-event modelling is very useful in the analysis of processes where timing plays a critical role such as scheduling, coordination and process flow optimisation. In the context of this PhD study, the primary research area centres around time-to-event modelling.

Capacity modelling is another form of DES modelling which focuses on resource utilization, capacity allocation and system performance analysis based on the constraints that are represented by those resources(Comas, 2005). In analysing situations where the availability and utilization of resources play a key role in determining the system's efficiency and throughput, capacity modelling becomes very important. Nonetheless, resource modelling is beyond the thesis scope as previously described.

2.3.1 Overview of DES

Discrete event simulation (DES) modelling is a computational methodology utilized to analyse the dynamics of systems that involve discrete events taking place at specific time instances. DES models are computer-based programs that simulate the interactions among components within a system (Law, 2015; Günal & Pidd, 2010).

Fundamentally, the DES modelling procedure is based on decomposing a system into its constituent entities and simulating their dynamic interactions over time. In the DES system model, such a system is represented by entities (for example, people, mechanisms, vehicles) interacting with each other through separate events that modify the state of the system.

DES models include a number of key components, such as entities, events, resources and queues. Entities are system actors (e.g., customers, products, employees), events represent occurrences that affect the state of the system (e.g., arrivals, departures, failures), resources involve facilities or equipment necessary for system activities (e.g., machines, tools, personnel), and queues show waiting lines created when entities need currently occupied resources (Law, 2015).

Development of a DES model requires the definition of a studied system, creation of entities, events, resources and queues and determining rules for each element behaviour. This involves the specification of processing times, probability distributions of occurrence events, and entity routing. Next, the model is run in a sequence of time steps where events take place, entities move through the system and resources are used.

The DES model gives several performance measures that are used to determine the effectiveness of a system such as throughput, wait times and service time. Analysts can use the output of the model to establish areas that need improvement, run scenarios and find out which changes are most effective.

In the health sector, DES modelling has been found useful in interpreting various systems and processes including emergency departments, operating rooms and patient flows. For example, Nikkhah Ghamsari (2017) used DES modelling in a study to evaluate how such factors as patient flow and staffing affect the performance of an emergency department.

The outcome of the model analysis revealed how to improve staffing and processes better so as to reduce patient waiting time and increase resource utilization.

The key terminologies used in Discrete Event Simulation (DES) modelling are presented in Table 2.2 below:

Table 2.2 Key terms used in discrete event simulation (DES) modelling

Term	Definition
Events	Discrete occurrences that affect the state of the system, such as arrivals, departures, failures, or repairs.
Entities	Objects or actors that participate in the system, such as customers, products, or machines
Resources	Facilities or equipment required to carry out the events in the system, such as machines, tools, or labour.
Queues	Waiting lines that form when entities require a resource that is currently occupied.
Time stamp	A discrete unit of time used in the simulation model.
Verification	The process of ensuring that the computerized model (i.e., the computer program) correctly implements the conceptual model as intended. It involves checking that the model is free from programming errors and logically follows the conceptual design. Verification methods include static testing (e.g., correctness proofs, structured walk-throughs) and dynamic testing (e.g., input-output checks, bottom-up or top-down testing, and internal consistency checks) (Sargent, 2010).
Validation	The process of assessing whether the conceptual model and computerized model adequately represent the real-world system for the intended purpose. This involves conceptual model validation (ensuring that the underlying theories and

	assumptions are correct), operational validation (checking if the model's outputs align with real-world data), and data validation (ensuring input data accuracy).

2.3.2 Inputs to DES modelling

In this section, we present an overall description of the inputs for Discrete Event Simulation (DES) modelling, highlighting key elements that support the process of simulation. The input parameters used in the case studies are discussed specifically in Chapters 4 and 5.

Input data plays a pivotal role in DES modelling, as it defines the essential characteristics of the system under study. The accuracy and quality of the input data directly influence the realism, precision, and meaningfulness of the simulation results. As recognised by Law and Kelton (2015), key components of input data include arrival rates, processing times, probabilities, routing rules, and resource constraints, all of which contribute to an accurate representation of real-world systems.

One of the fundamental inputs is the **arrival rate**, which defines how often entities enter the system. In a hospital setting, this could represent patient arrivals to an emergency department or scheduled hospital admissions.

Another critical input parameter is the **processing time,** which indicates how long a specific task or service takes to complete. For instance, in a hospital model, processing times may reflect the duration required for medical diagnosis, surgical procedures, or medication administration.

Probabilities are used to capture the likelihood of different events occurring. In a healthcare scenario, this could include the probability of a patient requiring surgery, developing complications, or being readmitted.

Routing rules define how entities move within the system, guiding transitions between different stages of care. For example, in a hospital, routing rules determine how patients are transferred between departments (e.g., from triage to consultation or from surgery to intensive care).

A critical but often overlooked aspect of DES inputs is **resource constraints**, which play a significant role in determining system performance. Resource availability including doctors, nurses, hospital beds, and medical equipment directly affects waiting times and patient flow. Unlike theoretical models where resources may be assumed to be unlimited, real-world hospital operations are constrained by staff availability, bed capacity, and equipment scheduling. Ignoring resource limitations can lead to overly optimistic simulation results that do not reflect actual queuing dynamics.

In this context, resource constraints must be considered alongside arrival rates and processing times to produce realistic hospital DES models. For example, reducing patient waiting times is not solely dependent on optimizing patient flow but also on ensuring that sufficient staff and medical facilities are available at each stage. By incorporating resource availability data (where possible) or making reasonable assumptions about resource constraints, DES models can provide more actionable insights for healthcare decision-making. Through simulation testing of different patient flow scenarios, analysts can assess the consequences of various hospital processes while taking resource constraints into account, ensuring that model-driven recommendations are both feasible and effective in practice.

2.3.3 Steps involved in developing DES modelling.

In this section below we present a brief description of the steps involved in the Discrete Event Simulation (DES) modelling. The following outlines the detailed steps in DES modelling:

Define the problem and objectives: Clearly define the problem and simulation objectives, identifying the processes, entities, inputs, and outputs of the system. For instance, in a hospital model, processes may include patient arrival, triage, diagnosis, treatment, and discharge. Objectives might focus on optimizing patient flow and reducing wait times.

Develop a conceptual model: Create a conceptual model that shows the relationships between processes, entities and resources. This model provides an overview of the system and its components' interrelations.

Collect data: Gather information about the system to be simulated, including arrival rates, processing times, resource availability and other relevant variables. Information can be obtained from different sources such as historical records, questionnaires and expert inputs.

Develop a simulation model: Build a simulation model using the conceptual model and input data as the foundation. Use software applications that have been designed for DES modelling. The simulation model should include the processes, entities and resources discovered in addition to the rules for their behaviour.

Validate the model: Validate the simulation model by comparing the output of simulated data to real-world data or other existing models. Validation ensures that the simulation model is able to make correct predictions and therefore supports quality decision-making.

Run experiments: Conduct experiments using the simulation model by specifying experimental conditions, including arrival rates and processing times. Run the simulation model for a specified period of time and capture the output.

Analyse the results: Evaluate the results of the simulation model in terms of performance measures like throughput, waiting times. Present the results in the form of graphs and charts to identify trends and patterns.

Make recommendations: Use findings from the simulation model to make suggestions on how to improve the system that is being simulated. Recommendations could include the changes in either processes, entities or experimental conditions for performance optimisation of the system.

These steps are typically iterative and can be repeated as needed until the desired objectives are achieved.

2.3.4 Outputs of DES

In this section, we present a general description and explanation of the outputs from Discrete Event Simulation (DES) modelling that can offer information regarding the performance of the modelled system. These mostly consist of different performance measures which are used to assess and improve the system.

Within the healthcare sector, such outputs can comprise critical measurements i.e patient average waiting time, length of stay as well as the average number of patients within the system at a particular point in time. Other outputs in DES include indicators such as throughput, utilization, wait time, cycle time, lead time and resource cost. Throughput is the rate with which entities, such as patients, are processed through the system while utilization is used to represent how well resources like hospital beds or operating rooms are being utilized. Resource cost evaluates the expenses associated with operating system resources, including salaries of healthcare professionals, as well as medical supplies and equipment costs. These results help in promoting evidence-based decision making and also encourage quality improvement in healthcare processes.

2.4 Systematic literature review: Integrating process mining in DES modelling.

This section presents a literature review of previous studies on process mining in DES modelling. A systematic search of relevant databases was conducted using specific keywords and followed the PRISMA protocol for this review. Data extraction was primarily focused on the integration of process mining in DES modelling, specifically in the areas of conceptual modelling, parameterisation, and validation. The results of this comprehensive literature review are presented and discussed in this chapter. Additionally, the gaps identified for further research in this field are discussed in the final section of this chapter.

2.4.1 Aim of the literature review

This literature review aims to provide an analysis of research evidence relating to the applicability of combining process mining methods in DES established by following these objectives:

- Review a range of published literature in the integration of process mining in DES.
- Report on key process mining elements that have been integrated with DES modelling in terms of conceptualization, parameterisation and validation.
- Describe existing applications, algorithms and software where process mining has been integrated with DES.
- Identify gaps in literature.
- Provide an analysis and discussion.

2.4.2 Methods

The steps taken in this literature review follow a review framework from Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA). This includes the following phases:

- **Searching:** the systematic identification of potential studies using key words search at full text level and all fields.
- **Screening:** the titles, abstracts and keywords are evaluated using pre-defined inclusion and exclusion criteria derived from the review question.
- **Data extraction:** Careful review of the studies that meet the pre-defined inclusion and exclusion criteria to extract evidence to support the review.
- **Synthesis:** the establishment of a framework for data analysis and the discovery of key themes.
- **Reporting:** documentation of the review findings.

2.4.3 Literature Searches

Using the key literature on process mining and DES, a search strategy was developed including the following key search terms and related text words: “process* mining”, “workflow* mining”, “mining workflow*”, “process* discovery”, “event-log*”, “event log*”, “log-file”, “log file”, “conformance checking”, “discrete-event*”, “discrete event*”, “simulat*”.

The searches for articles were conducted in June 2022 in the following databases: Web of Science, PubMed, Embase, JSTOR, IEEE Xplore, ACM Digital Library, MATHSCINET, and the Google Scholar search engine. Searches were conducted, where possible, at full text/all fields levels to ensure they were thorough. These searches were supplemented with manual searches using the reference list checking and citation searching. The search strings are supplied in Appendix 1.

2.4.4 Study Selection

At this stage, the articles from the databases and Google Scholar were first unified in Mendeley. Duplicated articles were removed to avoid reviewing the same articles twice. Abstract, title and keywords of each article were analysed using the inclusion/exclusion criteria. If the abstract did not provide sufficient details to determine whether an article was relevant, then the full text was reviewed.

Studies published in English that report the use of process mining techniques to support DES modelling were included. Studies that described the use of DES in process mining (e.g. DES models to generate dummy data for use in process mining), or studies that report the general purpose of software development combining process mining and DES, were excluded from the review but briefly reported in the discussion.

Following are the inclusion criteria used to add an article to the analysis:

Table 2.3 Inclusion criteria

No	Inclusion criteria	Justification
1	All identified articles that include process mining techniques to support DES modelling	• All articles that used process mining techniques in DES modelling were included. This helps answer the literature review question (i.e., how process mining has been used to support DES modelling) and excludes irrelevant studies outside the scope. • Articles from Google Scholar were also accepted in this review because relevant published articles and dissertations may emerge from this source.

2	Only publications in English were included.	• To ensure that the finalised articles are evaluated effectively, only English-language articles understandable to the reviewer were selected.

Following are the exclusion criteria used to add an article to the analysis:

Table 2.4 Exclusion criteria

No	Exclusion criteria	Justification
1	Articles that include the usage of DES in process mining (e. g DES models to generate dummy data for use in process mining) were excluded.	• The focus of this literature review is to examine integration of process mining in DES modelling. Thus, selecting publication with the usage of DES in process mining will not contribute to the literature review aim.
2	Studies that report general purpose of software development combining process mining and DES were excluded.	• The research focuses on integrating process mining to support DES modelling in healthcare processes. Therefore, selecting publications that only report software development combining the two is not directly aligned with the research aim.
3	Non-English articles were excluded.	• Non-English publications were excluded since only English-language papers can be rigorously evaluated by the reviewer.

2.4.5 Data Synthesis

A synthesis of the final included list of papers was then performed to describe how process mining has been used in DES modelling. Data on the applications and software used for process mining and DES was also extracted. A data extraction spreadsheet comprising multiple categories to aid in the extraction of relevant information was developed in Microsoft Excel.

2.5 Results

The procedure employed to reduce and examine the records is illustrated in the PRISMA flow diagram in Figure 2.3. The electronic searches identified 6,451 papers (combining articles from databases and Google Scholar). After considering duplicates, 60 papers were removed. The first level of screening excluded all irrelevant articles by scanning the title and abstracts. The second level of screening, i.e. evaluation of the full text, is carried out if the abstract is inconclusive. The remaining 34 articles were retained and supplemented with manual searches, using reference list checking, citation searching and hand searching techniques, which identified 10 further relevant papers, bringing the total included studies to 44.

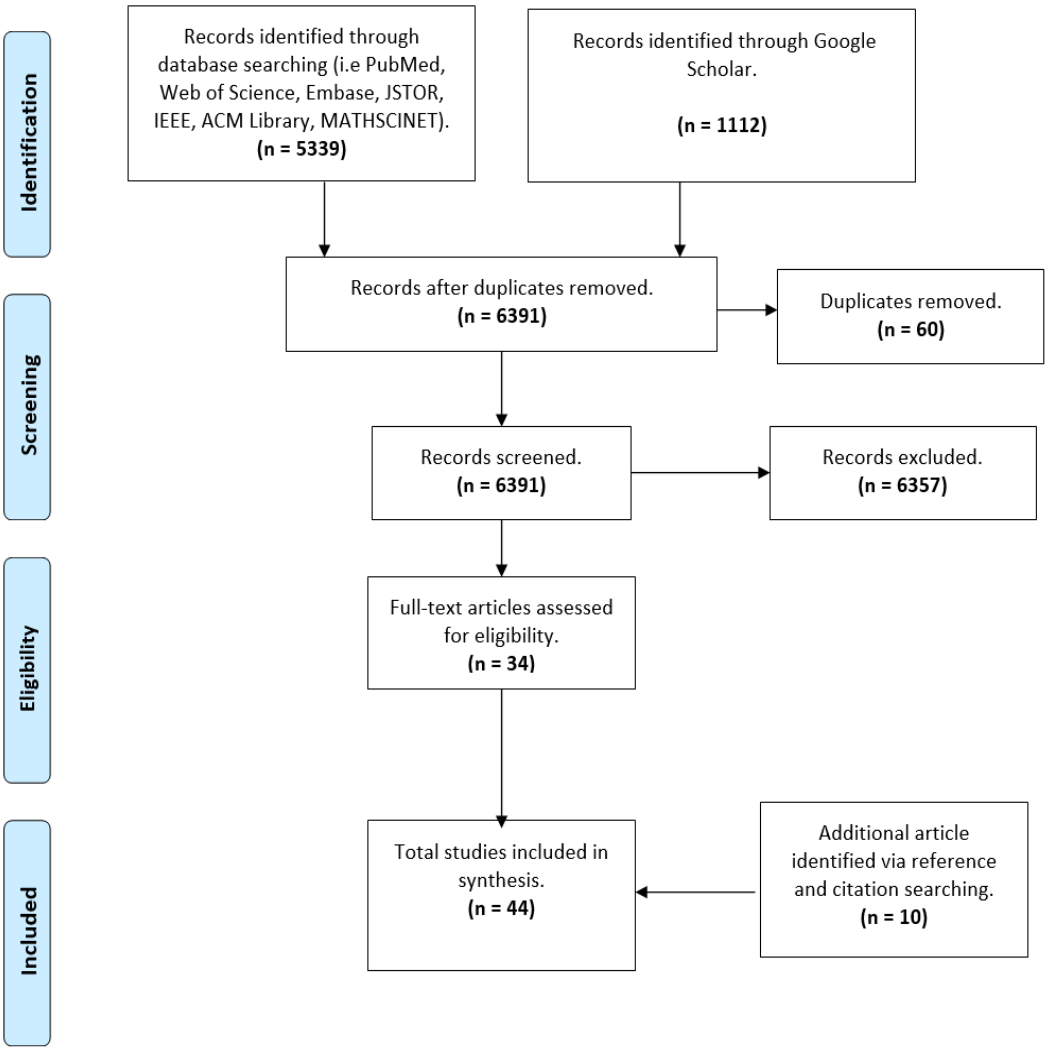


Figure 2.3 PRISMA diagram

2.5.1 Overview of Retrieved Articles

Table 2.5 provides a general overview of the 44 articles included in this review. There are three major areas in which process mining has been applied to support discrete event simulation. The most common is healthcare, followed by business and management, and manufacturing. In addition, a few papers discussed the subject more broadly without applying it to any field. The number of publications appears to be increasing over time. For instance, the volume of work conducted in this field between 2016 and 2022 was double that of the period between 2009 and 2015, suggesting that research in this area is of increasing interest.

Table 2.5 Areas integrating process mining in DES modelling.

Area	Author(s)	Year	Title
Healthcare	Abohamad et al.	2017	A hybrid process-mining approach for simulation modelling
	Antonio et al.	2022	A hybrid model to support decision-making in the stroke clinical pathway
	Antunes et al.	2019	A solution framework based on process mining, optimisation, and discrete-event simulation to improve queue performance in an emergency department
	Aloui et al.	2019	A generic framework to analyse and improve patient pathways within a healthcare network using process mining and discrete-event simulation
	Cho et al.	2019	An evidence-based decision support framework for clinician medical scheduling
	Fontanili et al.	2022	A discrete event simulation-based methodology for building a digital twin of patient pathways in the hospital for near real-time monitoring and predictive simulation
	Gunn	2017	Resource-based simulation model from process event log
	Halawa et al.	2021	Integrated framework of process mining and simulation-optimisation for pod structured clinical layout design
	Johnson et al.	2019	The clearpath method for care pathway process mining and simulation
	Kovalchuk et al.	2018	Simulation of patient flow in multiple healthcare units using process and data mining techniques for model identification
	Mans et al.	2013	A process-oriented methodology for evaluating the impact of IT: A proposal and an application in healthcare
	Mesabbah et al.	2019	A hybrid process mining framework for automated simulation modelling for healthcare
	Pegoraro et al.	2022	Framework for emergency department management: A proposal based on process mining and simulation
	Phan et al.	2019	Clinical pathway analysis using process mining and discrete-event simulation: an application to incisional hernia

	Tamburis	2019	Bridging the gap between process mining and DES modelling in the healthcare domain
	Tamburis et al.	2020	Process mining as support to simulation modelling: A hospital-based case study
	Van Hulzen et al.	2022	Supporting capacity management decisions in healthcare using data-driven process simulation
	Van Hulzen et al.	2021	The need for interactive data-driven process simulation in healthcare: A case study
	Zhou et al.	2014	Process mining based modelling and analysis of workflows in clinical care – A case study in a Chicago outpatient clinic
Business and management	Jadric	2019	Framework for discrete-event simulation modelling supported by LMS data and process mining
	Jadric et al.	2020	Process mining contributions to discrete event simulation modelling
	Khodyrev et al.	2014	Discrete modelling and simulation of business processes using event logs
	Lamine et al.	2015	Improving the management of an emergency call service by combining process mining and discrete event simulation approaches
	Maruster et al.	2009	Redesigning business processes: A methodology based on simulation and process mining techniques.
	Mesabbah et al.	2018	Presenting a hybrid processing mining framework for automated simulation model generation
	Nakatumba et al.	2012	Generating event logs with workload-dependent speeds from simulation models
	Petitdemange et al.	2019	A tool-based framework to assess and challenge the responsiveness of emergency call centres
	Rashid et al.	2022	Integrating process mining with discrete-event simulation for dynamic productivity estimation in heavy civil construction operations
	Rozinat et al.	2009	Discovering simulation models
	Tangkawarow et al.	2018	Evaluating the performance of tax determination using discrete event simulation

	Utama et al.	2020	Mining shift work operation from event logs
	Werner et al.	2011	Discrete event model structure identification using process mining
Manufacturing	Lugaresi et al.	2021	Automated manufacturing system discovery and digital twin generation
	Friederich et al.	2022	Process mining for dynamic modelling of smart manufacturing systems: Data requirements
	Friederich et al.	2022	A framework for data-driven digital twins for smart manufacturing
	Lugaresi et al.	2019	Manufacturing systems mining: Generation of real-time discrete event simulation models
	Lugaresi et al.	2021	Discovery and digital model generation for manufacturing systems with assembly operations
	Mayr et al.	2022	Abstracting process mining event logs from process-state data to monitor control-flow of industrial manufacturing processes
	Novak et al.	2022	Digitalised automation engineering of industry 4.0 production systems and their tight cooperation with digital twins
	Overbeck et al.	2021	Continuous adaption through real data analysis turn simulation models into digital twins
General papers	Pourbafrani et al.	2022	SIMPT: Process improvement using interactive simulation of time-aware process trees
	Pourbafrani et al.	2022	Interactive process improvement using simulation of enriched process trees
	Van Der Aalst	2021	Concurrency and objects matter! disentangling the fabric of real operational processes to create digital twins
	Van Der Aalst	2018	Process mining and simulation: A match made in heaven!

Almost all the studies used three main software tools for process mining: ProM, Disco and ProDiscovery. In contrast, there were more than ten different software tools used for DES modelling (including AnyLogic, Simul8, Witness, CPN Tools,ProModel, Automod,Autostat, Flexsim, SimPy and Arena) as outlined in the Table 2.6. Furthermore, there were two studies that developed bespoke software tools to automate the conversion of process mining outputs into DES models (Mesabbah & McKeever, 2019; Mesabbah et al., 2019).

Author(s)	**Process mining software**			**Discrete-event simulation software**									
	ProM	**Disco**	**ProDiscovery**	**AnyLogic**	**Simul8**	**Witness**	**CPN Tools**	**ProModel**	**Automod**	**Autostat**	**Flexsim**	**SimPy**	**Arena**
(Aloui, Thomas Franck Paolo Bercelli Saber, 2019)	✓	✓		✓									
(Cho et al., 2019)			✓						✓	✓			
(Gunn, 2017)	✓												
(Jadric, 2019)		✓					✓						
(Jadrić et al., 2020)		✓											
(Lamine, E., Fontanili, F., Di Mascolo, M., & Pingaud, 2015)		✓				✓							

(Mans et al., 2013)	✓						✓						
(Nakatumba et al., 2012)	✓						✓						
(Petitdemange et al., 2020)						✓							
(Phan et al., 2019)		✓		✓									
(A. Rozinat et al., 2009)	✓						✓						
(Tamburis, 2019)	✓	✓			✓								
(Tamburis & Esposito, 2020a)	✓	✓			✓								
(Utama et al., 2020)							✓						
(Werner-Stark et al., 2011)	✓						✓						
(Zhou et al., 2014)	✓	✓						✓					
(Fontanili et al., 2022)	✓										✓		
(Antonio et al., 2022)												✓	
(Pegoraro et al., 2022)		✓						✓					

(Halawa et al., 2021)		✓									✓		
(Overbeck et al., 2021)												✓	
(Rashid & Louis, 2022)	✓												
(Novák & Vysko˘cil, 2022)	✓												
(Pourbafrani et al., 2021)												✓	
(Lugaresi et al., 2019)													✓
(Friederich, Francis, et al., 2022)												✓	
(Johnson et al., 2019a)	✓												
(Antunes et al., 2019)		✓											✓

Table 2.6 Software used in past studies

This review found that several process discovery algorithms and process mining techniques were used to support the conceptualisation of DES modelling.

Conceptualisation: Fuzzy miner, Alpha-Miner, Alpha++, Heuristic Miner, Genetic algorithm, Disjunctive Workflow Schema (DWS) mining, Multi-phase miner, Inductive Miner- infrequent, and Frequency Mining.

Table 2.7 Algorithms used in past studies.

Process mining algorithm	**Reference**	**Number of Studies**
Alpha-Miner	Rashid & Louis, 2022; A. Rozinat et al., 2009; Werner-Stark et al., 2011; Zhou et al., 2014; Overbeck et al., 2021; Friederich, Francis, et al., 2022; Tangkawarow et al., 2018	7
Fuzzy miner	Abohamad Waleed, Ramy Ahmed, & Amr, 2017; Lamine, Fontanili, Di Mascolo, & Pingaud, 2015; Phan et al., 2019; Zhou et al., 2014; Pegoraro et al., 2022; Lugaresi et al., 2019	6
Heuristic Miner	A. Rozinat et al., 2009; Werner-Stark et al., 2011; Rashid & Louis, 2022; Lugaresi et al., 2019; Khodyrev & Popova, 2014	5
Genetic algorithm	Werner-Stark et al., 2011; Lugaresi et al., 2019; Khodyrev & Popova, 2014	3
Inductive Miner	Tamburis, 2019; Tamburis & Esposito, 2020a; Pourbafrani & van der Aalst, 2022	3
Customised algorithm	Halawa et al., 2021; Lugaresi & Matta, 2021a, 2021b	3
Disjunctive Workflow Schema (DWS) mining	Werner-Stark et al., 2011	1
Multi-phase miner	Werner-Stark et al., 2011	1
Frequency Mining	Cho et al., 2019	1

2.5.2 Steps in process mining

There are a number of steps involved in process mining as outlined in Figure 2.4. Data collection is the first step, and this can be from sources such as information systems (Cho et al., 2019; Kovalchuk et al., 2018; Jadrić et al., 2020), interviews (Abohamad Waleed, Ramy Ahmed, & Amr, 2017; Petitdemange et al., 2020; van Hulzen et al., 2021), observations (Abohamad Waleed, Ramy Ahmed, & Amr, 2017; van Hulzen et al., 2021), and documentation (Abohamad Waleed, Ramy Ahmed, & Amr, 2017).Data cleaning consists of removing outliers, finding missing data, imputing missing data, and correcting the data format (Cho et al., 2019; Petitdemange et al., 2020). Following this, the data extraction step involves the retrieval and storage of relevant data in a structure called an event log for use in process mining.

The event log is the input for process mining software (Marin-Castro & Tello-Leal, 2021; Jans et al., 2019). This storage structure fulfils the minimum requirements for storing data about a process, in that it needs the ID, the activity, and the timestamp of an executed activity. To meet this criterion, it is necessary to select appropriate variables from the cluster of raw data. The study by Johnson et al. (2019a) recommends that the selection of variables should be reviewed by process experts to ensure that the meaning of the variable is correct and that it is acceptable and relevant for process modelling.

Process modelling is performed automatically by algorithms derived from process mining, such as Alpha Miner (W. van der Aalst et al., 2004), Heuristic Miner (Saravanan & Rama Sree, 2011), Fuzzy Miner (Günther & van der Aalst, 2007), and Genetic Miner (De Medeiros et al., 2007), as outlined in Table 2.7. Most studies used Fuzzy Miner, which is suitable for mining complex processes, also known as spaghetti processes, and which exhibit a large amount of unstructured behaviour. Fuzzy Miner employs a variety of techniques, such as removing non-essential edges and grouping highly compatible nodes into one node (Abohamad et al., 2017; Lamine et al., 2015; Phan et al., 2019; Zhou et al., 2014; Pegoraro et al., 2022; Lugaresi et al., 2019), to produce simplified process models. Heuristic algorithms, by contrast, can be used to determine the main behaviour when not all details and exceptions are registered in the event log. They achieve this by considering the sequence of events in the case (Saravanan & Rama Sree, 2011).

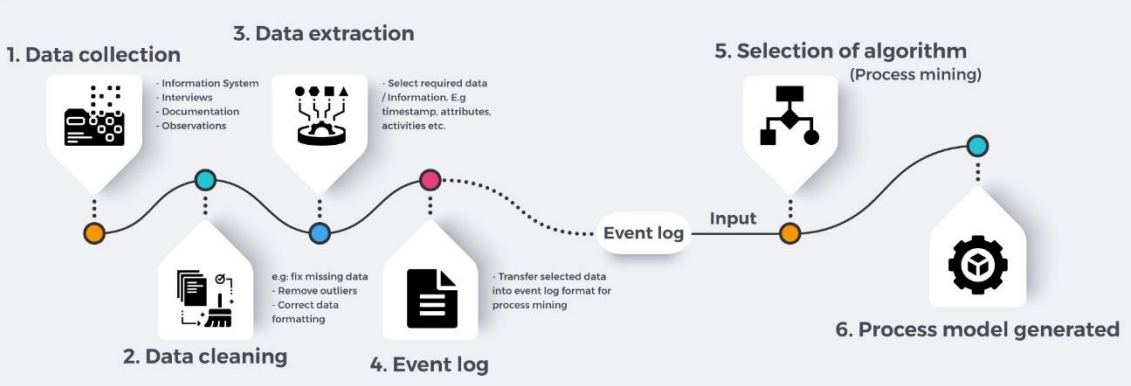


Figure 2.4 Steps involved in process discovery (adapted from Cho et al., 2019; Kovalchuk et al., 2018; Jadrić et al., 2020; Abohamad et al., 2017; Petitdemange et al., 2020; van Hulzen et al., 2021).

Expert opinion is a valuable source for validating the resulting process model. Studies have sought expert opinions to validate the process model and ensure that the identified process trajectories align with the actual process flow. Expert opinions help in verifying the correctness and coherence of the process model. This validation step aims to identify any anomalies or deviations in the process modelling. We have engaged domain experts to review and validate the process models obtained through process mining (van Hulzen et al., 2022; Antonio et al., 2022; Pegoraro et al., 2022; Overbeck et al., 2021; Mayr et al., 2022; Novák & Vyskočil, 2022; Rashid & Louis, 2022; Khodyrev & Popova, 2014; Friederich, Francis, et al., 2022; Johnson et al., 2019a). The expert input provides valuable insights into the accuracy and fidelity of the process model.

2.6 How process mining has been used in DES modelling

Based on reviewing the identified studies, there were three main ways in which process mining has been used to support DES: the development of the conceptual model, the generation of input parameters for simulation modelling, and the validation of the resulting simulation model. The following section describes in more detail each of these aspects:

2.6.1 Conceptual modelling

Process mining enables a deeper understanding of real-world processes by uncovering important characteristics and dynamics. The sequence of events, process variations, and resource utilization patterns identified through process mining can be directly incorporated into the design of the DES model structure, ensuring reliable representation of the real system being simulated.

Process mining was used to extract information from event logs to identify the underlying processes that occur in the system (Kovalchuk et al., 2018; Tamburis, 2019; Utama et al., 2020; Petitdemange et al., 2020; Jadrić et al., 2020; Zhou et al., 2014; Mesabbah et al., 2019). This information can be used to refine and validate the conceptual model. For example, in a study by Kovalchuk et al. (2018), the authors used process mining to identify the key processes that influenced the performance of a healthcare unit. They then used this information to refine the conceptual model, resulting in a more accurate representation of the system's behaviour.

One notable study demonstrating the integration of process mining into DES model design is the work of Mans et al. (2013). In their paper, they highlight how process mining techniques, such as discovering process models and performance metrics from event logs, can be utilized to inform the design of DES models, capturing the essential characteristics and dynamics of the underlying processes. The authors utilized process mining to develop the conceptual modelling phase of their process-oriented methodology. They collected factual execution data related to the business processes and pre-processed the data to ensure its quality. By applying process discovery algorithms, they automatically extracted process models that represented the actual execution flow, including activity sequences and decision points. These process models were then validated and refined using domain knowledge and expert input. Process mining provided quantitative insights into the processes, allowing for the calculation of key performance indicators. The validated process models served as the basis for creating simulation models, which were used to analyse the impact of different digital technologies on the processes. Through simulation experiments, the authors compared and evaluated various scenarios and variants, enabling them to assess the benefits and drawbacks of IT interventions. The

conceptual modelling process involving process mining was iterative, refining the models based on insights gained from the simulation experiments.

The sequence of events identified through process mining provides valuable insights into the chronological flow of activities within the process. For example, Rozinat et al. (2009) and Jadrić et al. (2020) discuss the alignment of process mining and DES by integrating process discovery techniques with DES models. They emphasize how the discovered process models can be used to guide the construction of the DES model structure, ensuring that the simulated activities accurately reflect the observed sequence of events.

Process variations play a crucial role in capturing the inherent complexity and diversity of real-world processes. Process mining techniques, such as process discovery algorithms, can identify different process variants and their frequency of occurrence. This information can be incorporated into the design of DES models to represent the different paths and decision points within the process. Van Hulzen et al. (2022) discuss the integration of process mining and DES, emphasizing the ability to capture and simulate process variations in healthcare environments. By incorporating the identified process variants into the DES model, the simulation can reflect the real-world dynamics more accurately.

By analysing actual data, process mining provides a framework for building process models more efficiently, as noted in several studies (Mesabbah et al., 2019; Gunn, 2017; Halawa et al., 2021; Johnson et al., 2019a; Kovalchuk et al., 2018). In the adaptation of the process model to the DES simulation, only the events and paths of interest are selected. This is because simulating complex processes is challenging, and a focused process model is necessary. Process experts can provide guidance in determining which section of the process is most critical for improvement, enabling the development of a focused process model. In the field of healthcare, selecting pathways and events to focus on in process mining analyses heavily relies on expert opinion, particularly that of clinicians. This is because each pathway has its own clinical interpretation, and only a clinical expert can assess the significance of the pathways and events and select those that are of interest (Antonio et al., 2022).

2.6.2 Parameterisation

Process mining can be used to identify appropriate parameter values by analysing event logs (Antonio et al., 2022). There are two types of parameters that can be obtained from process mining. The first type includes directly measurable parameters, such as frequency and activity duration (van Hulzen et al., 2022; Overbeck et al., 2021). These parameters can be directly extracted from the process data without any further computation (Pourbafrani & van der Aalst, 2022). A study by van Hulzen et al. (2022) mentioned that the frequency parameter reveals how often certain healthcare procedures occur, while activity duration represents the total time taken for an activity from start to completion. The authors emphasize that the second type of parameters, derived parameters, obtained from event log data, play a vital role in capacity management decisions. This knowledge is essential for understanding the demand for healthcare procedures and the time required to perform them.

The second type of parameters needs to be derived from the dataset (i.e., event log), as they cannot be sourced directly from process mining results. These parameters require additional analysis and calculations based on the event log data to derive meaningful insights about the process (Overbeck et al., 2021). For example, in a study published by Tamburis and Esposito (2020a), the routing-out probabilities between activities were determined by calculating absolute frequencies observed throughout the process. Additionally, the study explored the possibility of obtaining percentage probabilities for routing-out distribution. Another study by Tamburis and Esposito (2020b) calculated the routing probability based on the BPMN structure.

Many studies have focused on analysing various time-related perspectives, such as service time, as evidenced by research papers (Antonio et al., 2022; Antunes et al., 2019; Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Cho et al., 2019; Fontanili et al., 2022; Johnson et al., 2019a; Mans et al., 2013; Mesabbah et al., 2019; van Hulzen et al., 2022; Khodyrev & Popova, 2014; Mesabbah & McKeever, 2019; Nakatumba et al., 2012; Petitdemange et al., 2020; Rozinat et al., 2009; Friederich, Lugaresi, et al., 2022; Friederich, Francis, et al., 2022; Lugaresi et al., 2019; Lugaresi & Matta, 2021b; Rashid & Louis, 2022; Mayr et al., 2022; Novák & Vyskočil, 2022; Overbeck et al., 2021; Pourbafrani et al., 2021; Pourbafrani & van der Aalst, 2022).

Additionally, several studies have examined resource-related factors, including resource workload, as discussed in (Aloui et al., 2019; Antunes et al., 2019; Mans et al., 2013; Mesabbah et al., 2019; van Hulzen et al., 2022; Nakatumba et al., 2012; Jadrić et al., 2020; Rashid & Louis, 2022; Pourbafrani et al., 2021; Lugaresi et al., 2019). Resource analysis plays a critical role in evaluating the relationships between individuals, teams, and departments throughout the entire organisation, providing insights into their performance. By considering these inputs, we can enhance communication flow and gain a clearer understanding of how work is handled and shared across departments.

One study, conducted by Mesabbah et al. (2019), proposed an ASMB-H model to address resource planning challenges. The ASMB-H (Auto Simulation Model Builder for Healthcare) is an advanced hybrid simulation framework that extends the original ASMB model by integrating process mining, machine learning, and discrete event simulation (DES) to automate the generation of healthcare simulation models. This model improves decision-making by leveraging real-time patient data and predictive analytics to enhance resource allocation, patient flow modelling, and system performance forecasting. The resource analysis module within ASMB-H enhances traditional discrete event simulation (DES) by dynamically modelling staff availability across different shifts. This approach provides a more comprehensive representation of resource utilization and enables better planning and allocation.

By incorporating actual process data into the model, the accuracy and reliability of simulation results are improved, leading to more informed decision-making and a better understanding of the system's behaviour. These findings highlight the value of process mining as a tool for parameterisation in studies involving DES modelling.

2.6.3 Validation

Process mining plays a crucial role in validating various aspects of DES modelling, such as conceptual modelling and input parameters. Several past studies have utilized process mining to validate the process model generated and compare it with existing conceptual models or expert opinions. To validate conceptual modelling, researchers have compared the process model derived from process mining with the existing conceptual model. This comparison ensures that the process model accurately represents the real-world process. For instance, various studies have examined the process model obtained through process mining by comparing it with existing process models developed using formalisms like BPMN (van der Aalst, 2018; Antunes et al., 2019; Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Halawa et al., 2021; Johnson et al., 2019a; Tamburis, 2019; van Hulzen et al., 2022; Maruşter & van Beest, 2009; Pourbafrani & van der Aalst, 2022; Overbeck et al., 2021). This alignment of process models enables researchers to assess the accuracy and consistency of the process model derived through process mining.

Antunes et al. (2019) conducted a study where they employed a simulation model to examine two critical factors: the volume of patients arriving at the Emergency Department (ED) and the time needed for their assessment, using the original doctor schedule as a basis. To assess the simulation's accuracy, the researchers compared the simulation outcomes with actual data derived from the hospital's information system. The simulation revealed an average daily arrival of 160.7 patients at the ED, exhibiting a marginal deviation of only 2.7% in comparison to the real data. Similarly, the assessment time exhibited a similar pattern, with an average of 14.4 minutes and a slight difference of 2% when compared to the actual data. These findings indicate that the simulation model is highly accurate and reliable. Process mining was instrumental in confirming this accuracy by comparing the simulation results to real-world data.

Van Hulzen et al. (2022) used process mining to check the accuracy of their simulation results. They combined statistical methods and input from experts to validate the simulation. They compared specific performance measures, such as throughput and waiting times, using separate datasets for statistical validation. They also relied on expert input to validate aspects of the model that required domain knowledge, using methods such as structured walk-throughs.

During validation, they discussed the assumptions and simplifications made during model development with experts to determine whether they were valid. If the simulation did not meet the desired level of accuracy, the model was refined accordingly. Since healthcare processes are complex and require specialised knowledge, the authors collaborated with simulation experts to build the model and analyse the results. However, the experts did not always have detailed knowledge of the specific healthcare processes. The interaction between domain experts and simulation modellers occurred at different stages, which sometimes prolonged the project timeline.

In addition to validating the process model, the resulting parameters derived from process mining can also be validated to ensure their correctness for generating accurate simulation results. Authors have performed validation by comparing the decisions or outputs generated by DES simulation using the parameters derived from process mining with the decisions obtained through process mining analysis (Antunes et al., 2019; Overbeck et al., 2021; Pourbafrani & van der Aalst, 2022). This comparison helps confirm whether the input parameters used in the DES simulation accurately reflect the behaviour of the real process. For example, in a study focused on stroke patients, the authors calculated the distribution of length of stay using process mining and then validated this distribution by comparing it with the parameters generated through simulation modelling (Aloui et al., 2019).

By validating both the process model and the input parameters, authors can ensure the reliability and accuracy of the DES modelling results. These validation steps enhance confidence in the simulation outputs and provide a solid foundation for decision-making and process improvement initiatives. Process mining supports the validation of conceptual modelling and input parameters for DES modelling. Through comparisons with existing conceptual models, expert opinions, and validation of parameters against process mining results, authors can verify the accuracy and consistency of the process model and parameters.

Table 2.8 Key Elements of Integrating Process Mining into DES Modelling

Integrating PM in DES	Elements of integration	References
Conceptualisation	Automatic process model discovery using event log	Mans et al., 2013; Zhou et al., 2014; Kovalchuk et al., 2018; Phan et al., 2019; Tamburis, 2019; Mesabbah & McKeever, 2019; Cho et al., 2019; Rozinat et al., 2009; Werner-Stark et al., 2011; Lamine, Fontanili, Di Mascolo, & Pingaud, 2015; Mesabbah et al., 2019; Petitdemange et al., 2020; Jadrić et al., 2020; van der Aalst, 2018; Măruşter & van Beest, 2009; Abohamad Waleed, Ramy Ahmed, & Amr, 2017; van Hulzen et al., 2022; Pourbafrani & van der Aalst, 2022; Antonio et al., 2022; Fontanili et al., 2022; Pegoraro et al., 2022; Halawa et al., 2021; Overbeck et al., 2021; Friederich, Lugaresi, et al., 2022; Lugaresi & Matta, 2021b; Lugaresi & Matta, 2021a; Rashid & Louis, 2022; Novák & Vyskočil, 2022; van der Aalst, 2021; Mayr et al., 2022; Lugaresi et al., 2019; Khodyrev & Popova, 2014; Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Friederich, Francis, et al., 2022; Johnson et al., 2019a; Antunes et al., 2019
	Visualisation of trace variance	Mans et al., 2013; Rozinat et al., 2009; Jadrić et al., 2020; Tangkawarow et al., 2018; van Hulzen et al., 2022; Pourbafrani & van der Aalst, 2022; Antonio et al., 2022; Halawa et al., 2021; Overbeck et al., 2021; Friederich, Lugaresi, et al., 2022; Lugaresi & Matta, 2021b; Lugaresi & Matta, 2021a; Rashid & Louis, 2022; Mayr et al., 2022; Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Johnson et al., 2019a; Antunes et al., 2019; Fontanili et al., 2022; Pegoraro et al., 2022
	Clustering	Kovalchuk et al., 2018; Utama et al., 2020; Mayr et al., 2022; Rozinat et al., 2009; van Hulzen et al., 2022; Antonio et al., 2022
Parameterisation	**Time/performance measures:**	
	Service time	Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Antonio et al., 2022; Antunes et al., 2019; Cho et al., 2019; Fontanili et al., 2022; Friederich, Francis, et al., 2022; Friederich, Lugaresi, et al., 2022; Johnson et al., 2019a; Khodyrev & Popova, 2014; Lugaresi & Matta, 2021b;

		Lugaresi et al., 2019; Mans et al., 2013; Mayr et al., 2022; Mesabbah & McKeever, 2019; Mesabbah et al., 2019; Nakatumba et al., 2012; Novák & Vyskočil, 2022; Overbeck et al., 2021; Petitdemange et al., 2020; Pourbafrani & van der Aalst, 2022; Pourbafrani et al., 2021; Rashid & Louis, 2022; Rozinat et al., 2009; van Hulzen et al., 2022
	Arrival time	Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Antonio et al., 2022; Antunes et al., 2019; Cho et al., 2019; Fontanili et al., 2022; Halawa et al., 2021; Lugaresi & Matta, 2021b; Overbeck et al., 2021; Pegoraro et al., 2022; Pourbafrani et al., 2021; van Hulzen et al., 2022
	Waiting time	Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Antunes et al., 2019; Cho et al., 2019; Fontanili et al., 2022; Kovalchuk et al., 2018; Lamine, Fontanili, Di Mascolo, & Pingaud, 2015; Lugaresi & Matta, 2021a; Lugaresi & Matta, 2021b; Mans et al., 2013; Măruşter & van Beest, 2009; Mesabbah & McKeever, 2019; Mesabbah et al., 2019; Pegoraro et al., 2022; Petitdemange et al., 2020; Pourbafrani et al., 2021; Rozinat et al., 2009; Tangkawarow et al., 2018; van Hulzen et al., 2021; van Hulzen et al., 2022; Zhou et al., 2014
	Total time in process	Abohamad Waleed, Ramy Ahmed, & Amr, 2017; Fontanili et al., 2022; Kovalchuk et al., 2018; Lugaresi & Matta, 2021b; Mans et al., 2013; Pegoraro et al., 2022
	Transaction time between activities	Fontanili et al., 2022; Mesabbah & McKeever, 2019; Tamburis, 2019
	Probability measures:	
	Routing probabilities between the activities	Antonio et al., 2022; Jadrić et al., 2020; Johnson et al., 2019a; Lugaresi et al., 2019; Mesabbah et al., 2019; Petitdemange et al., 2020; Pourbafrani & van der Aalst, 2022; Pourbafrani et al., 2021; Tamburis, 2019
	Probabilities of taking alternative path	Halawa et al., 2021; Khodyrev & Popova, 2014; Rozinat et al., 2009
	Resources:	
	Resource workload	Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Antunes et al., 2019; Jadrić et al., 2020; Lugaresi et al., 2019; Mans et al., 2013; Mesabbah

		et al., 2019; Nakatumba et al., 2012; Pourbafrani et al., 2021; Rashid & Louis, 2022; van Hulzen et al., 2022
	Resource utilisation	Antunes et al., 2019; Friederich, Lugaresi, et al., 2022; Lugaresi et al., 2019; Petitdemange et al., 2020; Pourbafrani & van der Aalst, 2022; Rozinat et al., 2009; van Hulzen et al., 2022
	Resource performance (i.e., number of executions per activity)	Friederich, Lugaresi, et al., 2022; Pourbafrani & van der Aalst, 2022; Tamburis, 2019; van Hulzen et al., 2022
	Dependencies among resources	Pourbafrani et al., 2021; Rozinat et al., 2009; Zhou et al., 2014
Validation	Comparison of resulting process models with historical data	Aloui, Thomas, Franck, Paolo, Bercelli, & Saber, 2019; Antunes et al., 2019; Halawa et al., 2021; Johnson et al., 2019a; Măruşter & van Beest, 2009; Overbeck et al., 2021; Pourbafrani & van der Aalst, 2022; Tamburis, 2019; van der Aalst, 2018; van Hulzen et al., 2022
	Expert domain	Antonio et al., 2022; Friederich, Francis, et al., 2022; Johnson et al., 2019a; Khodyrev & Popova, 2014; Mayr et al., 2022; Novák & Vyskočil, 2022; Overbeck et al., 2021; Pegoraro et al., 2022; Rashid & Louis, 2022; van Hulzen et al., 2022

2.7 Discussion

Our study provides a comprehensive literature review on the application of process mining in supporting Discrete Event Simulation (DES). To the best of our knowledge, this is the first study to review how process mining has been utilized for this purpose. We examined 44 studies across a diverse range of fields, including healthcare, business, and manufacturing. While the earliest example we found dates back to 2009 (Rozinat et al., 2009), the majority of studies were conducted in the past five years, indicating a growing interest in this area among authors. Through our review, we identified three primary ways in which process mining has been used to support DES modelling: conceptualisation, parameterisation, and validation. The majority of studies utilized process mining to support conceptual modelling in DES, and a significant number of studies employed process mining to facilitate parameterisation of the DES model. However, we observed that only a few studies used process mining to support the validation of the DES model.

Process mining provides a data-driven approach to modelling that can significantly enhance the accuracy and efficiency of conceptual modelling. The ability to capture and analyse real-world event logs allows authors to create more precise models that capture the complex interactions and dynamics of the system being modelled. However, this review also highlights the importance of data preparation in process mining, which is often neglected in current literature. Data preparation is a crucial aspect of process mining, and expert opinions can be leveraged, particularly for datasets of low quality. Selecting relevant variables for event logs and addressing missing data can greatly benefit from process experts' advice (Zhou et al., 2014). Furthermore, expert confirmation is essential at all stages of process mining and DES, particularly for complex problems. In addition, data quality can also impact the accuracy of the conceptual model, as discussed in the next paragraphs. Several studies emphasize the significance of good-quality data in process modelling, as poor data quality can lead to poor representation of the actual process. Frameworks for data quality management have been developed, and facilitating accurate data registration can enhance data quality (Jadrić et al., 2020).

Our review highlights the critical role of data quality in process mining. Poor data quality can have significant impacts on the resulting conceptual model, leading to inaccurate or incomplete representations of the actual process. Several studies have emphasized the importance of good-quality data in process modelling, underscoring the need for accurate, complete, and relevant data (Jadrić et al., 2020; van der Aalst, 2018). For instance, consider a scenario where data is missing or incomplete—this can lead to inaccuracies in the process model. Similarly, irrelevant data can result in a cluttered or confusing process model, making it challenging to interpret. Therefore, it is essential to ensure high data quality before starting the process mining analysis. Additionally, a study by van der Aalst (2018) emphasized the need for accurate and complete event logs, as missing or incorrect data can result in biased or incorrect process models. Another study by Jadrić et al. (2020) showed that poor data quality can result in overfitting of the model, which may lead to poor generalisation and ineffective process improvement. Johnson et al. (2019b) proposed a framework for data quality management, which includes data profiling, data cleaning, and data enrichment. The authors suggest that these steps should be performed iteratively to ensure the highest quality of data for process mining. Another study (Jadrić, 2019) reported that data quality can be improved by facilitating accurate data registration, which helps to identify and address errors and inconsistencies in the data.

To create a process model using process mining, there are numerous algorithms available, but there is currently no consensus on how to choose the best one. Selecting the right algorithm is crucial for creating an accurate process model that captures important process behaviours. It is important to consider the data conditions and objectives of the process mining analysis to ensure the resulting process model is effective. For example, the Fuzzy Miner algorithm is used to address complex datasets, but it may not be the best option for all scenarios. In a study by Halawa et al. (2021) conducted in a heart clinic, three existing algorithms were explored to solve the layout and design problem. The study found that genetic algorithms were effective in improving patient flow and achieving layout automation. Therefore, choosing the appropriate algorithm for a specific problem is essential for creating a process model that reflects the current state of the process. The lack of consensus in choosing the best algorithm underscores the need for further research on this topic. Experts suggest that selecting the most appropriate algorithm is often a matter of trial and error and may require expert input.

Additionally, it is important to consider the scalability of the selected algorithm and ensure that it can handle large and complex datasets.

The quality assessment of process models is a critical aspect of process mining, yet it has received relatively little attention in the literature. There are several principal metrics used to assess the quality of the process model, such as fitness, precision, generalization, and simplicity. However, these metrics are not consistently employed to perform model assessments. There is a lack of standardization in how these metrics are applied and interpreted, which can make it difficult to compare results across different studies and to draw meaningful conclusions about the effectiveness of different process mining approaches. This can lead to inconsistencies and inaccuracies in the process models, which can ultimately undermine the value of process mining as a tool for improving organisational processes.

Fitness measures the degree of agreement between the process model and the event log used to create it, while precision measures the model's ability to correctly predict the behaviour of new process instances that were not used in its construction. Generalization measures the model's ability to capture the essential characteristics of the process instances and generalize them to new instances, and simplicity measures the complexity of the model and how well it can be understood and communicated to others. A consistent use of these metrics can help identify the strengths and weaknesses of a process model and inform the selection of appropriate process model discovery algorithms for future projects. A process model with high fitness, precision, generalization, and simplicity can accurately represent the real-world process, make reliable predictions, and be easily communicated to stakeholders.

Research has shown that utilizing the precision metric is particularly useful for capturing complex processes, and its consistent use can benefit the selection of process model discovery algorithms in the future. For instance, a study by Friederich, Francis, et al. (2022) employed the precision metric to evaluate the accuracy of the process model generated from an event log of an industrial case. The findings indicated that the precision metric provided a better understanding of the model's accuracy compared to other metrics.

In addition to precision metrics, it has been shown that process models with high levels of simplicity are more easily communicated to stakeholders. For example, a study by Buijs et al. (2012) evaluated the simplicity of process models created using different discovery algorithms. The study found that simpler models were easier to understand and communicate to stakeholders and were less likely to be misinterpreted or misunderstood. In another study, van der Aalst (2018) assessed the simplicity of process models using a measure called "control-flow complexity," which evaluates the complexity of the model's control-flow structure. The study found that simpler process models were preferred by end-users, who found them easier to understand and use in their work.

In process mining, event logs can be analysed to extract valuable parameters about a process. Some of these parameters, such as activity duration and arrival time, can be directly calculated from the event log. However, certain parameters cannot be directly obtained from the event log and require additional analysis or expert knowledge. These parameters include process cycle time, activity capacity, activity frequency, and process flow. To derive these parameters, it is necessary to examine the sequence and timing of activities, as well as the resources required for each activity. Moreover, the estimation of certain parameters, such as the cost of resources or the value of customer satisfaction, may also require expert knowledge.

Automating the parameter updating process can be a useful strategy for handling changes in data and ensuring the accuracy of the DES model. In particular, automation can help to streamline the process of updating parameters in response to changes in the data. For example, some studies have proposed using automated techniques to support real-time updates to simulation data, allowing the input parameters for the DES model to be continually updated (Rashid & Louis, 2022; Friederich, Lugaresi, et al., 2022).However, it is important to note that automating this process may involve additional costs, and it may not always be necessary or practical to incorporate real-time updates to the simulation model. Therefore, the decision to use automation should be based on a careful assessment of the costs and benefits of this approach in each specific case.

The strength of this review lies in its comprehensive coverage of various topics related to the application of process mining to support discrete event simulation. The review

describes the entire process, starting from the initial data preparation stage to the validation of the model. The review provides a detailed explanation of the process mining techniques used for conceptual modelling, parameterisation and validation and their strengths and limitations. One limitation of this review is that it did not involve a second reviewer in the process of article selection. Instead, the inclusion of relevant articles was ensured through manual searches such as reference searches and citation searches.

2.8 Addressing gap in the current state of art in integrating process mining in DES modelling.

The current literature on the integration of process mining in DES modelling suggests there are still gaps in the current knowledge that need to be addressed. One key gap is the lack of a comprehensive, step-by-step framework for this integration (Mesabbah et al., 2019; Kovalchuk et al., 2018). The need for a step-by-step framework arises from the fact that process mining and DES have different methodologies, data requirements, and analysis techniques. Integrating them requires clear guidance on how to combine their respective processes and leverage their strengths to derive meaningful inputs and accurate simulation models. Specifically, the gap lies in the lack of a comprehensive framework that encompasses the various stages of the integration process. Additionally, many existing studies on the integration of process mining and DES have a narrow scope, focusing on specific industries or processes. A comprehensive approach that considers various contexts will provide a better understanding of the potential benefits and limitations of integrating process mining and DES.

A step-by-step framework would outline the best practices, methods, and tools to be employed at each stage, ensuring a systematic and rigorous integration process. These stages include data collection, process discovery, parameterisation, simulation model development, and validation. The literature also highlighted limitations in the existing research related to data quality (Mesabbah & McKeever, 2019; Maruşter & van Beest, 2009), which can impact the accuracy and reliability of the integration. Therefore, we plan to address this in our step-by-step framework and support it with case studies.

The PhD study aims to understand issues around practical implementation by conducting case studies using clinical datasets. Specifically, the study will focus on two distinct datasets: the Mexican breast patient pathways dataset and the peritoneal dialysis dataset

for kidney patients. By examining these clinical datasets, the study will provide insights into the integration of process mining and DES within the context of clinical pathways and healthcare processes.

This approach will offer valuable knowledge and understanding of how process mining can be effectively integrated with DES in clinical settings. It will contribute to the broader applicability of process mining frameworks by exploring their effectiveness and potential benefits in healthcare domains (Gunn, 2017; Mans et al., 2013). The findings from these case studies will enhance the practical relevance and real-world impact of the integration, paving the way for future research and implementation in diverse industries and processes.

Moreover, this PhD study aims to address the existing gap in the literature by developing a comprehensive best-practice checklist for integrating process mining techniques into DES modelling. The integration of process mining and DES has been recognised as a promising approach, but the lack of systematic guidance has hindered its widespread adoption and limited its comparability and replicability across studies. The best-practice checklist serves as a solution to this gap by providing a systematic and standardized approach for integrating process mining into DES modelling. Drawing on inputs from the literature, the checklist encompasses a wide range of industries and contexts, ensuring its broad applicability. It offers clear guidance on key stages of the integration process, including data collection, pre-processing, process discovery, parameterisation, simulation development, validation, and result interpretation.

CHAPTER 3
INITIAL STEP-BY-STEP FRAMEWORK

3.1 Chapter Overview

This chapter presents a detailed, step-by-step framework for integrating process mining into DES modelling developed through a comprehensive review of existing literature. An improved framework is proposed (Chapter 7) to address limitations identified in the current literature, achieved by introducing additional steps to overcome these limitations.

3.2 A step-by-step framework from literature review

The literature on frameworks for the integration of process mining in DES modelling is relatively scarce, but several studies have proposed frameworks and methods for combining these techniques. For example, Rozinat et al. (2009) proposed a framework for integrating process mining and DES, which includes the following steps: (1) process discovery and data preparation, (2) process model transformation, (3) simulation model construction, (4) simulation execution, and (5) performance analysis. Similarly, Tamburis et al (2020) proposed a hybrid approach that combines process mining, DES techniques for healthcare process analysis.

3.2.1 Existing Framework from Rozinat and Tamburis

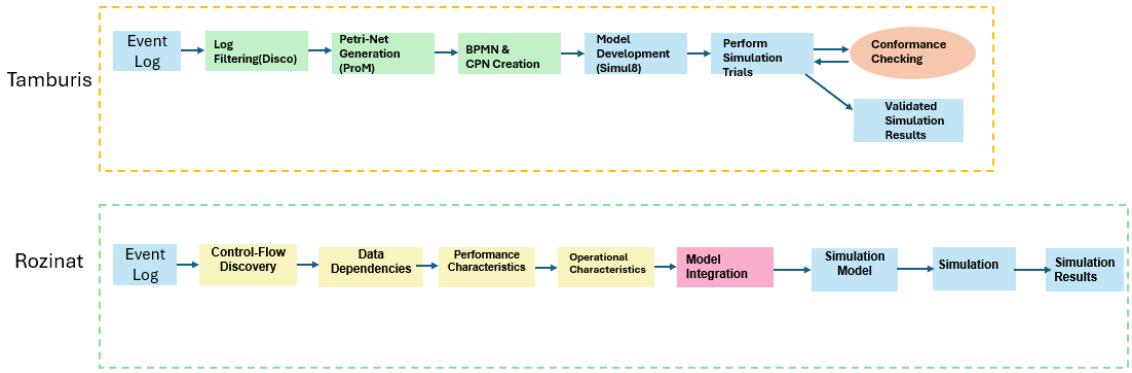


Figure 3.1: Existing frameworks by Rozinat et al. (2009) and Tamburis et al. (2020)

3.2.1.1 Rozinat's Framework

In this section, we explain the five main components of the Rozinat framework: (1) Event Log Analysis, (2) Control-Flow Discovery, (3) Data Dependency Analysis, (4) Performance Analysis, and (5) Model Integration.

Control-flow Discovery

Rozinat's framework employs process mining techniques to automatically extract a process model from event logs. Specifically, the alpha-algorithm was used to generate Petri net models that represent the sequence and dependencies of activities in the process. This Petri net visualises the flow of activities and serves as the foundational structure for further analysis. The aim was to model the causal relationships among activities, which is crucial for simulating how a process operates under different scenarios. In the study, a simplified version of a real-life outpatient clinic process was used as an example. The process involves patients visiting the clinic for various examinations. The event log captured activities such as First Visit, Lab Test, X-ray, ECG, CT Scan, MRI, and Third Visit.

The control-flow discovery algorithm identified that after a patient's First Visit, they might undergo a Lab Test and an X-ray either sequentially or in parallel. Based on this analysis, a Petri net model was created, showing the causal relationships between these activities. For example, the model indicates that ECG may or may not be required depending on specific conditions, and after completing these examinations, the patient would proceed to a Second Visit.

Data Dependency (Decision Point Analysis)

Decision point analysis seeks to understand how data attributes influence the routing of cases at decision points within a process. This is achieved by converting decision points into classification problems and using decision tree algorithms to discover decision rules. In the outpatient clinic process, Rozinat identified decision points such as whether an ECG is needed after the X-ray. The decision tree was constructed using attributes like the patient's age, diagnosis, and anesthetic risk score (ASA). For instance, the analysis revealed that ECG was only required for patients over 60 years old or with an ASA score

greater than 2. This rule was then incorporated into the process model to ensure that the simulation accurately reflected the conditions under which different paths were chosen.

Performance Analysis

Rozinat conducted a performance analysis to enhance the process model by incorporating critical metrics from the event log, such as execution times, waiting times, and probabilities for different paths. This information enabled a more accurate simulation of the process's performance. For instance, the execution time for the CT Scan had a mean of 45.1 minutes with a standard deviation of 1.7 minutes, while the waiting time for the CT Scan averaged 4320.4 minutes, with a standard deviation of 486.1 minutes. Additionally, the probability of requiring an ECG was 72%, compared to a 28% probability of it not being required. These statistics were used to parameterise the simulation model, ensuring it accurately reflected the real-world timing and flow of activities within the outpatient clinic process.

Model Integration

Model integration combines the control-flow, data dependency, and performance perspectives into a single comprehensive simulation model. Coloured Petri Nets (CPNs) were chosen for this purpose due to their expressiveness and strong simulation capabilities. In the integrated model, each activity is represented with details from all perspectives. For example, the CT Scan activity in the CPN model is triggered only if the patient’s diagnosis is corpus carcinoma or ovarium carcinoma (data dependency perspective), takes approximately 45.1 minutes to execute with variability modelled by a normal distribution (performance perspective), and is performed by a radiology department resource, which becomes unavailable for other tasks during this period (organisational perspective). The integrated model was validated by running simulations and comparing the generated event logs with the original logs. The results closely matched the original data.

3.2.1.2 Tamburis Framework

In this section, we outline the main components of Tamburis et al.'s framework, including (1) Event Log Extraction, (2) Log Filtering, (3) Petri Net Generation, (4) BPMN 2.0 Conversion, (5) Simulation Model Creation, (6) Performing Simulation Trials, (7) Event Log Generation for Each Simulation Run, and (8) Conformance Checking.

Event Log Extraction

In Tamburis' work, event log extraction involves obtaining raw data from the Hospital Information System (HIS) used in the healthcare process under study. The logs capture detailed patient care activities, such as admissions, treatments, examinations, and discharges. This data serves as the foundation for constructing a reliable process model, enabling accurate simulation and analysis of the hospital's patient care processes.

Log Filtering

In Tamburis' framework, log filtering is a step where the raw event logs extracted from the Hospital Information System (HIS) are cleaned and refined to ensure that only relevant and high-quality data is used for subsequent process analysis and modelling. This involves removing noise, incomplete cases, and any events that are not directly related to the core healthcare process. The goal is to focus the analysis on the most important activities that directly impact patient care, eliminating extraneous data that could skew the results or add unnecessary complexity to the model.

Tamburis employed the DISCO software to carry out the log filtering process. The software was used to sift through the raw event logs and remove activities that were not essential to the patient care process, such as administrative tasks like patient registration or billing. Instead, the focus was placed on clinical activities, such as medical examinations, treatments, and procedures, which are directly related to the patient's healthcare journey. By filtering out non-essential activities, Tamburis ensured that the resulting event logs were cleaner and more accurate, providing a solid foundation for building a reliable process model and conducting meaningful simulations.

In Tamburis' work, the noise that was removed during the log filtering process primarily consisted of activities and events that were not directly related to the core clinical processes being studied. This included:

1. **Infrequent Activities**: Activities that occurred very rarely and did not significantly impact the overall patient care process were considered noise. These infrequent activities could complicate the model without providing meaningful insights into the typical workflow.

2. **Administrative Tasks**: Non-clinical activities such as patient registration, billing, or scheduling that do not directly affect the patient's medical treatment were filtered out. These activities, while necessary for hospital operations, were not relevant to the clinical process being modelled and could introduce unnecessary complexity into the process model.

3. **Incomplete Cases**: Cases where essential data was missing, or incomplete were removed. For instance, if an event log did not capture the entire sequence of a patient's journey (e.g., missing records of certain treatments or diagnostic procedures), it was considered noise and filtered out to maintain the accuracy and reliability of the process analysis.

Petri Net Generation

In the Tamburis framework, Petri Net generation is a step where the filtered event logs are transformed into a graphical model that represents the sequence and dependencies of activities within the healthcare process. Petri Nets are used to visualise and analyse the workflow, allowing for the identification of key activities, decision points, and the overall structure of the process. This model serves as the foundation for further process analysis and simulation.

Tamburis used the Inductive Miner Infrequent (IM) algorithm in ProM6 to generate the Petri Net. This algorithm is particularly effective for handling complex and noisy event logs, which are common in healthcare settings. It ensures that the Petri Net accurately reflects the main behaviours of the process while appropriately managing infrequent or exceptional activities. The Petri Net generated by ProM6 included places (states), transitions (activities), and arcs (flows between activities). This model captured essential

aspects of the process, such as parallel activities, decision points, and the sequence of care steps, providing a comprehensive view of the healthcare workflow. Tamburis refined the initial Petri Net by performing fitness checks to verify its alignment with the observed behaviour in the event logs. Adjustments were made as necessary to improve the model's precision and relevance.

BPMN 2.0 Conversion

In Tamburis' framework, after generating the Petri Net from the event logs, the next step is to convert this Petri Net into a Business Process Model and Notation (BPMN) 2.0 format. BPMN 2.0 is a widely recognised standard for modelling business processes, offering a more intuitive and accessible graphical representation than Petri Nets. This conversion is essential for making the process model understandable to a broader audience, including non-technical stakeholders, and for facilitating further analysis and simulation.

Tamburis employed a specialized plug-in in ProM6 to automate the conversion of Petri Nets into BPMN 2.0 diagrams, translating structural elements such as places, transitions, and arcs into corresponding BPMN components like tasks, gateways, and sequence flows. The mapping process ensured that places were represented as events or conditions, transitions as activities, and arcs as sequence flows. Efforts were made to maintain the logical flow and integrity of the original Petri Net, ensuring that parallel processes, decision points, and sequences were accurately captured in the BPMN model without losing essential information.

Simulation Model Creation

In Tamburis' framework, the simulation model is created based on the BPMN 2.0 diagram to replicate the healthcare process in a virtual environment, enabling experimentation, analysis, and optimisation. Using Simul8, a commercial Discrete Event Simulation (DES) tool, the BPMN elements were mapped into Simul8 to define activities, resources, and timings, allowing for the simulation of patient flow and resource utilization. Key aspects of the simulation included:

Resource and Timing Definition: Activities were assigned resources like medical staff, equipment, and rooms, with timings derived from real-world data to reflect variability in healthcare processes.

Flow and Routing Logic: Patient flows, probabilities at decision points, and parallel activities were modelled to simulate real-world scenarios accurately.

Testing and Refinement: Iterative testing ensured the simulation outcomes matched expected real-world results, with adjustments made to address issues such as bottlenecks or resource constraints.

Simulation Trials: Various scenarios were tested, such as altering patient arrival rates, resource availability, or introducing new equipment, to assess their impact on efficiency.

Analysis and Recommendations: Simulation results were analysed to identify KPIs, such as patient wait times and resource utilization, leading to recommendations for improving the healthcare process.

The simulation model, applied to an ophthalmology ward, replicated the patient care process from consultation to surgery and follow-up, allowing Tamburis to explore how changes in variables, such as operating room availability or patient arrival patterns, influenced process efficiency.

Performing Simulation Trials

In Tamburis' framework, simulation trials validate the accuracy of the simulation model and explore different scenarios for process optimisation. Using Simul8, Tamburis tested variations in parameters such as patient arrival rates, resource levels, and process flow changes. Multiple trials were run for each scenario to ensure statistically reliable results and account for variability.

Key performance indicators (KPIs), such as patient wait times, resource utilization rates, and process bottlenecks, were analysed to gain insights into process performance. Comparing results across scenarios helped identify optimal configurations for resource

allocation and process steps. The simulation model was iteratively refined based on trial outcomes to address inefficiencies or bottlenecks and improve accuracy.

In Tamburis' study, trials modelled patient care processes in an ophthalmology ward, exploring scenarios like increased patient arrivals or reduced operating room availability. Insights from these trials informed recommendations for process improvements, such as optimizing staff schedules or increasing resource capacity, to enhance efficiency and patient care.

Event Log Generation for Each Simulation Run

In Tamburis' framework, event logs generated after each simulation trial capture the sequence of activities, timestamps, and resource usage during the simulated process. These logs, structured similarly to the real-world event logs from the Hospital Information System (HIS), are essential for comparing the simulated process with the actual one, enabling conformance checking and further refinement of the simulation model.

Simul8 was configured to automatically generate event logs for each trial, capturing detailed information about patient flow and resource use. The logs were formatted for compatibility with process mining tools, such as the .xes format used by ProM, to facilitate analysis. Customization using Simul8's Visual Logic ensured that all relevant details were accurately recorded, such as activity durations and resource involvement.

Event logs were exported for analysis in ProM, allowing comparisons between simulated and real-world processes. Multiple logs, generated across different simulation trials, supported scenario analysis and model validation. Through conformance checking, discrepancies between the simulated and actual processes were identified and addressed, refining the model to improve its accuracy.

In Tamburis' study, event logs for scenarios in an ophthalmology ward captured activities like Pre-operative consultation, Surgery, and Post-operative care. These logs were exported and analysed to validate the simulation model against the real-world workflow, ensuring the model closely replicated the actual healthcare process.

Conformance Checking

Conformance checking in Tamburis' framework used to compare the simulated event logs generated during the simulation trials with the actual event logs extracted from the Hospital Information System (HIS). The goal of conformance checking is to ensure that the simulation model accurately reflects the real-world process. This step involves identifying any discrepancies between the simulated and real processes, which helps in refining the simulation model to improve its accuracy and reliability.

Tamburis used the ProM tool to compare actual event logs from the Hospital Information System (HIS) with simulated event logs generated during simulation runs. Alignment techniques in ProM matched activities in the logs based on activity names, timestamps, and case identifiers, enabling a detailed comparison. For example, if a real-world log showed a surgery at 10:00 AM and the simulation log at 10:15 AM, the alignment would assess if such delays were consistent or indicated discrepancies.

Conformance checking techniques, such as fitness checking and precision checking, were employed to evaluate how well the simulated process aligned with the real-world process. Fitness checking measured whether the sequence of events in the simulation matched the actual process, while precision checking ensured that the simulation adhered closely to observed behaviours without unnecessary deviations. Deviations, such as shorter waiting times in simulations compared to actual logs, were analysed to identify areas needing refinement, such as timing inaccuracies or resource constraints.

Tamburis iteratively refined the simulation model based on these findings, adjusting parameters or modifying process flows. For instance, if earlier discharge times were observed in the simulation, adjustments accounted for real-world delays like waiting for lab results. The refined model was re-simulated and rechecked until it aligned closely with the actual logs

3.2.2 Comparison of Rozinat's and Tamburis' frameworks

Section 3.2.2 compared existing frameworks by Rozinat and Tamburis, structured around a set of common aspects: data handling, process mining, simulation model translation, model integration, validation, and integration of perspectives. These aspects provided the organising categories for the proposed framework, where each aspect was operationalised as a corresponding phase or step (Figure 3.2).

Although both Rozinat and Tamburis originally developed their approaches in the context of specific case studies, they abstracted their work into structured methodological steps that extend beyond the original datasets. This abstraction is what makes them frameworks rather than simple case descriptions. For instance, Rozinat outlined phases such as event log preparation, process discovery, conformance checking, and enhancement, while Tamburis proposed a structured workflow for integrating process mining into healthcare simulation modelling.

We acknowledge, however, that the generalisability of these frameworks is more limited compared to fully generic frameworks. Their value lies in providing repeatable methodological components that can be adapted to other datasets, and this limitation motivated the synthesis in this thesis, where recurring elements from both were combined with insights from the case studies to form an enhanced framework tailored for integrating process mining into DES modelling.

Framework Phase (Figure 3.2)	Mapped Aspect (Section 3.2.2)	Rozinat's Approach	Tamburis' Approach
Phase 1: Data Preparation	Data Handling (event logs and filtering)	Emphasizes high-quality input data, extracted and standardized (e.g., XES, MXML). Robust filtering based on timestamps, event types, and resource involvement. Addresses data quality issues like missing data and inconsistencies.	Focuses on domain-specific filtering aligned with tools like BPMN 2.0 or Petri Nets. Refines data based on context (e.g., patient flow in healthcare). Incorporates domain knowledge and business rules for advanced filtering.
Phase 2: Process Mining(Conceptual Modelling and Parameterisation)	Process Mining and Control-Flow Discovery	Uses algorithms like the α-algorithm for discovering control-flow, including parallelism and loops. Incorporates decision mining to analyse data attributes influencing routing decisions. Provides deep structural and behavioural insights.	Relies on practical techniques like Inductive Miner Infrequent (IMi) for noisy data. Outputs models in BPMN for direct use in simulations. Prioritizes simplicity and robustness but lacks decision mining and multi-perspective analysis.

Phase 3: Transforming to DES Model	Simulation Model Translation	Uses Coloured Petri Nets (CPNs) for detailed, flexible models integrating control-flow, performance, and organisational aspects. Requires significant expertise for effective use.	Translates BPMN diagrams into executable models for tools like Simul8. Prioritizes user-friendliness and accessibility but lacks the depth and flexibility of CPNs.
Phase 4: Validation	Validation and Iterations	Incorporates iterative "second pass" validation, comparing simulated and original event logs for refinement. Highly adaptable to process changes and dynamic environments.	Focuses on conformance checking using metrics like fitness and precision. Validation is simpler but lacks iterative refinement, limiting adaptability to evolving processes.

3.2.3 Discussion from Tamburis and Rozinat Framework analysis

The Rozinat and Tamburis frameworks provide distinct methodologies for integrating process mining with discrete-event simulation (DES) modelling. Both frameworks leverage event logs to derive process insights but are designed to address different challenges. The Rozinat framework, introduced in 2009, focuses on semi-automatically generating simulation models by integrating various process perspectives, such as control-flow, performance, and resource allocation, using Coloured Petri Nets (CPNs). The primary aim of this framework is to enhance the accuracy of simulation models while minimizing manual effort through automation and iterative improvements. On the other hand, the Tamburis framework, developed in 2020, takes a more application-specific approach, focusing on healthcare environments. It provides a practical methodology for building DES models from event logs, emphasizing data preparation, conformance checking, and usability with commercial tools such as Simul8.

While both frameworks attempt to bridge the gap between process mining and simulation, their design philosophies diverge significantly. Rozinat prioritizes a theoretically rigorous and automated approach that integrates multiple process perspectives, whereas Tamburis emphasizes practical application, addressing real-world challenges like noisy data and domain-specific requirements in healthcare.

Data Preparation

Data preparation is a crucial component of any process mining and simulation framework, as the quality of input data significantly impacts the accuracy of the resulting models. The Rozinat framework assumes that event logs are complete and clean, which limits its applicability in real-world settings where data is often noisy or incomplete. The framework focuses on extracting dependencies and control-flow relationships from event logs but lacks explicit mechanisms for handling data quality issues. This assumption makes it less effective in domains like healthcare, where event logs often require substantial pre-processing.

In contrast, the Tamburis framework explicitly addresses the challenges of data preparation. It incorporates a structured approach to filtering and cleaning event logs, using tools like Disco to remove infrequent behaviours and ProM to handle noise. This focus on data preparation ensures that the extracted control-flow models are both robust

and accurate, making the framework particularly suitable for complex environments like hospitals, where event logs may be fragmented or inconsistent. By emphasizing data preparation, Tamburis ensures that subsequent steps in the framework are built on a solid foundation, a feature that is often overlooked in other frameworks.

Process Mining and Process Model Discovery

Process mining lies at the heart of both frameworks, but their approaches to control-flow discovery differ significantly. The Rozinat framework employs advanced algorithms, such as the α-algorithm, to automatically extract control-flow relationships from event logs. These algorithms are designed to uncover causal dependencies and routing probabilities, allowing the framework to model complex branching behaviours. Moreover, Rozinat incorporates decision point analysis, which uses data attributes to derive rules for routing cases through the process. This capability enhances the accuracy of simulation models, particularly in scenarios where decision-making is influenced by multiple factors.

The Tamburis framework, while also focusing on control-flow discovery, takes a more pragmatic approach. It utilizes inductive mining algorithms, such as the Inductive Miner Infrequent (IMi), to handle noisy and incomplete data. The resulting process models are expressed in Business Process Model and Notation (BPMN) diagrams, which are later converted into executable simulation models. Although this approach lacks the advanced decision point analysis found in Rozinat, it is better suited to real-world applications, as it prioritizes robustness and simplicity over theoretical rigor. This trade-off makes the Tamburis framework more accessible to practitioners in fields like healthcare, where users may lack expertise in advanced process mining techniques.

Simulation Model Translation

One of the key challenges in integrating process mining with simulation is the translation of discovered models into executable simulation formats. The Rozinat framework uses Coloured Petri Nets (CPNs) for this purpose, a choice that provides significant expressive power and flexibility. CPNs allow the framework to integrate multiple perspectives, including control-flow, performance, and organisational characteristics, into a single model. This comprehensive approach enables detailed analysis and simulation but requires expertise in CPN tools, which can be a barrier for non-technical users.

In contrast, the Tamburis framework relies on commercial simulation tools like Simul8, which use BPMN diagrams as input. While this approach limits the depth of integration compared to CPNs, it significantly improves usability and accessibility. By leveraging user-friendly tools, the Tamburis framework ensures that domain experts, such as healthcare professionals, can actively participate in the simulation modelling process. However, this reliance on commercial tools may restrict customization and scalability, particularly in scenarios requiring complex, multi-dimensional analysis.

Validation and Iteration

Validation is a critical step in ensuring that simulation models accurately reflect real-world processes. The Rozinat framework incorporates validation mechanism, referred to as "second pass" analysis. In this approach, simulated event logs are generated and compared with the original logs to identify discrepancies. This process allows for continuous refinement of the simulation model, ensuring alignment with observed behaviour and enabling adaptability to process changes. By emphasizing validation, Rozinat provides a robust mechanism for improving model accuracy over time.

The Tamburis framework, on the other hand, focuses on conformance checking as its primary validation technique. Using metrics such as fitness, precision, and simplicity, the framework compares simulation outputs with event logs to verify the accuracy of the initial model. While this approach is effective for initial validation, it lacks the iterative refinement capabilities of Rozinat. As a result, the Tamburis framework may be less adaptable to dynamic processes or environments where process changes occur frequently.

3.2.3.1 Strengths and Weaknesses

Strengths

The Rozinat framework demonstrates remarkable strengths through its comprehensive integration of control-flow, performance, and organisational perspectives. By addressing these key dimensions simultaneously, it provides a holistic approach to process analysis and modelling. The integrated generation of simulation models is a standout feature, significantly reducing manual effort and time required for modelling. This automation is especially beneficial in large-scale or complex projects where manual construction would be labour-intensive and error prone. Additionally, the framework employs iterative

validation, which enhances the accuracy of the models and ensures they remain adaptable to changes in process dynamics.

On the other hand, the Tamburis framework excels in its strong emphasis on data preparation, particularly in handling noisy and incomplete event logs. This focus makes it highly practical for real-world applications, especially in industries like healthcare where data quality issues are prevalent. Its user-friendly tools and specific attention to healthcare applications make it accessible and relevant for practitioners in this domain. Moreover, the framework includes robust conformance checking mechanisms for validating simulation models, ensuring alignment between discovered processes and their real-world counterparts.

Weaknesses

Despite its strengths, the Rozinat framework has notable limitations. It assumes the availability of clean and complete event logs, which is often unrealistic in practical scenarios where data is messy or incomplete. This assumption can limit its applicability in environments with poor data quality. Furthermore, the reliance on CPN (Coloured Petri Nets) tools requires users to have technical expertise, which may pose a barrier to non-technical users or those unfamiliar with these tools. This lack of accessibility could hinder broader adoption in diverse industries.

The Tamburis framework, while strong in data preparation, has a more constrained focus when integrating multiple perspectives into simulation models. Unlike Rozinat, it does not fully incorporate control-flow, performance, and organisational aspects, potentially reducing the depth of its insights. Additionally, the absence of iterative validation mechanisms means that the framework is less adaptable to evolving processes. This limitation could lead to models that become outdated or misaligned with real-world changes over time.

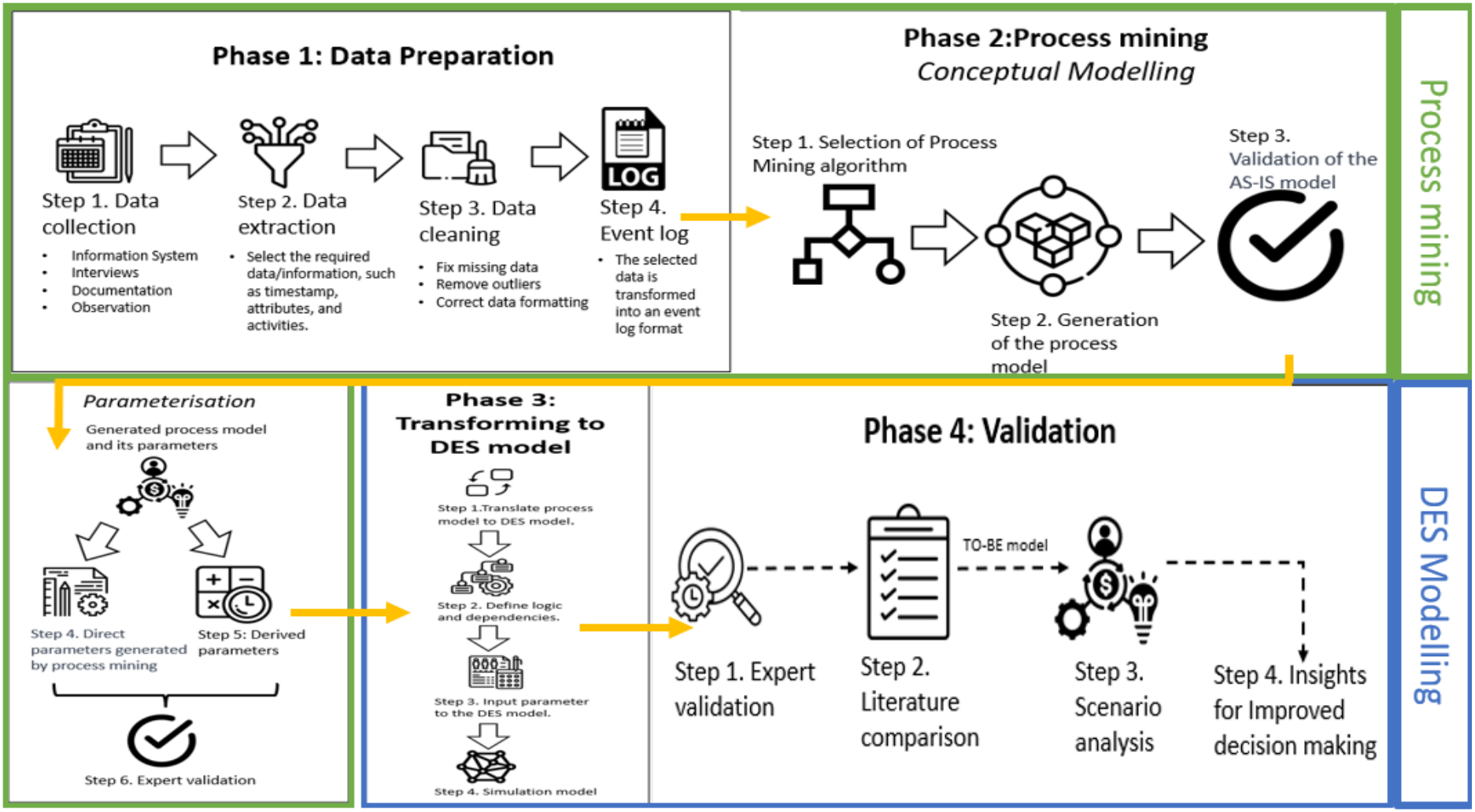


Figure 3.2:Initial Proposed Framework

The phases and steps of the proposed framework (Figure 3.2) are derived from the aspects identified in Section 3.2.2. These aspects were distilled from the comparison of Rozinat et al. (2009) and Tamburis et al. (2020) (Figure 3.1), together with insights from the wider literature. Each aspect was operationalised into a corresponding phase and its steps: for example, data handling informed Phase 1: Data Preparation, which includes the steps of collection, extraction, cleaning, and formatting; process mining informed Phase 2: Process Mining and Conceptual Modelling, which includes algorithm selection, process model generation, and AS-IS validation; simulation model translation informed Phase 3: Transforming to DES, which incorporates parameterisation and model integration; and validation informed Phase 4: Validation, which expert validation.

3.2.4 Phase 1: Data Preparation

In the first stage of the framework, data is prepared for process mining and DES modelling. This incorporates determining the process to analyse, gathering data from information systems, and cleaning and preprocessing it to ensure high quality and consistency. The data preparation phase is critical for successful integration, as it forms the foundation for subsequent stages. A research study by Tamburis et al. emphasizes the necessity of accurate data gathering to achieve precise outcomes. Lamine et al. also highlights the importance of sourcing reliable data.

The data preparation framework consists of four steps: Data Collection, Data Cleaning, Data Extraction, and Event Log Generation. Each step is detailed below, with relevant criteria and techniques derived from literature.

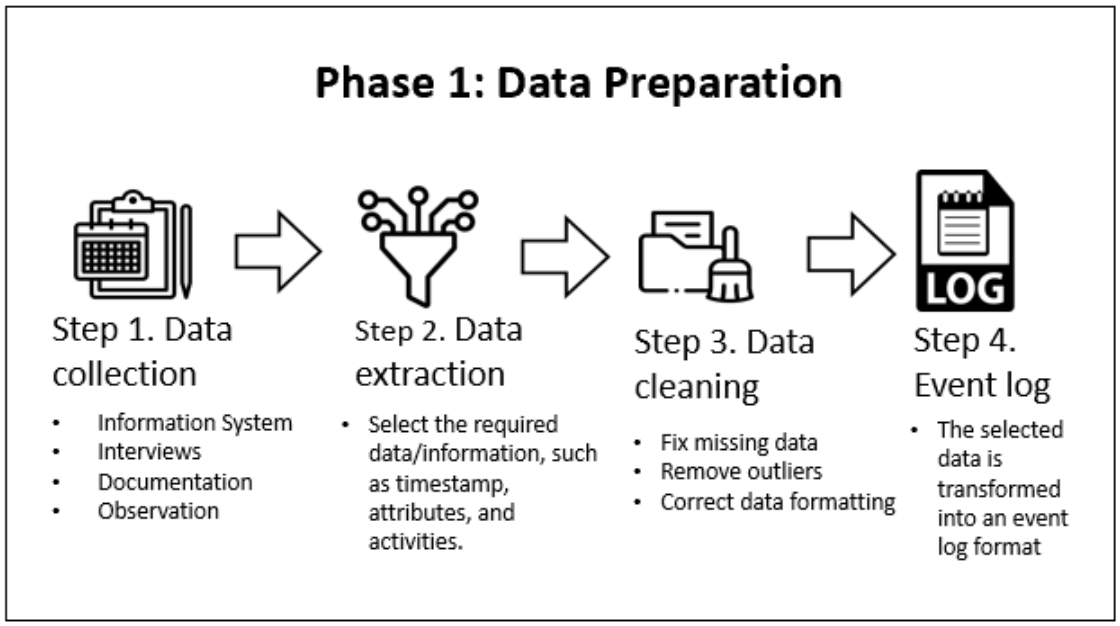


Figure 3.3 Phase 1: Data Preparation Framework derived from literature (Tamburis et al., 2017; Tangkawarow, Sarno, & Fauzan, 2018; Lamine et al., 2015; Phan et al., 2019).

3.2.4.1 Step 1: Data Collection

Data collection involves gathering relevant information from multiple sources, such as information systems, interviews, documentation, and observations. Ensuring the completeness and accuracy of the collected data is vital for subsequent steps. The methods used should capture a comprehensive view of the process activities, timestamps, and relevant attributes. Tamburis et al. stress the importance of using reliable sources to ensure high-quality data.

3.2.4.2 Step 2: Data Extraction

After collection, data extraction begins. This step involves identifying and retrieving relevant event data from the cleaned dataset, such as timestamps, attributes, and other critical fields. Several techniques can be applied during this process.

One common approach is querying databases, where SQL queries or similar tools are used to extract data efficiently from large datasets (Mesabbah & McKeever, 2019). Another technique is log conversion, which involves converting raw data into event logs that are compatible with process mining tools, often requiring the merging of data from

multiple sources (Phan et al., 2019). Automated extraction is also widely used, employing scripts or software tools to ensure consistent and efficient data extraction (Lamine et al., 2015).

To further enhance the structure of the extracted data, grouping and ordering rows by resource and sorting events by start time is recommended (Lugaresi et al., 2019). Lastly, combining these methods with a manual review helps to validate the results and ensure the accuracy of the extracted data (Johnson et al., 2019). Manual review is often used to complement automated techniques. Despite the efficiency of automated methods, human expertise is essential to verify the accuracy and completeness of the extracted data.

3.2.4.3 Step 3: Data Cleaning

Data cleaning aims to improve the quality and consistency of data by detecting and resolving errors, inconsistencies, or missing values. This crucial step ensures that the dataset is accurate and reliable for further analysis. Several key techniques are commonly applied during the data cleaning process.

One important technique is to handle typos, which involves detecting and automatically correcting typographical errors using tools such as MATLAB (Tangkawarow, Sarno, & Fauzan, 2018). Another technique is to handle outliers, where outliers in event duration times are identified and removed using visual tools like boxplots (Tangkawarow, Sarno, & Fauzan, 2018). Handling missing values is equally critical, which involved filling in missing data by estimating start and finish times based on dataset patterns or imputing average values (Tangkawarow, Sarno, & Fauzan, 2018; Sitova & Pecerska, 2019).

Additionally, correcting incorrect data by cross-referencing inaccurate entries with reliable sources or consulting domain experts helps improve data accuracy (Johnson et al., 2019; Pourbafrani et al., 2022). It is also essential to standardize formats to ensure uniform representation of data, such as timestamps and event identifiers (Phan et al., 2019). Another critical step is to filter irrelevant data, which involves removing non-essential information to focus on relevant aspects of the dataset (Johnson et al., 2019). Lastly, addressing inconsistencies enhances data precision and ensures uniformity across event logs (Johnson et al., 2019; Mesabbah & McKeever, 2019).

3.2.4.4 Step 4: Event Log Generation

Once data is extracted, an event log is generated. The event log should contain essential fields, such as Case ID, Event ID, Timestamp, and Description, to provide a structured representation of process events. Best practices include adopting a Standard Event Log Format, which involves including fields such as case ID, activity name, timestamp, and resource (Phan et al., 2019; Tangkawarow, Sarno, & Fauzan, 2018). Additionally, maintaining Consistent Timestamp Formats is crucial, and standardized date and time formats should be used to ensure clarity and accuracy (Lamine et al., 2015). It is also important to have Unique Identifiers for each event to prevent duplication and ensure that every event is uniquely identifiable (Mesabbah & McKeever, 2019). Finally, creating Custom Event Logs tailored to specific analysis requirements is highly recommended to align the data with the objectives of the analysis (Lamine et al., 2015).

3.2.5 Phase 2: Process Mining

In this phase, process mining techniques are applied to the event log data to discover the process flow and generate performance parameters. The results of the process mining are visualised and analysed using tools like process maps. This phase provides insights into the current state of the process and forms the foundation for conceptual modelling, enabling a structured representation of the process flow.

Conceptual Modelling

Conceptual modelling allows for creating an abstract representation of the process, describing its key elements and relationships, and serving as a basis for analysing behaviour, identifying patterns, and understanding dependencies between activities. As emphasized by Mans et al., the role of conceptual modelling in process mining is critical for representing processes in a clear and structured way. The conceptual modelling process can be divided into three detailed steps as the following:

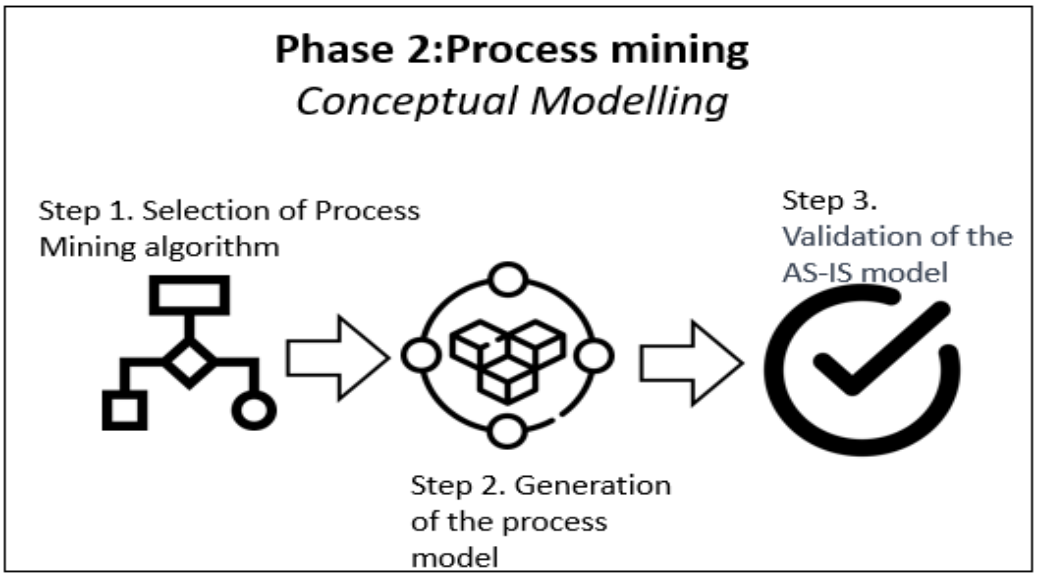


Figure 3.4 Phase 2: Process Mining - Conceptual Modelling Framework
Derived from Antunes et al., 2019; Lugaresi et al., 2019; Khodyrev & Popova, 2014; Johnson et al., 2019; Van Hulzen et al., 2021; Cho et al., 2019.

3.2.5.1 Step 1: Selection of Process Mining Algorithm

Selecting an appropriate algorithm is crucial for discovering meaningful process models from event logs. This decision depends on the purpose of analysis and characteristics of the logs. Different algorithms offer unique strengths that cater to specific needs and contexts.

Heuristic Miner: Chosen for its robustness in handling noisy and incomplete logs, especially in healthcare contexts involving complex processes (Antunes et al., 2019).

Fuzzy Miner: Selected for simplifying complex processes by focusing on significant activities and transitions, particularly useful for stakeholder-friendly visualisations (Lugaresi et al., 2019).

Alpha Miner: Preferred for its transparent approach, making it suitable for educational purposes and identifying fundamental process structures (Khodyrev & Popova, 2014).

Inductive Miner: Opted for its ability to process large datasets efficiently and produce generalized models suitable for varied pathways (Johnson et al., 2019).

Across these studies, common criteria for algorithm selection include the ability to handle noise, accommodate process complexity, work with varying data quality, generate

desired model outputs, and provide ease of interpretation (Antunes et al., 2019; Lugaresi et al., 2019; Khodyrev & Popova, 2014; Johnson et al., 2019; Van Hulzen et al., 2021).

3.2.5.2 Step 2: Generation of the Process Model

After selecting the algorithm, the process model is derived from the event log. This step involves analysing the event log's chronological records, timestamps, and other attributes to uncover key process flows. The discovered models can be represented using notations like Petri nets or Business Process Model and Notation (BPMN). Various process discovery techniques highlight the importance of using appropriate representations to ensure clarity and actionable insights (Antunes et al., 2019; Lugaresi et al., 2019).

3.2.5.3 Step 3: Validation of the AS-IS Model

The generated process model is validated by comparing it with the original event log to identify discrepancies and errors. Feedback from domain experts is incorporated to improve accuracy and reliability, ensuring the model aligns closely with the actual process. As highlighted by Cho et al., model validation and refinement are critical for improving process reliability and ensuring it accurately represents observed behaviour.

Parameterisation

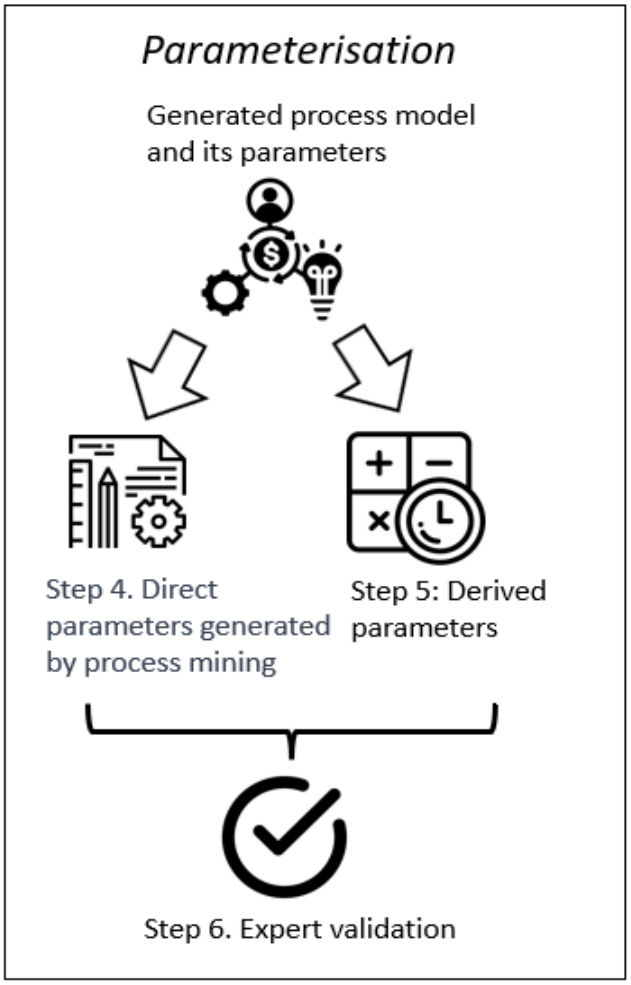


Figure 3.5 Parameterisation Framework Derived from Literature (Phan et al., 2019; Van Hulzen et al., 2020; Abohamad et al., 2017; Pegoraro et al., 2021; Tamburis & Esposito, 2020; Liu et al., 2015; Uehara & Hiraishi, 2019; Werner-Stark et al., 2011; Antonio et al., 2022)

3.2.5.4 Step 4: Direct Parameters Generated by Process Mining

In this phase, process mining is used to extract key parameters that reflect the real-world process being analysed. The generated process model serves as a visual representation of the activities, transitions, and resources involved. Direct parameters are obtained from the event log data and provide a basic understanding of the process flow. These include:

Activity Durations: The time taken for each activity to be completed within the process. Tools like Disco and ProM extract these values from event log timestamps (Phan et al., 2019). As emphasized by Van Hulzen et al. (2020), direct parameters such as activity

durations are critical for understanding the behaviour of the process and identifying potential inefficiencies.

Frequency of Activities: The number of times a specific activity occurs within the process. This metric helps identify common or rare activities and prioritize optimisation efforts (Abohamad et al., 2017).

Throughput Time: The total time it takes for a case to pass through the entire process. This metric is important for evaluating overall process efficiency and identifying delays (Pegoraro et al., 2021).

These parameters are directly measurable and are crucial for understanding the basic flow and time efficiency of the process.

3.2.5.5 Step 5: Derived Parameters through Additional Calculations

Once the direct parameters have been obtained, additional derived parameters are calculated to offer deeper insights into process performance. These parameters often require more advanced analysis methods, including statistical calculations or aggregations. Derived parameters include:

Transition Probabilities: The likelihood of moving from one activity to another within the process. These help in understanding the flow and dependencies between activities (Tamburis & Esposito, 2020).

Resource Utilization: The frequency and extent to which specific resources (e.g., staff or equipment) are used during the process. This parameter is vital for assessing resource bottlenecks and optimisation (Liu et al., 2015).

Cycle Times: The total time it takes for a cycle of activities to complete, from start to end. This helps assess the efficiency and timeliness of the process (Uehara & Hiraishi, 2019).

3.2.5.6 Step 6: Expert Validation

After obtaining and deriving the parameters, the next phase involves expert validation to ensure the accuracy, relevance, and usability of the process model. The following criteria are used for expert validation:

Accuracy of Process Representation: Experts assess whether the parameters—such as transition probabilities, activity durations, and cycle times accurately represent the real-world flow of activities. They validate if the relationships and timing between activities are correctly captured in the process model, ensuring that the model's transitions, durations, and overall process cycle reflect real-world data (Werner-Stark et al., 2011).

Clinical or Operational Relevance: Experts in fields like healthcare evaluate whether the model captures essential activities and decision points accurately, particularly through parameters like resource utilization, frequency of activities, and throughput time. These parameters help experts verify if the model includes the critical decision points and activities that influence real-world clinical or operational decisions (Antonio et al., 2022). They ensure that the model's parameters align with key objectives.

3.2.6 Phase 3: Transformation To DES modelling

In the third phase, the process model generated by process mining is transformed into a DES model. This step includes the mapping of the process activities and events to DES entities and events, defining the state variables and transition rules, as well as specifying simulation parameters and inputs. The process model transformation phase presents an important step in the integration as it makes sure that the DES model correctly depicts the real behaviour of the process. When moving the outputs from process mining to DES, a number of factors are considered.

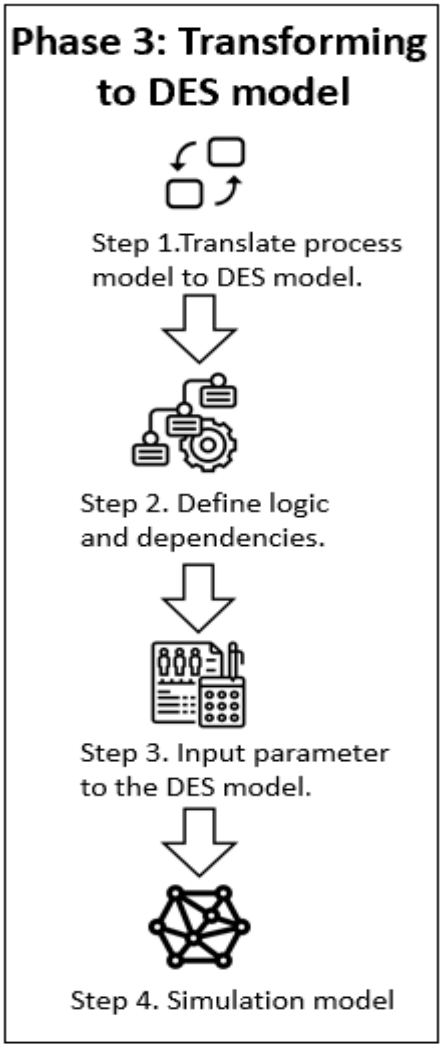


Figure 3.6 Steps Involved in Transforming Process Models into DES Framework
Derived from Antunes et al., 2019; Lugaresi et al., 2019; Rozinat et al., 2009; Mans et al., 2013; Pegoraro et al., 2021; Khodyrev & Popova, 2014; Uehara & Hiraishi, 2019; Tangkawarow et al., 2018.

3.2.6.1 Step 1: Translate Process Model to DES Model

The first step involves translating the process model derived from process mining into a DES model using appropriate tools and methodologies.

Activity Mapping: Antunes et al. (2019) explained that each activity in the process model is represented as a discrete event in the simulation model. Resources required for each activity are identified and mapped into the DES model to reflect real-world constraints. Similarly, Lugaresi et al. (2019) emphasized that transitions in the Petri Net process model correspond to events in the DES model, and places in the Petri Net are mapped to queues or buffers. Rozinat et al. and Mans et al. further highlighted the importance of accurately capturing these mappings to ensure that the DES model mirrors the underlying process model effectively.

3.2.6.2 Step 2: Define Logic and Dependencies

This step focuses on defining the rules and dependencies necessary for accurately simulating the dynamics of the process.

Routing Logic: Routing logic determines how entities or tasks are directed through different pathways within the process. For example, in a healthcare scenario, routing logic ensures that patients are sent to the appropriate treatment pathway based on their diagnosis. This logic incorporates probabilities or conditions derived from process mining outputs, ensuring that the simulation accurately reflects real-world decision-making processes (Lugaresi et al., 2019; Antunes et al., 2019).

Time Dependencies: Time delays between activities, such as transportation or setup times, are incorporated into the DES model to simulate real-world waiting times accurately (Tangkawarow et al., 2018).

3.2.6.3 Step 3: Input Parameters to the DES Model

The third step involves integrating parameters from the process model into the DES model. Parameterisation ensures that timing parameters (e.g., activity durations and inter-

arrival times) and resource availabilities are accurately reflected. For example, patient arrival times and treatment durations ensure that the DES model mirrors real-world operations (Khodyrev & Popova, 2014). Resource constraints, such as defining capacities and availability based on process model data, significantly impact the efficiency of manufacturing processes by ensuring machines and operators are optimally allocated (Antunes et al., 2019).

3.2.6.4 Step 4: Execute Simulation Model

The fourth step involves executing the DES model and analysing its outcomes. Simulation Execution employs tools such as Simul8, AnyLogic, or Arena to run the model. Adjustments are made based on discrepancies between the simulated and real-world data (Khodyrev & Popova, 2014). Validation ensures that the simulated outcomes align with real-world data or expert insights. For instance, Uehara & Hiraishi (2019) compared simulation results against event logs to verify the accuracy of the DES model.

3.2.7 Phase 4: Validation

Validation is a phase to ensure the reliability and accuracy of the simulation model by comparing its results with process mining outcomes obtained in earlier phases. This comparison identifies discrepancies and validates the accuracy of the discrete event simulation (DES) model. If significant differences are found, further refinement of the model is necessary. According to Mesabbah and McKeever (2018) and Tamburis (2019), comparing simulation results with real-world observations or process mining outcomes is essential to ensuring the reliability and credibility of the simulation model. The validation framework consists of four main steps:

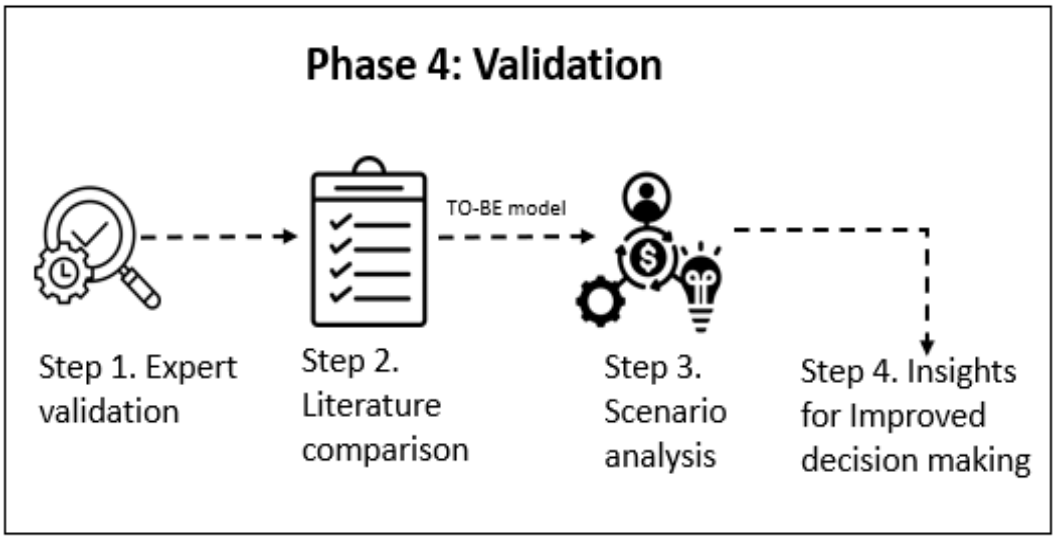


Figure 3.7 Validation Framework

Derived from Mesabbah and McKeever, 2018; Tamburis, 2019; Antunes et al., 2019; Lugaresi et al., 2019; Khodyrev & Popova, 2014; Rozinat, 2008; Pourbafrani, Jiao, and van der Aalst, 2022; Petitdemange et al., 2020; Pegoraro et al., 2021; Uehara & Hiraishi, 2019.

3.2.7.1 Step 1: Expert Validation

The first step involves validating the baseline simulation model through expert assessment. Domain experts reviewed the simulation structure, assumptions, and baseline results to ensure that the model adequately represented the actual healthcare processes. This step ensured that the model was clinically meaningful and reliable before proceeding to literature comparison and scenario analysis (Petitdemange et al., 2020; Tamburis and Esposito, 2019).

3.2.7.2 Step 2: Literature Comparison

The simulation results are then compared with findings reported in the literature to establish external validity. Literature comparison provides an additional layer of credibility by aligning the model outputs with prior studies or known performance benchmarks (Mesabbah and McKeever, 2018; Pegoraro et al., 2021). This step ensures that the simulation model is not only internally valid but also consistent with broader empirical evidence.

3.2.7.3 Step 3: Scenario Analysis

Once the model is validated, scenario analysis is conducted to explore the effects of alternative strategies or system configurations. Scenario testing enables the identification of opportunities for process improvement, allowing for the assessment of interventions such as early diagnosis, resource allocation, or pathway redesign (Lamine et al., 2015; Abohamad et al., 2017).

3.2.7.4 Step 4: Improved Decision-Making

The final step involves drawing insights for improved decision-making. The outputs from scenario analysis provide actionable knowledge that can guide policy makers and healthcare managers in redesigning processes or allocating resources more effectively (Mesabbah and McKeever, 2018; Uehara & Hiraishi, 2019).

3.2.8 Overall Discussion

The literature has brought to light specific limitations and gaps in existing research. One significant challenge is related to data availability and quality. Many studies rely on event log data collected from information systems or data collection efforts, which may exhibit inconsistencies, missing data, or limited coverage. These issues can impact the accuracy and reliability of the integrated approach. Addressing these data-related challenges requires careful consideration and the development of appropriate data cleaning and preprocessing techniques. To tackle these challenges, we suggest the addition of further steps such as Data Protocol and Data Profiling to ensure data integrity and quality. Data Protocol would establish rules for standardized data collection and validation, ensuring consistency and correct formatting of the data before its use. Data Profiling can involve examining data distributions, detecting anomalies, and identifying missing or inconsistent data, providing a clearer understanding of the dataset before applying process mining.

Selecting the appropriate algorithm is a critical step in constructing an accurate process model that effectively captures essential processes. It is important to consider the data conditions and objectives of the process mining initiative to ensure the resulting model serves its purpose. We propose adding steps for comparing process models generated by different algorithms, which will help in selecting the most suitable one. Additionally, guidance for algorithm selection based on dataset characteristics such as complexity, and noise levels will further refine the process.

Finally, the proposed framework will be implemented in the case studies, allowing us to identify any challenges and opportunities to improve the framework further based on the insights gained from the case studies. By including these additional steps, we aim to enhance the robustness of the framework, ensuring that the final model is both accurate and applicable to real-world scenarios.

CHAPTER 4

EXPLORING USABILITY OF PROCESS MINING IN DES MODELLING USING MEXICAN BREAST CANCER DATASET

4.1 Chapter Overview

This chapter explores the integration of process mining into DES modelling for breast cancer pathways in Mexico, employing the proposed framework outlined in Chapter 3. We elaborate on our methodology to support conceptual modelling, parameterisation, and validation of the simulation model. Importantly, the implementation presented reflects refinements introduced through this and subsequent case studies, which were consolidated into the enhanced framework described in Chapter 7. As a result, certain steps (e.g., algorithm selection and validation) are organised differently from the initial literature-based framework in Chapter 3 but remain consistent with the enhanced framework.

4.2 Introduction to Breast Cancer

Breast cancer is a chronic disease that begins in the cells of the breast, and women are at greater risk of developing it. The World Health Organisation (WHO) in 2020 reported that breast cancer is the most prevalent cancer among women worldwide and is second only to skin cancer as a cause of cancer-related death in women (Hanf & Kreienberg, 2020). In Mexico, limited access to timely diagnosis and effective treatment has contributed to a higher number of deaths from breast cancer, particularly among women in the public healthcare system.

There are a number of breast cancer pathways, which include stages that support the identification and treatment of the disease. Typically, the patient pathway begins with the identification of symptoms such as a lump in the breast, nipple discharge, or changes in the size or shape of the breast. When such symptoms are detected, the patient is referred to a breast cancer specialist for a more detailed examination.

Commonly, the first stage in the pathway is diagnosis, which involves a set of tests and examinations aimed at identifying whether a woman has breast cancer. This category of tests may include mammography (X-rays), CT scans, MRIs, ultrasounds, or biopsies conducted on the patient to screen for cancer. The tissue samples are evaluated by a pathologist, who determines the grade of the cancer and the tumour receptor status. These are used to decide the most suitable treatment regimen.

The second stage in the pathway is staging, which identifies the extent and spread of the cancer. This is critical because the stage of the cancer determines the treatment plan and the patient's prognosis. Staging combines imaging and laboratory tests, which may include blood tests, bone scans, CT scans, MRIs, or PET scans.

Once the cancer is staged, the next stage is treatment. The treatment plan depends on the stage of the cancer, as well as the patient's overall health and preferences. The standard forms of treatment for patients diagnosed with breast cancer include surgery, chemotherapy, radiation therapy, targeted therapy, and hormone therapy. The aim of treatment is to eliminate or kill the cancer cells while causing minimal harm to healthy tissue.

Following treatment, patients usually receive post-treatment care and monitoring to ensure there is no cancer recurrence. This may include follow-up visits, imaging, and blood tests to check for signs of return.

Late-stage presentation, caused by delays in diagnosis and lack of access to standard treatment, is one of the main reasons for increasing breast cancer mortality rates particularly in low-income countries such as Mexico. This emphasizes the need for early diagnosis and timely access to appropriate treatment (Unger-Saldaña et al., 2020).

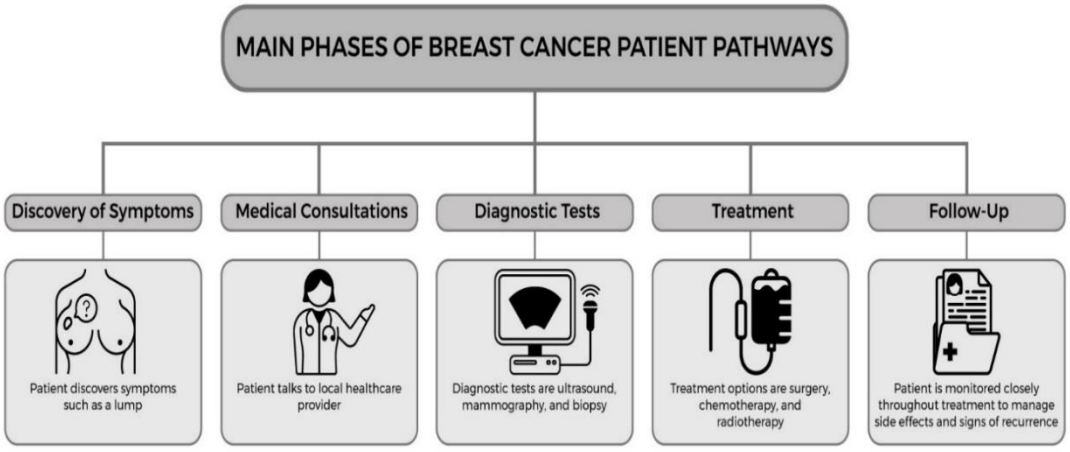


Figure 4.1 Phases of breast cancer patient pathways (adapted from Unger-Saldaña et al., 2020).

4.2.1 Delay in breast cancer pathways

The time between when a patient first discovers a breast lump or other symptom and when they receive a definitive diagnosis and treatment is referred to as the delay in breast cancer pathways. Delays can occur at any point along the pathway, including patient, diagnostic, and treatment delays.

The interval between finding a breast lump or other symptom and seeking medical assistance is referred to as patient delay. This delay may be caused by a lack of understanding of breast cancer symptoms, fear or humiliation, cultural or societal hurdles, or financial limits. People may sometimes put off seeking medical assistance if they feel the symptom will resolve itself.

Diagnostic delay is the time between patients seeking medical attention and diagnosis. Lack of diagnostic tools like mammography or biopsies or long wait periods for expert consultations may explain this delay(Unger-Saldaña et al., 2019).

Treatment delay occurs after diagnosis. The time between receiving a diagnosis and starting therapy is referred to as treatment delay. This may be caused by a shortage of healthcare resources or extended wait times for appointments with oncologists or other

specialists. In some cases, treatment delays may be influenced by patient preferences, such as a desire to explore alternative treatments or concerns about the side effects of conventional therapies (Unger-Saldaña, 2014; Unger-Saldaña et al., 2018).

Delays in breast cancer treatment pathways can have major consequences for patients, as early detection and timely treatment are critical for ensuring earlier treatment initiation and improving clinical outcomes. Delays are more common in low- and middle-income countries due to limited healthcare resources and infrastructure. Reducing delays in breast cancer pathways requires a holistic approach, including raising public awareness of breast cancer symptoms, improving access to diagnostic tools and treatment facilities, and shortening wait times for appointments with healthcare specialists (Unger-Saldaña et al., 2015).

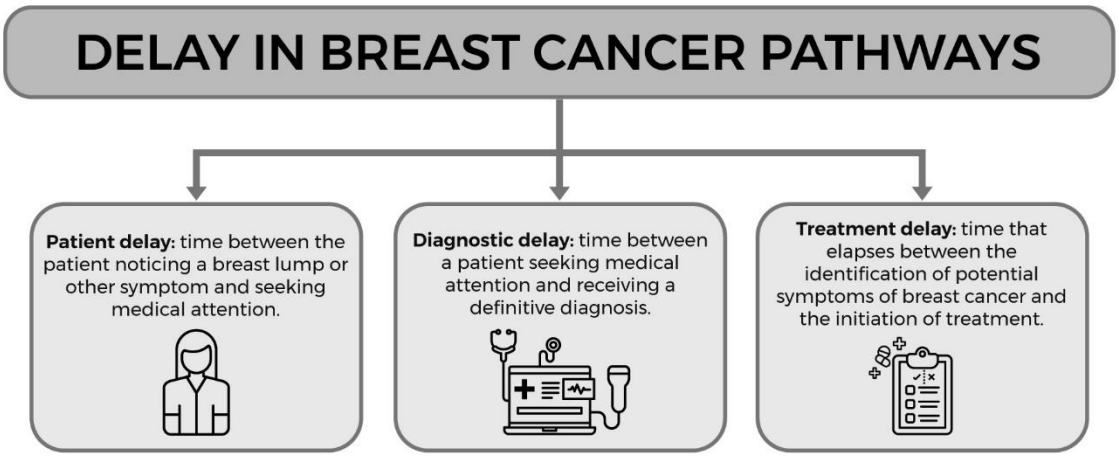


Figure 4.2 Different types of delay in breast cancer pathways (adapted from Unger-Saldaña et al., 2020).

4.3 Aims of case study

The purpose of this case study is to accomplish four objectives. First, we aim to create a process model for the breast cancer pathways in Mexico using process mining techniques. Second, we intend to investigate potential delays in the pathways by analysing patient subgroups based on the type of healthcare service they utilized during presentation to help identify any differences in pathways and time delays. Third, we will validate the process model and parameters and transform it into a discrete-event simulation (DES) model. Using this model, we will perform scenario analyses to assess how reducing delays may influence earlier treatment initiation across different stages of breast cancer. Finally, we will examine the potential benefits and challenges of using process mining in a DES model.

4.4 Dataset

4.4.1 Overview of the dataset

This study utilized a secondary dataset provided by a research collaborator, who is an oncologist based in Mexico. The dataset comprises of data on breast cancer patients collected during a previous research project. Data was collected through interviews with patients and medical record reviews using standardized questionnaires between June 1, 2016, and May 31, 2017, at two public hospitals in Mexico City: the Instituto Nacional de Cancerología (INCan) and the General Hospital de México "Eduardo Liceaga" (HGMEL). INCan, as the largest comprehensive cancer centre in Mexico, provides specialized services in medical oncology, hematology, and radiation oncology, among others, and is a public institution under the Mexican Health Secretariat. HGMEL, also run by the Secretariat of Health, offers expertise and support in 39 medical services, providing essential services in the second and third stages of healthcare.

The dataset contains 688 cases, providing information on demographics, consultations, diagnostics, and treatments. Face-to-face interviews were conducted by trained psychology and medical students, and the medical files of primary care services were reviewed to obtain the exact date of the first medical consultation. The initial study aimed to measure the time intervals of care for breast cancer patients, identify obstacles

associated with delays in care intervals, and explore patient experiences of using healthcare services. All data collected for this study has been anonymized to protect the privacy of the patients, and the original identities of the patients are unknown to the researchers, and the collaborators obtained ethical approval for data collection (Ref. INCAN/Of.CEI 572/15 and Ref. INCAN/Of.CI 485/15). A previous publication by the collaborator, describing delays in breast cancer diagnosis among young Mexican women linked to a lack of suspicion by healthcare providers at first presentation (Unger-Saldaña et al., 2019), provides more information about this dataset. The current study received ethical approval from the Ethics Unit at the University of Sheffield (Project: 040225) to receive and use the data for process mining.

4.4.2 Variables in the dataset

The dataset comprises of 181 variables that are categorized into four parts, namely: (1) sociodemographic information of the patient, (2) time intervals, (3) health services, and (4) perceived barriers. Each section is summarized in the tables presented below.

Table 4.1Sociodemographic characteristics (Redacted)

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

Table 4.2 Time Intervals and Health Service Use (Redacted)

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

Table 4.3 Perceived barriers (Redacted)

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

The three tables above summarised the sociodemographic characteristics, time intervals, and perceived barriers for breast cancer patients in Mexico. These aggregated results have been redacted in the public version to preserve confidentiality.

4.5 Implementation of the Proposed Framework Using the Mexican Breast Cancer Dataset

The proposed framework (Figure 3.2) provides a structured approach to integrating process mining with discrete event simulation (DES) modelling. It consists of four key phases: Data Preparation, Process Mining, Transformation to DES Model, and Validation. This section demonstrates how the framework was applied to the Mexican breast cancer dataset. We outline the implementation of each phase, detailing the methods, tools, and results, showcasing the framework's effectiveness in addressing real-world healthcare dataset.

4.5.1 Phase 1: Data preparation

The preparation phase is a critical step in the process mining analysis. The objective of this phase is to ensure that data is available, reliable, and structured in the format required for process mining, which is typically event logs. In this case study, a secondary dataset was obtained from a collaborator, which required careful checking and formatting to ensure that it was suitable for analysis. This phase consisted of six steps: (1) Data protocol, (2) Data collection, (3) Data profiling, (4) Data extraction, (5) Data cleaning, and (6) Event log generation.

4.5.1.1 Step 1: Data Protocol

The data protocol establishes the structure, format, and confidentiality procedures that govern how datasets are prepared and shared for process mining analysis. In this case study, the protocol was predefined by the collaborating institution, which determined the scope of data to be extracted and the way it was delivered. While this step was not designed by the research team, it served as the foundation for subsequent phases by ensuring that datasets were shared securely and could be reformatted into event logs for analysis.

4.5.1.2 Step 2: Data collection

Data Acquisition

We obtained a secondary dataset from a collaborator in Mexico. The data was provided to us in the form of a .dta file, which was then imported into the R programming environment for data manipulation. This dataset comprised the following key files:

1. **Cancer Patient Demographics–** This file contains demographic records of cancer patients, capturing aspects of their background and medical history. The dataset includes columns such as
   ID, Cancer_Type, Hospital, History_of_Benign_Condition, Municipality_of_Residence,Patient_Age,Marital_Status,Employment_Status, Patient_Income, and Occupation_Categories.
   The data spans numerous rows and covers a range of socio-economic and geographic data related to patients undergoing cancer treatment. It also includes categories for education, literacy, transportation, and household information, providing demographic profile.

2. **Cancer Patient Suspicion Data**– This file focuses on cancer patients' early symptom discovery and healthcare-seeking behaviours. The dataset includes columns such as *First_Symptom, Perceived_Seriousness_of_Symptoms, Reason for_Seeking_Healthcare, Patient_Delay_Perception.* It also contains detailed data regarding *Access_to_Healthcare, Reasons_for_No_Access and Reasons_for_Patient_Delay.* The data captures a wide range of factors contributing to patient behaviour, providing a detailed account of how symptoms are identified, how seriously they are perceived, and what leads patients to seek or delay healthcare.
3. **First_Service_Data**– This file captures detailed information on the first medical service used by cancer patients. It includes columns such as
   SS1_1st_Service_Used,SS1_Type_of_First_Service, SS1_Reason_for_Not_Using_Entitled_Service, SS1_State_and_Municipality_of_First_Service, SS1_Travel_Time, SS1_Transportation_Type, SS1_Transportation_Expenses,

SS1_Consultation_Expenses, SS1_Specialty_of_Doctor, SS1_Studies_Requested (e.g., mammography, breast biopsy, Pap smear), SS1_Date_of_Consultation, and Patient_Rating_of_Service.

This dataset provides insights into patient experiences during their first healthcare service encounter, covering factors such as accessibility, diagnostic impressions, and reasons for choosing or not using the services.

4. **Second_Service_Data**– This file provides detailed information on the second medical service used by cancer patients. Key variables include *SS2_Type_of_2nd_Service_Used,SS2_Travel_Time_to_2nd_Service, SS2_Type_of_Transportation, SS2_State_and_Municipality_of_2nd_Service, SS2_Date_of_Consultations, SS2_Specialty_of_Doctor_in_Service_2, SS2_Directed_Clinical_Exam, SS2_Diagnostic_Impression_of_2nd_Doctor, and whether SS2_Non_Cancer_Medications were prescribed. The file also covers Consultation_Expenses, Number_of_Consultations_in_Service_2, and SS2_Patient_Rating_of_Service_2.*

   This dataset captures the patient's journey through their second service encounter, offering insights into the type of healthcare accessed, time taken, transportation used, diagnostic impressions, and overall satisfaction with the service.

5. **Third_Service_Data** – This Excel file contains records of cancer patients' third medical service encounter. The dataset includes columns such as SS3_Type_of_3rd_Service_Used, SS3_Travel_Time_to_3rd_Service, SS3_State_and_Municipality_of_3rd_Service, SS3_Specialty_of_Doctor_in_Service_3, SS3_Directed_Clinical_Exam_in_3rd_Consultation, SS3_Diagnostic_Impression_of_3rd_Doctor, and whether SS3_Non_Cancer_Medications were prescribed. It also includes Consultation_Expenses, Number_of_Consultations_in_Service_3, Date_of_First_Consultation, Date_of_Last_Consultation, and SS3_Patient_Rating_of_Service_3.

   This dataset offers comprehensive insights into the third level of care accessed by patients, examining service type, diagnostic impressions, treatment continuity, and patient satisfaction. It provides a detailed view of how patients progress

through multiple healthcare services, contributing to analyses of healthcare delivery and patient outcomes over time.

6. **Diagnostics_and_Treatments_Data**– This file offers comprehensive data on cancer diagnostics and treatments for patients, with a focus on the processes leading up to and during their care at cancer hospitals. Key columns include *Number_of_Health_Services_Used_Before_Cancer_Hospital, Number_of_Consultations_Before_Arrival_at_Cancer_Hospital, Type_of_Diagnostic_Studies (e.g., Biopsy, Mammography, Ultrasound), Cost_of_Diagnosis, and Time_Between_Tests_and_Results.*
   The dataset also provides details on *Referral_Services, Perceived_Barriers_to_Healthcare (such as long wait times, lack of money, and administrative barriers), Perceived_Barriers_Most_Important, and Expenses_Incurred. Furthermore, it includes patient experiences with Screening_Tests and data on Cancer_Stage, Treatment_Type, and Biopsy_Results. Additionally, several time intervals, such as the time between Symptom_Discovery_and_Treatment_Start and Diagnosis_Confirmation_and_Treatment, are recorded.*

Data Delivery Process

The collaborator shared these datasets via a secure file-sharing platform, with specific instructions regarding data confidentiality and compliance with the institution's data usage policies. Upon receiving the files, we conducted a preliminary check to ensure all files were intact, uncorrupted, and in their correct formats.

4.5.1.3 Step 3: Data Profiling

This step was introduced as an additional refinement to the initial framework based on the requirements of the case study. While not explicitly stated in the earlier generic framework, data profiling was carried out to systematically assess the quality and structure of the dataset before proceeding to extraction and analysis. It involved examining the dataset to understand its structure and content, detecting missing values,

identifying duplicates, and verifying data types to ensure that the subsequent steps produced a reliable event log.

Assessing Data Types

Each variable in the dataset is assigned a specific data type to ensure accurate analysis and interpretation. The primary data types identified in the dataset are:

- **Character (chr):** Used for categorical data, including demographic information, medical history, and healthcare access details. Many categorical variables are encoded with numeric values representing different categories (e.g., cancer type, marital status, means of transportation).
- **Numeric:** Represents continuous and discrete numerical values, such as patient income, number of consultations, and medical expenses. These variables are crucial for statistical analysis and comparisons.
- **Boolean (chr with binary encoding):** Variables indicating binary choices, such as "Yes/No" or "0/1" (e.g., history of benign conditions, literacy, employment status, and access to healthcare services).
- **Date:** Used for time-based variables, including symptom discovery, first consultation, diagnostic confirmation, and treatment start dates. These variables are essential for analysing time intervals in the patient journey.

The full list of variables, their descriptions, and corresponding data types can be found in the appendix.

Table 4.4: Example of Variables and Their Corresponding Data Types (Redacted)

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

Detecting Missing or Incomplete Data

As part of data profiling, we analysed missing values across key date variables related to the patient journey. Identifying missing or incomplete data is crucial for ensuring accurate analysis and minimizing biases.

No Missing Values:
Several critical variables, such as Date of first medical consultation, Date of cancer diagnosis confirmation (biopsy report), Date of first auxiliary diagnostic study, Date of 2nd diagnostic test, and Date of first consultation in cancer service, have complete data. This ensures reliability in analysing the diagnostic and early consultation processes.

Moderate Missing Data:
The Date of symptom discovery or first abnormal screening test has 92 missing values, indicating some patients may have entered the healthcare system without documented symptom onset. Similarly, Date of biopsy has 327 missing values, which may suggest that biopsy records are incomplete or some patients did not require a biopsy.

High Missing Data:
The Date of treatment start has 562 missing values, the highest among all variables. This could indicate incomplete treatment records, delays in data entry, or patients not proceeding to the treatment phase within the dataset's timeframe.

Since missing data can impact time-based analysis, appropriate handling strategies are necessary. A detailed explanation of how we addressed missing values is provided in a later section.

```
dtafileMissingVal <- read_stata("dtafile.dta", encoding = "latin1")
# Create a summary of missing values
missing_values <- dtafileMissingVal %>% summarise_all(~sum(is.na(.)))

# Combine labels and missing values into a data frame for display
missing_values_df <- data.frame(
  Label = unlist(var_labels),
  Missing_Values = as.numeric(missing_values)
)
# View the result
missing_values_df
write.csv(missing_values_df, "missing_values.csv", row.names = FALSE)
```

Table 4.5:Summary of Missing Values in Key Date Variables

Label	Missing_Values
ID	0
Date of symptom discovery or first abnormal screening test	92
Date of first medical consultation for symptom or abnormal screening test	0
Date of cancer diagnosis confirmation (biopsy report)	0
Date of first auxilliary diagnostic study	0
Date of 2nd diagnostic test	0
Date of biopsy	327
Date of first consultation in cancer service	0
Date of treatment start	562

Detecting Duplicates

Another critical check during the review of the data structure is identifying duplicate entries. Duplicates occur when the same record appears more than once in the dataset, either due to data entry errors or duplication during data collection. The code provided is written in R and is used to detect duplicate rows in a dataset read from a Stata file ("dtafile.dta").

First, the dataset is loaded using the read_stata function with Latin-1 encoding. Then, the duplicated function is used to identify duplicate rows. The code extracts and prints these duplicate rows, but based on the output, there are none in the dataset.

The sum function is used to count the number of duplicate rows, and the result is printed. The output shows "Number of duplicate rows: 0", indicating that no duplicate rows were found in the dataset.

```
dtafile <- read_stata("dtafile.dta", encoding = "latin1")

# Check for duplicate rows in the dataset
duplicate_rows <- dtafile[duplicated(dtafile), ]

# Display duplicate rows
print(duplicate_rows)

# Count the number of duplicate rows
num_duplicates <- sum(duplicated(dtafile))

# Print the number of duplicate rows
print(paste("Number of duplicate rows: ", num_duplicates))
```

```
> print(paste("Number of duplicate rows: ", num_duplicates))
[1] "Number of duplicate rows:  0"
```

4.5.1.4 Step 4: Data Extraction

Selection of relevant variables

We select the relevant variables that are important for achieving our objectives and meeting the criteria for the event log. To produce event logs required for process mining, a minimum set of variables is needed (Aalst, 2015), including the case ID, activity, and timestamp. In the current dataset, 15 out of 181 variables represent activities with recorded dates but no time of day, meaning process mining can be performed at a granularity of days.

The following is the list of the related activities:

- Symptom discovery or first abnormal screening test
- First healthcare seeking
- First medical consultation for symptom or abnormal screening test
- Last consultation in service 1

- First consultation in service 2
- Last consultation in service 2
- First consultation in service 3
- Last consultation in service 3
- First auxiliary diagnostic study
- Second diagnostic test
- Biopsy
- Cancer diagnosis confirmation (biopsy report)
- First consultation in cancer service
- Treatment start
- Surgery

4.5.1.5 Step 5: Data Cleaning

Handling missing values and data error

The data cleaning was a crucial part of the data analysis process that aimed to ensure the accuracy and reliability of the data being analysed. It involved several tasks such as handling missing values, handling outliers, and identifying and correcting errors.

Table 4.6 Issues relating to Mexican breast cancer dataset

Data issue category	Issue	Example	Solution Handling Error
Data inconsistencies	Typo errors	Start date for the ID M0423 that underwent surgery was marked with 6201	Mistakes were made while entering data into the system. For instance, the start date for the ID M0423 that underwent surgery was marked with 6201. The data owner was consulted to obtain the correct information, and the error was corrected with the accurate data.
	Cases with no symptoms	92 cases don't start with the	Process experts believe there was an error during the data entry process, and as a result, each case

	discovery data recorded	discovery of symptoms	needs to be investigated by process specialists.
	Missing Biopsy Data	327 cases with no biopsy despite a consultation date in cancer services	Process specialists have confirmed that each patient needs to undergo a biopsy to confirm the presence of cancer, and therefore, this information is not true. They found out that some patient biopsy data has been accidentally excluded from the dataset provided to the researchers.
	Treatment Discrepancies	Only 562 cases ended in treatment, despite all 688 cases undergoing treatment	Process experts have confirmed that this information is incorrect, and they have investigated and provided the correct information.
Data formatting	Same date of event	Two or more events with the same date	Some cases had two or more events with the same date, which occurred because the data set did not record detailed timestamps. This issue was discovered during data analysis, and the events were arranged in the right sequences based on expert knowledge. In the future, it would be helpful to obtain more detailed date and time information from electronic medical record systems, as these are recorded by computers and are more accurate.
	Error when entering dates	Backward arrows in the process model	Errors occurred when reading and entering dates from questionnaires or medical records into the dataset, resulting in backward arrows in the process model. The data owner was consulted to obtain the correct dates and fix the errors.

Data validation	Confusion on N/A marked data	7302 events marked with N/A in the dataset	7302 events were marked with N/A in the dataset, and investigators were unsure whether the event did not occur or did not have a date. The data owner was consulted to clarify the meaning of N/A. It was confirmed that an event marked N/A means that the event did not occur, and this information was used in the analysis. Obtaining direct input from the data owner or specialist is critical to ensuring accurate and correct information.

Verifying the event definitions with process expert.

Some of the variable meanings were unclear among the list of events, and there may be discrepancies compared to the actual definitions. To address this issue, we consulted with process experts to obtain accurate definitions and understanding.

For instance, there was initially confusion about the definitions of "first healthcare seeking" and "first medical consultation for symptoms." Although these two events may seem similar, they are different. "First healthcare seeking" refers to the patient's attempt to get a first consultation, whereas "first medical consultation for symptoms" is when the patient meets with a physician/doctor for the first time.

Similarly, the term “First Auxiliary Diagnostic Study” was initially unclear due to the presence of multiple diagnostic procedures in breast cancer care. After consulting with process experts, it was confirmed that the “First Auxiliary Diagnostic” refers to the diagnostic mammogram performed following clinical suspicion or an abnormal screening result, while the “Second Diagnostic Test” refers to the ultrasound used for further investigation. Mexico does not have a routine, population-wide breast cancer screening programme, most mammograms are performed opportunistically. Therefore, the term “abnormal screening test” in the dataset typically refers to an initial mammogram or

clinical breast examination that raised suspicion, and the “First Auxiliary Diagnostic” is usually the patient’s first imaging test, not a second mammogram.

Lastly, the definitions of "1st consultation in service 1," "1st consultation in service 2," and "1st consultation in service 3" were also unclear. The process expert explained that it refers to the number of first medical consultations received from different service providers [1: 6], where 1-Secretaría de Salud, 2-IMSS, 3-ISSSTE, 4-Servicio Privado, 5-Consultorio de Farmacia, and 6-Another. Although patients may have more than three first medical consultations, only a maximum of three consultations are recorded.

Renaming events and consolidating multiple events.

During data analysis, it was observed that several events were named differently or inconsistently across the dataset. This inconsistency created confusion and made it difficult to identify related events. To resolve this issue, we standardized event names to ensure consistency, improving the accuracy of event identification and analysis.

For example, the events "date of first healthcare seeking" and "date of first medical consultation" were consolidated under the new name "start consultation" to streamline the representation of initial healthcare engagement.

Furthermore, the start consultation date is determined by selecting the earliest date among the given events. Specifically, it takes the minimum date from the following: date of first medical consultation, first consultation service 2, and first consultation service 3. This means that whichever of these occurs first is assigned as the start consultation date.

Start consultation = min {date of first medical consultation, first consultation service 2, first consultation service 3}

The end consultation date is determined by selecting the latest date among the recorded consultation services. Specifically, it is set to the maximum date from the consultation events to ensure a complete representation of the consultation period.

End consultation = max {last consultation service 1, last consultation service 2, last consultation service 3}

4.5.1.6 Step 6: Event Log

This step involved generating the event log from the processed dataset. The event log is a structured file that stores the key elements required for process mining, including Case ID, activities, and timestamps. In some cases, additional information such as resources may be included, but these were not available for this study. Once the data was correctly formatted, the event log was ready to be loaded into the process mining software for analysis.

4.5.2 Phase 2: Process mining

The objective of this phase is to discover and refine an as-is process model from the event log and to derive parameter inputs for DES. We first construct and compare candidate discovery models, then apply objective-led abstraction, frequency- and time-based thresholding and clinical validation to obtain a clinically meaningful conceptual model. We also compute time-to-event and routing parameters for DES. A key comparison in this case study is between pathways and delays for patients whose initial consultation occurred in public versus private services. The final process models and parameters are integrated into the DES.

Conceptual Modelling

In this phase of the case study, process discovery was conducted to generate a process model using the event log assembled in the previous phase. Process discovery is a key step in process mining, where algorithms analyse the event log to extract an as-is process model that represents the actual flow of activities.

4.5.2.1 Step 1: Selection of process mining algorithm

The first step in process discovery is the selection of an appropriate algorithm to generate a meaningful process model from the event log. The choice of algorithm depends on factors such as the structure and complexity of the log (e.g., number of cases, activities,

and paths), the presence of noise, and the intended level of detail (simplified vs. highly detailed models).

For this case study, four algorithms were chosen because they are widely supported in process mining software, commonly applied in healthcare contexts, and offer complementary strengths:

- Alpha Miner – Works well for structured processes but struggles with noise and complex behaviour.
- Heuristic Miner – Handles noise better and identifies frequent patterns.
- Inductive Miner – Generates structured and hierarchical models, reducing complexity.
- Fuzzy Miner – Useful for exploring complex, unstructured processes by highlighting important paths.

Other techniques, such as genetic algorithms (Halawa et al., 2021), show promise in discovery tasks. However, their evaluation has largely focused on structured datasets, and they are less supported in standard process mining tools, often requiring custom implementation. Given the complexity of our healthcare data, their direct application was not applied in this thesis

The detailed strengths, weaknesses, and working mechanisms of these algorithms were introduced in Chapter 2, while their practical comparison using the Mexican dataset was conducted in Phase 2, Step 3 (comparison of process models).

4.5.2.2 Step 2: Generation of process models

We generated process models in Disco (Fuzzy Miner) and in ProM (Alpha Miner, Heuristic Miner, Inductive Miner, and Fuzzy Miner) using the event log assembled in Phase 1. Running Fuzzy Miner in both ProM and Disco allowed us to compare outputs across platforms. While the underlying logic was consistent, Disco was preferred for subsequent analysis because its visualisation was clearer and more user-friendly for communicating results to clinical collaborators.

The Fuzzy Miner model generated in Disco provided a visual representation of the breast cancer patient pathways, highlighting the sequence of activities, their dependencies, and the time taken for each activity to complete. The resulting process model revealed nine key activities: Symptom discovery, Start consultation, Mammogram, Ultrasound, Biopsy, End consultation, Consultation in cancer centre, Biopsy report, and Treatment start.

The most common pathway was: Symptom discovery → Start consultation → Mammogram → Ultrasound → Biopsy → End consultation → Consultation in cancer centre → Biopsy report → Treatment start.

However, analysis also revealed a high number of sequence variations, 153 in total demonstrating the complexity and diversity of patient diagnostic and treatment journeys. These variations were characterised by differences in the order of diagnostic procedures (e.g., mammogram before ultrasound versus ultrasound before biopsy) as well as in the types and timing of treatments administered. For example, one sequence involved mammogram → biopsy → treatment, while another involved ultrasound → biopsy → treatment.

Overall, the initial process models provided a comprehensive view of the patient pathways, highlighting both dominant flows and atypical variations. Figure 4.3 shows the initial process map, which, while informative, remains complex and therefore required further abstraction and analysis to focus on pathways relevant to diagnostic delays. The next section (Step 2c) provides a detailed comparison of the models generated by different algorithms.

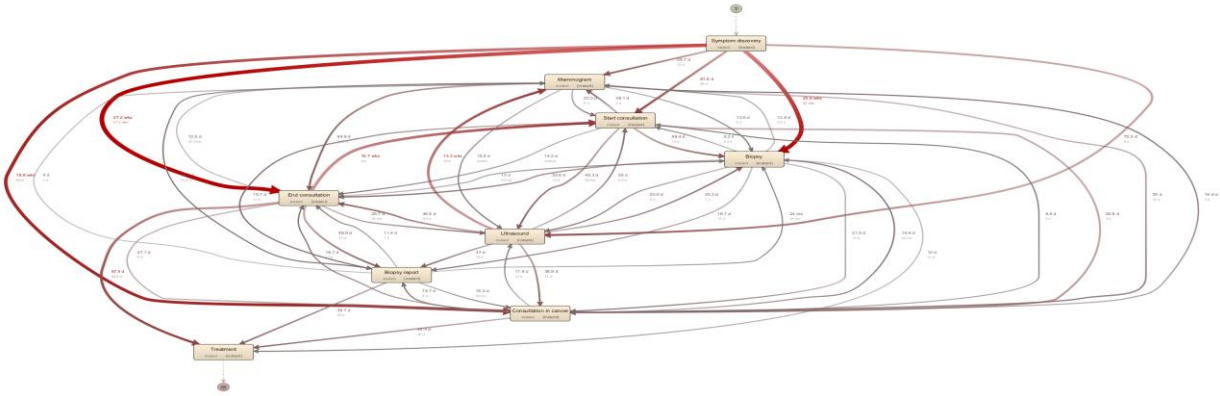


Figure 4.3: Initial process model from symptoms discovery to treatment start.

4.5.2.3 Step 3: Compare process models from different algorithms

Alpha Miner

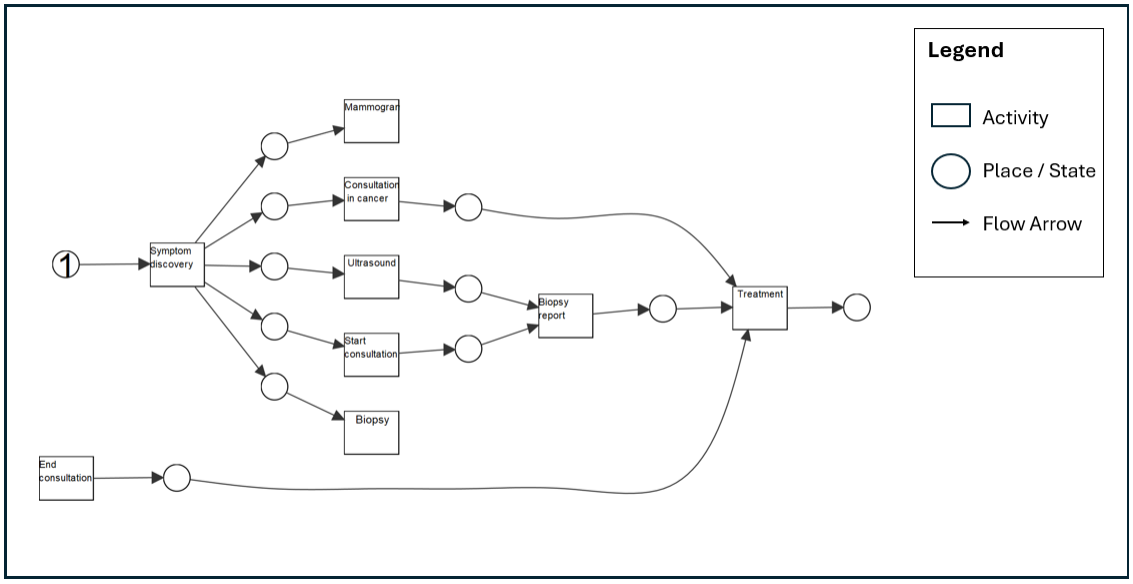


Figure 4.4: Process model generated by the Alpha Miner algorithm (ProM Software).

Figure 4.4 presents the process model discovered using the Alpha Miner algorithm in ProM, following a Petri net representation.

Notation Explanation

1. **Activity (Rectangles):**

   Represent activities extracted from the event log, such as *Symptom Discovery*, *Mammogram*, and *Biopsy*.

2. **Place / State (Circles):**

   Represent a waiting point in the process. They hold tokens that show when an activity can start or has finished, helping to control the flow between steps in a Petri net.

3. **Flow Arrow:**

   Indicates the direction of flow, showing the causal relationship between places and activities.

1. **Ordering Limitations in Diagnostic Sequences**

The model shows multiple parallel flows emerging from “Symptom Discovery,” including “Mammogram,” “Ultrasound,” “Consultation in cancer,” and “Biopsy.” This reflects Alpha Miner’s core logic: when it detects multiple activities that follow the same activity in varying orders across traces without a consistent sequence it assumes these are executed in parallel. However, analysis of the event log revealed that these diagnostic procedures often followed a structured sequence in practice, such as a Mammogram followed by an Ultrasound, and subsequently a Biopsy if warranted. Alpha Miner’s generalisation of these steps as concurrent fails to preserve the actual ordering observed in the data, resulting in a model that oversimplifies and misrepresents the diagnostic workflow.

2. **Inability to Represent OR Relationships**

After "Symptom Discovery," the care pathway varies based on clinical judgment and patient-specific needs. Some patients undergo only a "Mammogram," others receive both a "Mammogram" and an "Ultrasound," and some proceed through all diagnostic steps including a "Biopsy." This kind of variation reflects an OR relationship where different combinations of steps are possible depending on the case. However, Alpha Miner cannot explicitly model OR logic. Instead, it classifies such variability as either AND (parallel execution) or XOR (exclusive choice), depending on how the traces appear in the event

log. This binary interpretation forces the model to oversimplify the process either by displaying all diagnostic steps as occurring together (AND) or selecting only one possible path (XOR). Neither accurately represents the clinical decision-making flexibility inherent in real diagnostic workflows.

3. **Simplified Loop Handling**.

The event log shows that diagnostic steps such as “Ultrasound” and “Biopsy” are sometimes revisited, indicating the presence of short loops in real patient flows. However, Alpha Miner (Figure 4.4) fails to capture these repetitions, instead producing a simplified model where each activity occurs only once in a linear or parallel structure. This limitation affects the model’s ability to represent iterative diagnostic behaviours, which are critical in healthcare scenarios involving uncertain or evolving diagnoses.

4. **Lack of Non-Local Dependency Representation**.

In the Alpha Miner model (Figure 4.4), "Biopsy report" directly precedes "Treatment," reflecting only a direct succession relationship. However, real healthcare workflows often involve decisions that depend on earlier events, even when other steps occur in between, these are known as non-local dependencies. Alpha Miner is unable to capture such indirect relationships and only connects activities that immediately follow each other.

In the event log, ‘Biopsy’ and ‘Biopsy report’ are distinct events: ‘Biopsy’ represents the diagnostic procedure, while ‘Biopsy report’ records the reporting of results. Similarly, ‘Start consultation’ refers to the patient’s initial hospital encounter, whereas ‘Consultation in cancer’ refers to a specialist oncology consultation. These were therefore retained as separate activities and not merged during data cleaning.

As a result, essential clinical constraints such as requiring a "Biopsy report" before initiating "Treatment” care not explicitly represented if they are not part of a direct sequence. Additionally, while the model shows diagnostic and consultation steps in structured parallel branches, it lacks the conditional flexibility, iteration, and indirect routing that characterise real-world processes.

In practice, ‘End consultation’ was often followed by ‘Consultation in cancer,’ ‘Biopsy report,’ and eventually ‘Treatment start.’ However, in the Alpha Miner model, ‘End consultation’ appears unlinked because the algorithm could not detect a consistent direct successor across all traces. This reflects Alpha Miner’s limitation in oversimplifying variable but clinically meaningful paths, leaving the activity isolated in the discovered model.

To accurately reflect these complex dependencies and workflow variations, more advanced algorithms such as Heuristic Miner, Fuzzy Miner, or Inductive Miner should be used, as they are better equipped to handle parallelism, OR-relations, and complexity in healthcare pathways.

Heuristic miner

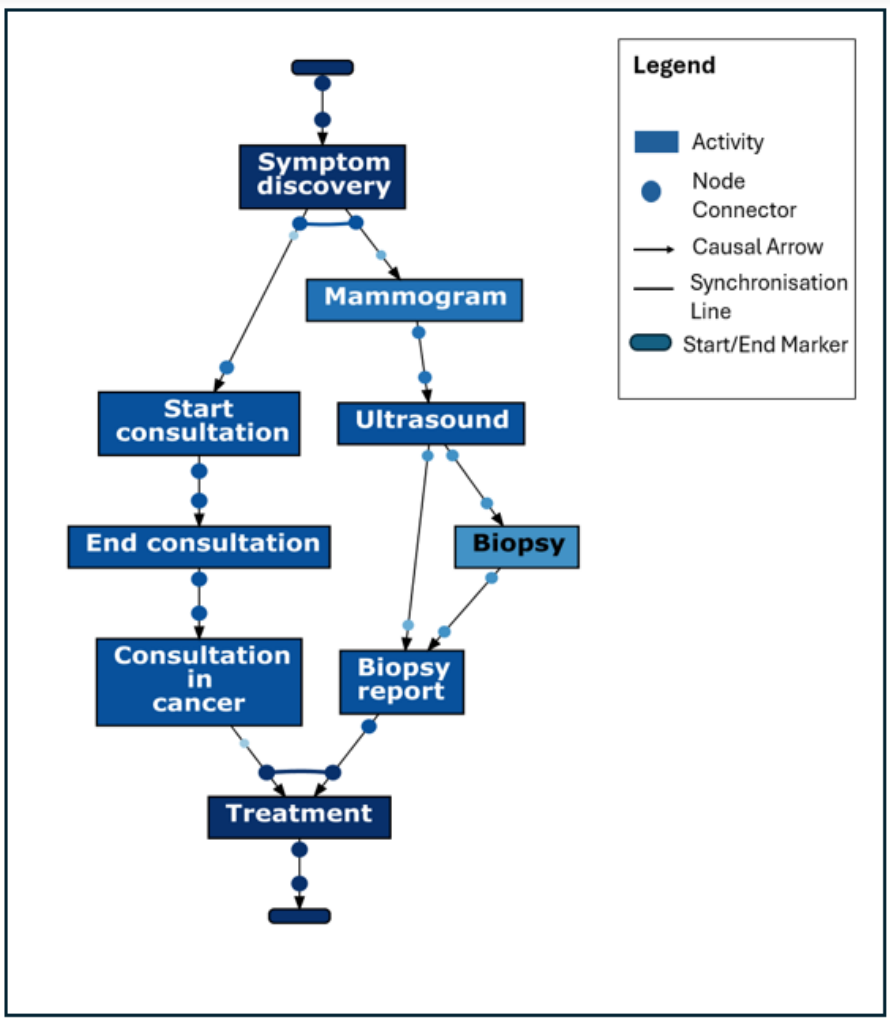


Figure 4.5: Process model generated by the Heuristic Miner algorithm (ProM Software).

Figure 4.5 displays the process model generated using the Heuristics Miner algorithm via the ProM software. This model captures observed pathways in the event log by relying on frequency-based heuristics, allowing for some tolerance to noise and minor variations in the log.

Notation Explanation

1. **Activity (blue rectangles):**
   These represent events or actions recorded in the event log (e.g., “Start consultation”, “Biopsy”, “Treatment”).

2. **Node Connector (blue circles):**
   These small, coloured dots indicate branching or joining points in the process flow. They serve as visual connectors between activities, allowing the process to split into parallel paths or merge multiple flows. While they are not activities themselves, they play a key role in routing and visualising the logical structure of the model.

3. **Causal Arrow (solid black arrows):**
   These arrows indicate the observed direction of flow between activities, reflecting direct succession based on event log traces.

4. **Start/End Marker (black ovals):**
   These indicate the starting and ending points of the process model.

The Heuristic Miner effectively captures the parallel structure between Mammogram and Ultrasound and the overall sequential flow of activities in the breast cancer diagnostic and treatment pathway. However, the model does not reflect repeated diagnostic steps

and filters out infrequent variations, which makes the process appear more linear than it is in the event log. Additionally, although Mammogram and Ultrasound are shown in parallel, the model does not explicitly represent alternative sequences (e.g., Ultrasound occurring before Mammogram), which limits its ability to fully capture the complexity and variability of clinical workflows.

1. Parallel Activities

The Heuristic Miner captures parallelism between activities using Node Connectors, shown in the model as small blue circles. One clear example of parallelism occurs after Symptom Discovery. In Figure 4.5, the model shows that some patients underwent a Mammogram while others initiated a Start Consultation, with both activities occurring in parallel. This structure reflects actual behaviour observed in the event log, where imaging and consultation steps were performed independently and not in a fixed order. Diagnostic imaging and clinical assessment therefore proceeded concurrently, depending on the patient's pathway.

2. Loops and Short Loops

Although the Heuristic Miner algorithm is capable of capturing loops, including short loops involving repeated activities, Figure 4.5 presents a simplified view where such loops are not represented. All transitions flow forward, and there are no backward arrows or cycle-forming connections that would indicate repeated sequences such as:

- *Consultation in cancer → Biopsy → Consultation in cancer*, or
- *Ultrasound → Biopsy → Ultrasound*.

However, the original event log reveals clear evidence of looping behaviour. These include both full loops and short diagnostic iterations, where patients revisit consultations or repeat imaging and biopsy procedures before progressing to treatment. These patterns, while clinically meaningful, occur less frequently and were likely filtered out by the algorithm's dependency threshold settings. Thus, the absence of visible loops in the Heuristic Miner output is not a weakness of the algorithm, but a result of its focus on dominant behaviour patterns. This

abstraction simplifies the process model, making it more readable but also more linear than the underlying clinical reality.

3. OR Relationships

One limitation of the standard Heuristic Miner is its inability to explicitly model OR splits and joins. OR relationships refer to situations where one or more—but not necessarily all—activities follow a given event. In Figure 4.5, both Mammogram and Ultrasound are shown as parallel branches after Symptom Discovery, but it is not clear whether they always co-occur or happen selectively. In reality, the event log includes cases where only one of these diagnostics was performed. However, this nuance is not captured in the current model, which may give the impression that both tests are routinely conducted together, even though that is not supported by all patient cases.

4. Handling of Rare or Exceptional Paths

Due to the frequency and dependency threshold settings used by the Heuristic Miner, rare or exceptional paths are filtered out, resulting in a model that highlights only the most common pathways. For example, sequences such as skipping Ultrasound and proceeding directly from Mammogram to Biopsy are not shown in Figure 4.5, even though such cases are present in the event log. This abstraction improves model clarity and reduces visual complexity, but it also excludes clinically relevant variations that could be important for analysis or simulation purposes.

Fuzzy Miner

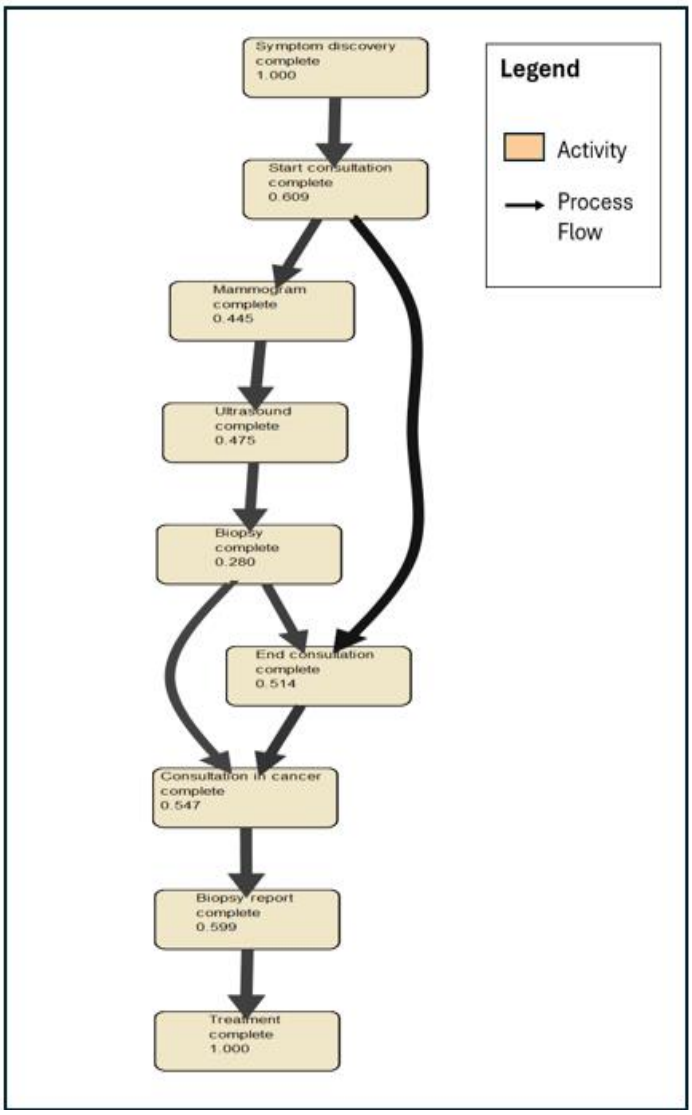


Figure 4.6: Process model generated by the Fuzzy Miner algorithm(ProM software).

Figure 4.6 shows a process model generated using the Fuzzy Miner algorithm in process mining. It follows a Directly-Follows Graph (DFG) representation.

Notation Explanation

1. **Activity Boxes:**

   Rectangular nodes represent process activities (e.g., *Biopsy complete, Treatment complete*). Each label includes the word "complete", indicating the lifecycle stage logged. The number beneath each activity (e.g., 0.280) is the significance score,

which reflects the frequency of that activity in the event log. This score ranges from 0 to 1:

- 1.000 means the activity appears in all cases.
- Lower values reflect less frequent activities in the event log.

2. **Process Flow:**

   Arrows represent transitions between activities. Arrow thickness reflects the frequency or strength of the connection:

   - **Thicker arrows** = more frequent transitions.
   - **Thinner arrows** = less common or weaker transitions.

Fuzzy Miner works by applying abstraction techniques to emphasize the most significant process paths while filtering out less relevant details. The core capability of Fuzzy Miner lies in its ability to abstract paths based on relevance and significance metrics. It uses frequency-based and correlation-based measures to determine the strength of relationships between different process activities. The algorithm assigns a numerical weight to each activity and connection, allowing it to determine which parts of the process should be retained and which should be hidden. One of the unique strengths of Fuzzy Miner is its ability to deal with loops, concurrent activities, and deviations in a more natural way than rigid process discovery techniques.

1. Highlighting Strong Relationships

Fuzzy Miner has captured the diagnostic and treatment journey for the breast cancer patients, presenting a structured yet flexible process model. The thickness of the arrows in the diagram represents the strength of relationships between activities, indicating the most common pathways taken by patients. Each transition is weighted based on frequency and significance, allowing for a clearer understanding of how patients move through the diagnostic and treatment process.

For example, the pathway from “Symptom discovery” to “Start consultation” is thick, reflecting a strong and frequently followed sequence. This suggests that most of patients who experience symptom discovery proceed to a consultation. Similarly, Fuzzy Miner highlights key diagnostic steps that patients undergo, such as mammograms, ultrasounds,

biopsies, consultations and treatment, each with its respective weight, ensuring that critical transitions are preserved

2. Tuneable abstraction

Fuzzy Miner provides tunable abstraction, allowing users to control the level of detail in the discovered process model. One of the key features of tunable abstraction is the ability to balance between detail and simplicity. Users can set thresholds for significance and correlation metrics, determining which activities and transitions should be retained or hidden. For example, in a breast cancer diagnostic process, highly frequent activities like "Symptom discovery" and "Treatment" are always retained, while less significant activities, such as rare alternative diagnostic steps, can be filtered out to avoid unnecessary complexity.

Fuzzy Miner's tunable abstraction not only highlights common and high-frequency pathways but also helps identify rare or infrequent activities, which can be crucial for understanding process variations, exceptions, and potential inefficiencies. By adjusting abstraction settings, users can control the level of detail, ensuring that both dominant patterns and rare deviations are visible based on analytical needs.

One key benefit of detecting rare activities is the ability to identify outliers or uncommon patient pathways, which may indicate potential issues or special cases. For example, in the breast cancer diagnostic process model, most patients follow the common pathway from "Start consultation" → "Mammogram "→ "Ultrasound" before a biopsy. However, there are alternative paths where some patients skip certain steps or undergo a biopsy directly without an ultrasound.

Inductive Miner

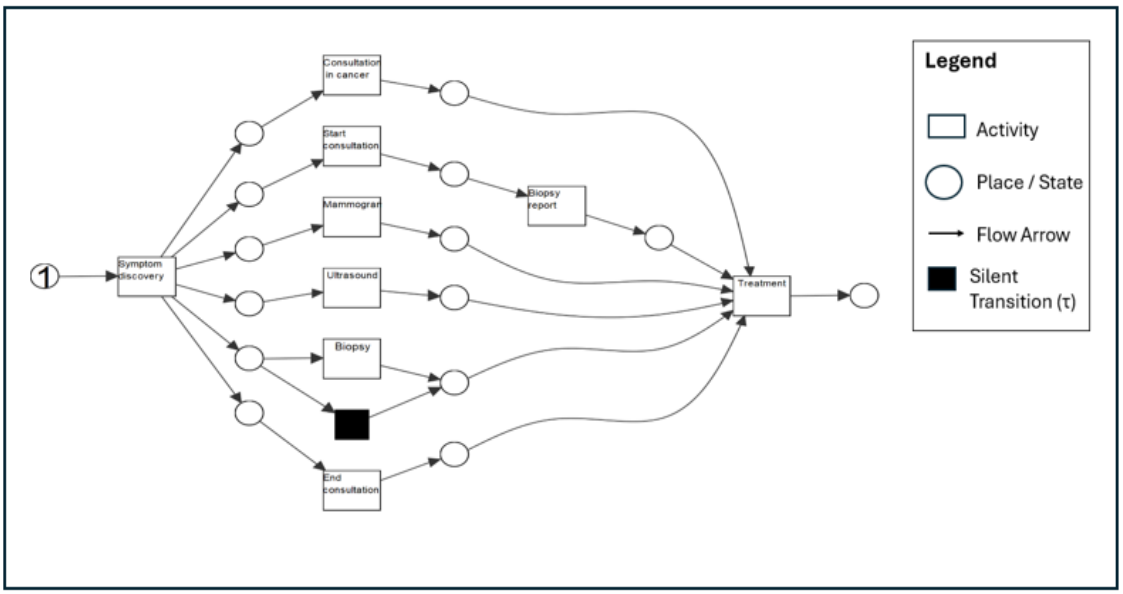


Figure 4.7: Process model generated by the Inductive Miner algorithm (ProM Software).

Notation Explanation

1. **Activity Boxes:**
   Rectangular boxes represent healthcare activities such as "Symptom discovery," "Ultrasound," or "Biopsy."
2. **Place/State (Circles):**
3. Represent a waiting point in the process. They hold tokens that show when an activity can start or has finished, helping to control the flow between steps in a Petri net.
4. **Flow Arrows:**
   Represent the control flow or logical sequence of events from one activity to another.
5. **Black Square (Silent Transition):**
   Denotes a silent transition, an artificial step not recorded in the event log, used by Inductive Miner to merge paths and preserve a structured model.

The process map in Figure 4.7 was generated using the Inductive Miner algorithm in ProM software. This algorithm follows the Petri net representation. From our observation, the Inductive Miner effectively captured a structured process model by recursively decomposing the event log into smaller, well-defined patterns. The resulting model illustrates clear parallel branches and controlled flow between activities, consistent with Petri net representation. However, a known limitation of the Inductive Miner is its difficulty in handling unstructured or highly variable processes. When multiple looping behaviours are present, the algorithm struggles to represent them accurately, often leading to oversimplifications or incorrect interpretations. We explain this with an example in the following section.

1. Structured Processes

The process model generated by the Inductive Miner (Figure 4.7) begins with "Symptom discovery" and branches into several diagnostic activities, including Mammogram, Ultrasound, Biopsy, Start consultation, End consultation, and Consultation in cancer. These activities are structured in parallel, indicating that different diagnostic procedures occur independently across patient cases before converging toward Treatment.

One notable issue in the model is that "Biopsy report" appears as an independent step contributing to the decision-making before "Treatment," but it is not directly connected to "Biopsy." In real clinical workflows, the biopsy report is a direct outcome of the biopsy procedure. The absence of this link introduces a semantic inconsistency in the model. This reflects a limitation of the Inductive Miner, which enforces block-structured models and can exclude important dependencies when they are not frequent or cleanly structured in the event log.

Despite this, the Inductive Miner effectively captures structured behaviour by organising activities into well-defined parallel and sequential components. It excels in discovering clear process logic, free from excessive variation or loops. However, it also oversimplifies complex workflows by omitting less frequent but clinically important transitions resulting in a model that prioritises structure and clarity at the expense of completeness.

2. Limited Handling of Unstructured Processes

In the real event log, patients often undergo repeated diagnostic procedures such as additional imaging after inconclusive results—or revisit consultations before progressing to a biopsy. These looping and rework behaviours are not represented in the Inductive Miner model (Figure 4.8), which displays a clean, linear flow. Activities such as Start consultation, Mammogram, and Ultrasound appear only once and in fixed positions, with no indication of repetition or feedback.

The Inductive Miner algorithm enforces block-structured constraints and applies filtering that removes low-frequency or non-conforming patterns. As a result, less frequent but clinically relevant paths such as patients skipping certain steps or undergoing out-of-sequence diagnostics are excluded from the visualisation. This produces a model that appears more structured than the actual process and fails to capture the variability and complexity found in real-world healthcare workflows.

3. Artificial Transitions (Silent Transitions)

In the Inductive Miner-generated model (Figure 4.8), the black square denotes a silent transition (τ-transition) an artificial step does not present in the event log but introduced by the algorithm to enforce a block-structured, sound process model. These transitions are used to merge parallel branches, synchronize optional sequences, and maintain logical consistency when real-world behaviour deviates from strict structural patterns.

In this case, patient diagnostic journeys include multiple paths such as mammogram, ultrasound, biopsy, and consultations that do not always occur in the same order or combination. Some patients proceed directly to treatment after biopsy, while others undergo end consultations beforehand. To merge these different sequences into a unified flow, Inductive Miner inserts the silent transition before the treatment step.

This mechanism allows the model to accommodate variation in the process without becoming fragmented. However, by relying on silent transitions instead of explicitly modelling all behaviour, the algorithm simplifies complex pathways, filters out rework,

and removes low-frequency loops. As a result, the model appears more rigid and linear than the actual diagnostic workflows observed in the event log.

Discussion on the Comparison of Algorithms.

In the context of healthcare process mining, selecting an appropriate discovery algorithm is essential for accurately reflecting real-world patient pathways and treatment workflows. Each algorithm such as Alpha Miner, Heuristic Miner, Fuzzy Miner, and Inductive Miner offers different capabilities in handling noise, parallelism, loops, and complexity. Their respective strengths and limitations influence how well the resulting models align with the variability and structure found in clinical event logs.

The Alpha Miner algorithm is best suited for structured, well-defined healthcare workflows such as routine testing or standardized surgical procedures. It is based on Petri net theory and produces models with strict control-flow logic. However, as shown in Figure 4.5, Alpha Miner performs poorly with real-world healthcare event logs, which often contain noise, missing data, and process deviations due to individual patient circumstances or emergency interventions. The resulting model appears overly linear, with no backward arrows, no loops, and no parallel branches, even though the event log shows such behaviours. This makes Alpha Miner unsuitable for capturing concurrent diagnostic or treatment steps such as imaging and consultation which often occur in parallel in healthcare practice.

Heuristic Miner improves upon Alpha Miner by using frequency-based heuristics to handle noise and variability. It is especially suited for environments with dominant patterns but some variation, such as treatment routines or chronic disease management. As illustrated in Figure 4.6, Heuristic Miner successfully captures parallel activities—for example, the ability for Mammogram and Start Consultation to occur independently after Symptom Discovery. However, it fails to represent loops and short cycles. Despite actual event log data showing repeat sequences like Biopsy → Consultation in Cancer → Biopsy, these loops are not shown in the model due to the algorithm's filtering thresholds.

The lack of backward arrows or cycle-forming transitions in Figure 4.6 suggests a more linear process than what actually occurred. Thus, while Heuristic Miner handles concurrency and noise better than Alpha Miner, it oversimplifies the process by filtering out infrequent but clinically relevant variations.

Fuzzy Miner is designed for unstructured and variable processes and applies abstraction techniques based on frequency (significance) and correlation strength. As shown in Figure 4.7, it filters out less significant behaviours while highlighting the most frequent and important process paths. The thickness of the arrows indicates relationship strength, and the numerical weights on nodes and edges help identify which activities and transitions are central to the process.

Fuzzy Miner does not enforce a rigid control-flow structure and is especially useful for exploring complex clinical processes such as cancer diagnostic pathways or emergency department workflows. However, this abstraction can come at the cost of precision. Rare but clinically meaningful transitions may be filtered out, and the curved layout may require interpretation. Despite this, its flexibility and tunable thresholds make it ideal for presenting simplified and detailed process views based on analytical needs. This adaptability is particularly useful when communicating with both clinicians and analysts.

Inductive Miner enforces a block-structured model and guarantees soundness, making it suitable for structured hospital workflows or administrative processes. It handles concurrency more effectively than Alpha Miner, and Figure 4.7 shows parallel branches for diagnostic activities such as Mammogram, Ultrasound, and Start Consultation. However, it struggles with unstructured behaviour and repeated loops. For example, Biopsy Report is not directly connected to Biopsy, even though this dependency exists logically and clinically.

To maintain a structured model, Inductive Miner introduces silent transitions (black squares in Figure 4.7), which help synchronise different branches but do not correspond to any real event. While this maintains logical consistency, it abstracts away rework and optional steps, potentially misrepresenting the clinical reality. When patient journeys are

irregular, such as skipping steps or repeating tests, Inductive Miner tends to filter these out, leading to oversimplified models.

This study employed Fuzzy Miner because it preserves the structure of the event log without enforcing artificial constructs like silent transitions or strict block structures. Rather than imposing predefined control-flow logic, it observes the frequency and strength of activity connections. Its use of significance (how often an activity appears) and correlation (the strength of activity-to-activity transitions) provides a flexible, data-driven way to analyse highly variable healthcare processes.

Fuzzy Miner also offers thresholding features that allow for tunable abstraction enabling both high-level visualisations and detailed exploration of rare patient pathways. This made it especially effective in this case for presenting multiple views of the breast cancer diagnostic process to clinicians, highlighting typical patient journeys while also enabling the identification of deviations that may require further investigation.

Although process mining algorithms may not capture every clinically important pathway, their outputs added significant value compared with relying on clinical input alone. Process mining grounded the conceptual model in empirical evidence, quantified the frequency of pathways, and highlighted variations and delays in actual patient journeys. Clinical input was then used to identify and retain important low-frequency pathways that the algorithms might miss. The integration of both approaches ensured that the DES model was based on real-world data while remaining clinically valid.

4.5.2.4 Step 4: Data abstraction



In this section, we describe the process of simplifying complex process maps, often referred to as 'spaghetti diagrams,' to create more manageable and actionable representations for discrete event simulation (DES) modelling. Spaghetti diagrams depict detailed, unfiltered event flows, where high levels of complexity can obscure critical insights and complicate the modelling process. The aim of reducing these diagrams is to retain essential pathways that reflect the system's core behaviours while excluding less relevant or infrequent paths, ultimately creating a streamlined process model suited for DES.

The Process of Reducing 'Spaghetti Diagrams'

To achieve an effective reduction, a sequential three-step process was applied: impact on objectives, frequency of occurrence, and expert input. This ensured that filtering was not based on frequency alone.

1. **Impact on Objectives**: To better align the process model with the simulation's objectives, we analysed each pathway's relevance to the outcomes. This evaluation allowed us to focus on pathways that significantly affected the outcome.

2. **Frequency of Occurrence**: After objective-based filtering, frequency analysis was applied to highlight how common or rare different pathways were. Very rare or unusual pathways were not excluded automatically but were flagged for further review. This helped ensure that the final model did not become unnecessarily complex while still retaining potentially important variations.

3. **Expert Input**: Process simplification also relied on expert input to validate decisions regarding pathway inclusion or exclusion. Experts (i.e., the clinician familiar with the operational context of the event log data) provided insights into

pathways that might hold relevance despite low frequency or perceived minimal impact. Their confirmation represented the final step, ensuring that rare but clinically meaningful pathways were retained while exceptional cases could be excluded.

4.5.2.5 Impact on Objectives

The relationship between data abstraction and the importance of objectives in modelling, particularly in discrete event simulation (DES), lies in how abstraction helps achieve these objectives effectively. Data abstraction transforms raw, detailed data into manageable, meaningful components, aligning the model with its specific goals. Without a clear connection between the abstraction process and the model's objectives, the simulation risks becoming either too simplistic or overly complex, reducing its usefulness.

The relationship between data abstraction and the objectives of modelling, particularly in discrete event simulation (DES), has been highlighted in prior research. Abstraction enables complex event data to be simplified into meaningful representations that align with the intended purpose of the model (Rozinat & van der Aalst, 2009; van der Aalst, 2016; Mans et al., 2015). Data abstraction transforms raw, detailed data into manageable components so that the model reflects the chosen objectives rather than the full complexity of the underlying process (Rozinat & van der Aalst, 2009; van der Aalst, 2016). Without such alignment, the resulting simulation risks becoming either too simplistic or overly complex, limiting its usefulness (Rojas et al., 2016).

In the context of the Mexican breast cancer case study, the conceptual model for data abstraction is guided by the objective of identifying delays from diagnosis to the initiation of treatment. Breast cancer is a time-sensitive condition where early diagnosis and timely initiation of treatment are critical for improving the effectiveness of care. The goal of the study is to enable time-to-event modelling, focusing on analysing event sequences, intervals, and patterns in the patient pathways, rather than examining resource capacity.

In time-to-event modelling, data abstraction emphasizes identifying key events and their timing, extracting meaningful sequences, and understanding transitions between states

(e.g., diagnosis to treatment). This abstraction filters out irrelevant details, ensuring the model captures the temporal dynamics critical for assessing delays and their impact on outcomes (Rozinat & van der Aalst, 2009; van der Aalst, 2016; Rojas et al., 2016).

Outliers in time-to-event modelling, such as extremely long or short delays, are carefully considered because they may indicate systemic inefficiencies, unusual patient circumstances, or data entry errors. These outliers can significantly affect the model's findings and may require contextual interpretation to determine whether to include, exclude them. For example, an extremely long delay might reveal a critical gap in the care pathway that needs attention (Mans et al., 2015; Rojas et al., 2016).

In capacity modelling, data abstraction targets parameters like resource availability, workload distribution, and bottlenecks. Outliers in this context often represent exceptional resource demands or anomalies in system usage. These outliers may skew performance metrics or suggest resource inefficiencies, requiring analysis to decide whether they reflect rare events or a pattern that warrants systemic change. Unlike time-to-event modelling, where outliers might highlight patient-specific issues, outliers in capacity modelling often point to operational challenges (Partington et al., 2015; Mans et al., 2015).

The data abstraction-based impact of objectives is carried out in the following:

1. **Initial Filtering of variation**: Initial filtering of variation involved excluding pathways that were not relevant to the study objectives of analysing diagnostic delays. For example, some pathways ended after only a first consultation, with no subsequent diagnostic or treatment step. These were excluded as they did not provide information about diagnostic progression or delays. Pathways that continued through the diagnostic and treatment sequence were retained. After this objective-based filtering, frequency analysis was then applied in the next step to highlight very rare or infrequent pathways. These were not excluded automatically but were flagged for further review, with final decisions confirmed through clinical expert input.

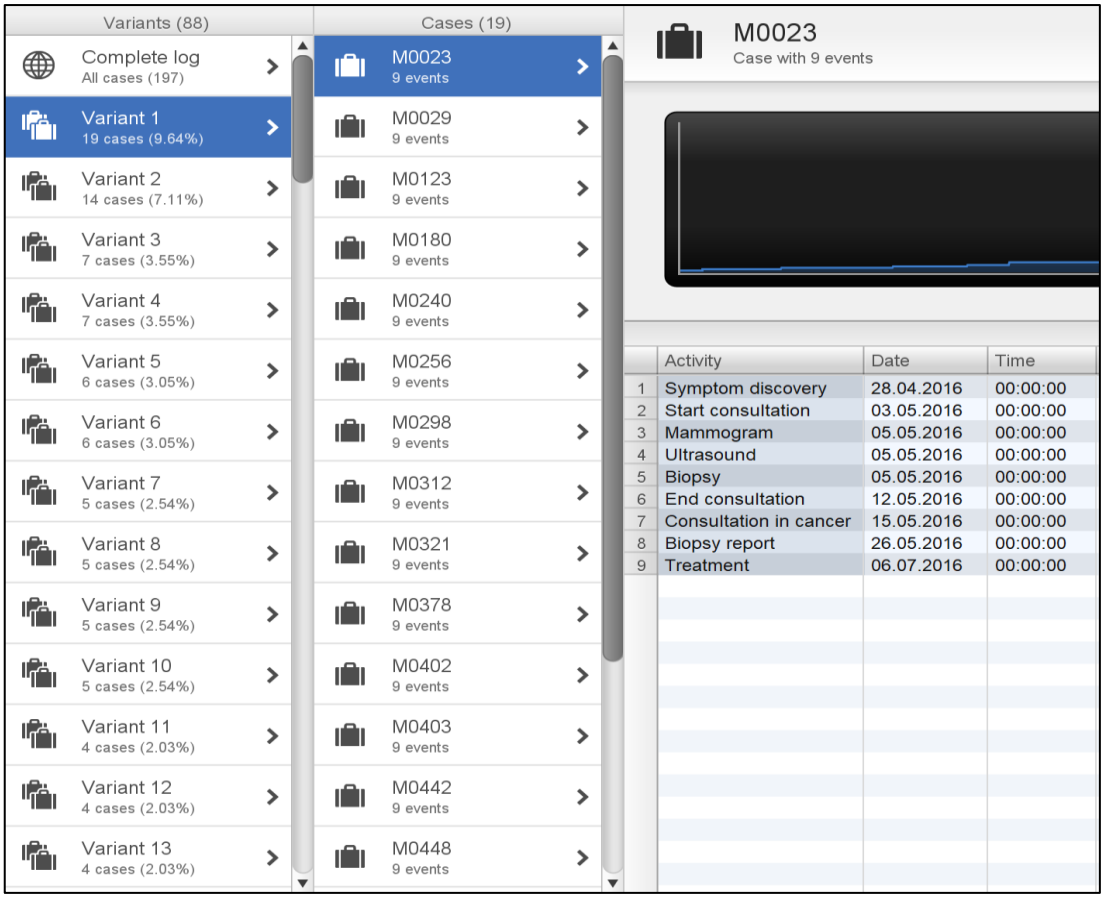


Figure 4.8: Example of Variation in the Mexican Breast Cancer Case Study.

2. **Consolidation of Events**: The consolidation of events at the abstraction stage differs from the earlier consolidation carried out during data cleaning. During data cleaning, the focus was on improving data quality by correcting inconsistencies in the event log. At this stage, event labelling was standardised to ensure clarity and consistency. At the abstraction stage, however, consolidation was guided by clinical expert input to achieve the modelling objectives. The aim was not to correct inconsistencies but to ensure the model reflected the right level of detail for analysing diagnostic delays. Events were therefore consolidated in a way that adjusted the level of granularity, allowing the model to emphasise the critical transitions in the diagnostic pathway and retain only the detail relevant for analysing delays.

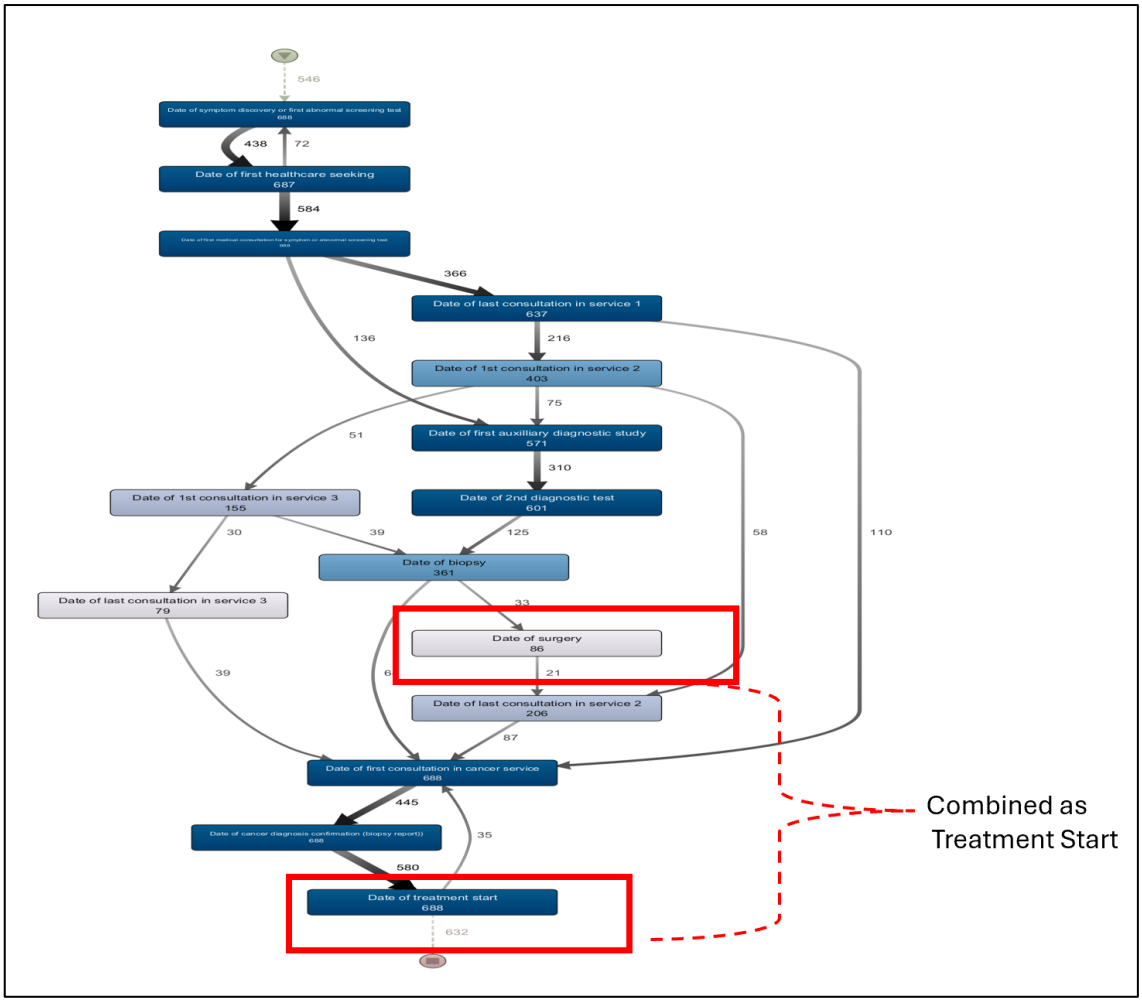


Figure 4.9: Example of event consolidation at the abstraction stage

4.5.2.6 Frequency of Occurrence

Simplification of process models plays a crucial role in managing the inherent complexity of real-world event logs. After objective-based filtering, frequency analysis was applied as the next step to highlight how common or rare different pathways were. Common pathways, which represent frequent behaviours, were typically prioritised, while very rare pathways were flagged for further consideration. Importantly, low-frequency pathways were not excluded automatically. Instead, they were reviewed again and final decisions were confirmed with clinical expert input. This approach ensured that rare but clinically meaningful trajectories were preserved.

The Fuzzy Miner algorithm was particularly effective in supporting pathway thresholding for data abstraction. Initially, Fuzzy Miner generated a process model that captured all pathways present in the event log. It then allowed thresholding to be applied, which highlights low-frequency behaviours and helps adjust the level of abstraction, rather than automatically removing them. Threshold values were defined with guidance from domain experts, not as rigid cut-offs. For example, in the Mexican case study, setting a 10% threshold provided a balanced abstraction that reduced complexity while still retaining pathways considered relevant for analysing diagnostic delays.

Applying thresholding at different levels produced models with varying degrees of abstraction. At a high level of abstraction (using a higher threshold value), the model focused on the most frequent pathways, offering a streamlined view of primary behaviours. At a moderate level, the model balanced detail with simplicity, retaining moderately frequent pathways to capture variations. At a low level (using little or no thresholding), all pathways were included, creating a highly detailed but complex process model. Each level served a different purpose: high-level models supported DES, while lower-level models were more suitable for detailed audits or investigating exceptions. Threshold values were initially proposed by the modeller in the research team and then refined through consultation with clinical experts to ensure that clinically meaningful but less frequent pathways were not excluded. The modeller suggested an initial threshold, while domain experts provided judgement on whether the resulting level of abstraction remained clinically valid.

The Mexican case study exemplified this approach. Frequency analysis, pathway thresholding, and expert validation were combined iteratively to balance simplification with the retention of clinically important but less common trajectories. This approach ensured that frequency was not the sole basis for exclusion; instead, it served to flag candidate pathways for expert review, allowing low-frequency but meaningful pathways to be retained where appropriate.

4.5.2.7 Analyse Frequency and Time Between Steps Using Threshold Based

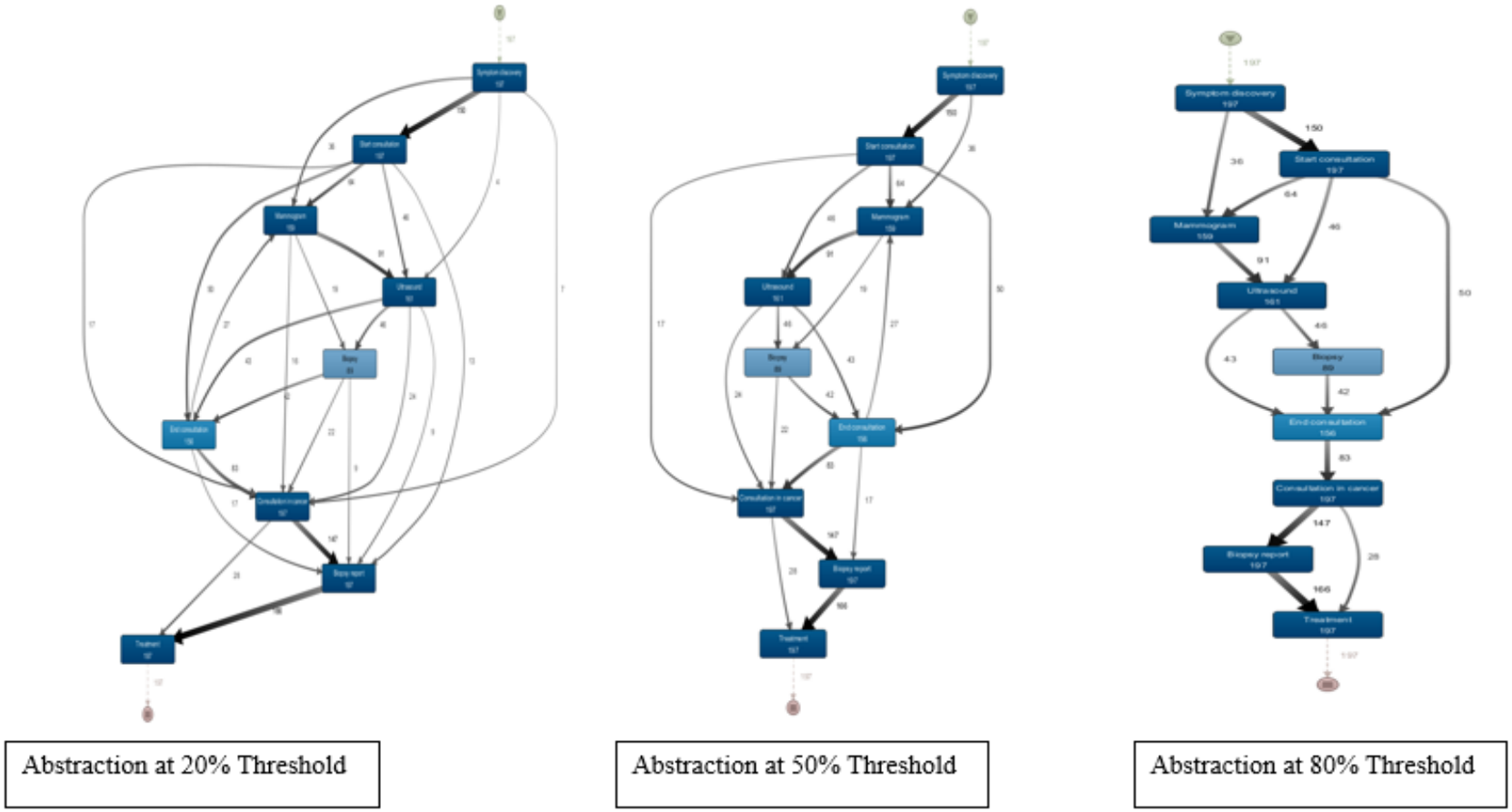


Figure 4.10: Example of Process Model Abstraction at 20%, 50%, and 80% (Frequency Thresholds).

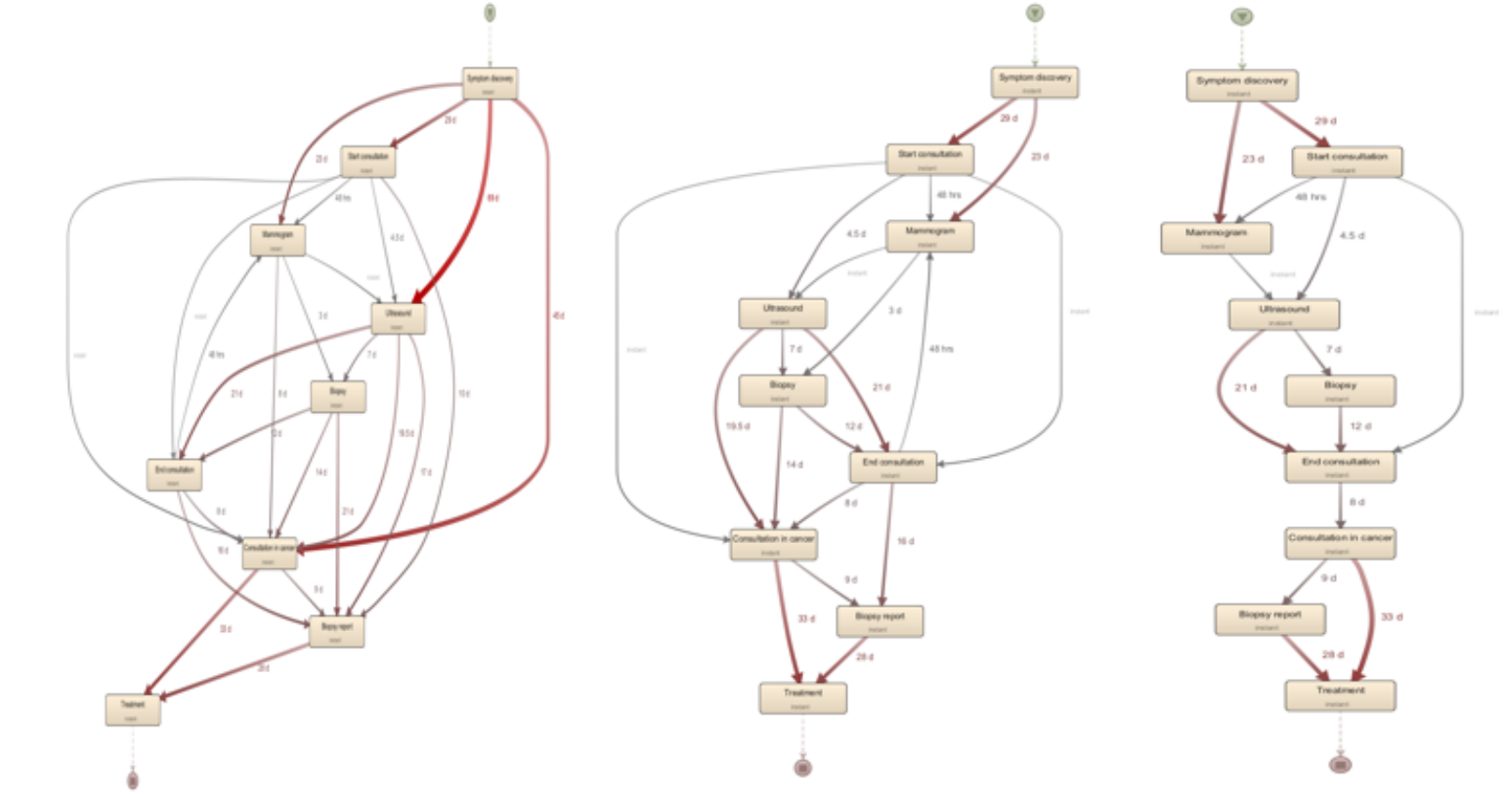


Figure 4.11: Example of Process Model Abstraction at 20%, 50%, and 80% (Time Thresholds).

The final model represents a clinically meaningful abstraction of the breast cancer diagnostic pathway and the initiation of treatment. After pathways were first reviewed against the study objectives, frequency and time-based thresholding were applied during the abstraction step to highlight transitions associated with long delays and to manage the level of detail in the process map. Pathways identified through this process were not excluded automatically but were flagged as candidates for further consideration in consultation with clinical experts (see Section 4.5.2.8). This approach reduced unnecessary complexity while preserving pathways relevant to analysing diagnostic differences between public and private healthcare services.

Symptom Discovery

Symptom discovery was the starting point for all patients (197 cases, 100%), marking the moment they became aware of symptoms and sought medical attention. The transition Symptom Discovery → Start Consultation (88.3 days) was retained as the dominant pathway for analysis and enabled comparison of consultation waiting times in public vs. private hospitals.

A small number of patients ended their consultation early without proceeding with diagnostic steps. The transition Symptom Discovery → End Consultation (~96 days delay) was flagged for expert review, as these cases reflected incomplete diagnostic journeys.

First Consultation

Following symptom discovery, all patients proceeded to consultation (197 cases, 100%). The transition Start Consultation → Mammogram (41.1 days) was retained as a key pathway in structured diagnosis.

Some patients had repeated consultations before imaging (Start Consultation → End Consultation → Start Consultation, 53.5 days delay). This loop, along with the direct path Start Consultation → Biopsy, was flagged as atypical and set aside for expert review.

Diagnostic Testing (Mammogram and Ultrasound)

The transitions Mammogram → Ultrasound (18.7 days) and Ultrasound → Biopsy (56.4 days delay) were retained as clinically appropriate steps in the diagnostic workflow.

Rare alternative sequences, such as Mammogram → Consultation in Cancer (skipping Biopsy), were flagged for expert review as non-standard.

Figure 4.11 also shows a frequent "consultation-only" pathway. This should not be interpreted as patients bypassing diagnostic tests. Rather, it reflects the Mexican healthcare context, where some patients completed diagnostic investigations in external facilities (e.g., private clinics or standalone imaging centres) before entering the cancer centre for consultation. Since the event log was derived from the hospital system records, these externally performed diagnostics were not captured, resulting in the appearance of consultation-only pathways as a common variant in the models.

Biopsy and Diagnosis Confirmation

Biopsy (89 cases, 45.18%) was an essential step for diagnosis. The transition Biopsy → End Consultation (20.6 days delay) was retained as an important step for assessing delays before specialist referral.

Consultation in Cancer and Biopsy Report

The transition Consultation in Cancer → Biopsy Report (20.8 days) was retained as a central pathway in the diagnostic workflow.

A rare transition Ultrasound → Biopsy Report (40.2 days delay, 4 cases) was flagged for expert review as an atypical shortcut.

Treatment Initiation

Treatment began after biopsy confirmation in most patients (166 cases, 84.26%). The transition Biopsy Report → Treatment (39.7 days) was retained as the main treatment initiation pathway. A small number of patients (2 cases, 1.02%) started treatment without biopsy confirmation (End Consultation → Treatment) was flagged for expert review as an exceptional case.

4.5.2.8 Review of Flagged Pathways with Expert

This step operationalises the validation activity from the framework: clinicians reviewed unusual or low-frequency pathways flagged during the frequency and time-based analysis (Section 4.5.2.7) to confirm inclusion or exclusion against clinical plausibility. These pathways were not excluded automatically; instead, clinical experts determined whether they should be retained in, or excluded from, the DES abstraction. This ensured that rare but meaningful clinical variations were retained, while exceptional or non-standard behaviours that did not reflect systematic diagnostic delays were removed.

Examples of Exclusions Based on Expert Review

1. Start Consultation → Biopsy (17 weeks, 6 cases)
   Flagged for bypassing imaging and showing an unusually long delay. Experts confirmed this was atypical and not representative of structured diagnostic practice. Excluded.

2. Mammogram → Consultation in Cancer (skipping Biopsy, 57.5 days, 5 cases)
   Flagged because it skipped biopsy confirmation. Experts confirmed this was a non-standard sequence. Excluded.

3. Ultrasound → Biopsy Report (40.2 days, 4 cases)
   Flagged as an atypical shortcut bypassing consultation. Experts confirmed this was exceptional and not representative. Excluded.

4. End Consultation → Treatment (skipping Biopsy Report, 2 cases)
   Flagged as treatment without biopsy confirmation. Experts confirmed this was a rare exceptional case and excluded.

Examples of Retained Pathways Based on Expert Review

1. Ultrasound → Biopsy (23.4% of cases): Though moderate in frequency, this path was emphasised by experts as a clinically appropriate progression when ultrasound reveals suspicious results. Retained.

2. Mammogram → Ultrasound (46.2% of cases): A high-frequency and common sequence that provides diagnostic detail before biopsy. Retained.

4.5.2.9 Step 5: Subgrouping analysis

The purpose of this phase is to compare patient pathways for different service providers and analyse wait times for these sub-groups. To facilitate subgrouping, the types of service providers were listed based on the presentation, including Secretaría de Salud, IMSS, ISSSTE, Servicio Privado, Consultorio de Farmacia, and Otro (others). Table 4.6 provides a brief overview of these sub-groups, allowing for a better understanding of the different types of service providers. However, for this particular analysis, we only compared the performance of two different service providers within these sub-groups. In this study, the process expert is interested in analysing the pattern of diagnostic time interval for two different types of service providers - public service- Secretaría de Salud and private services Servicio Privado.

Table 4.7 Brief information on types of subgroups based at presentation (Redacted)

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

The subgroup analysis compared public and private providers, a decision guided by clinical input since this distinction is central to understanding patient pathways in Mexico. Other providers had small case numbers and largely represented subsets of private care; therefore, the analysis focused on Secretaría de Salud (public, n=256) and Servicio Privado (private, n=321). This ensured that the comparison was both clinically meaningful and aligned with the study objectives, allowing examination of differences in patient pathways and evaluation of diagnostic delays between the two systems.

After analysing the pathways for the two subgroups, it was found that the main sequence of activities that emerged from both model processes was as expected, namely symptoms discovery > first medical consultation > last consultation in health services prior to cancer centre > consultation in cancer > biopsy > treatment. Diagnostic events such as

mammogram, ultrasound, and biopsy occur between the first and last medical consultations. Figure 4.12 in the next subsection shows the final process model for the public and private subgroups.

Moreover, an analysis of the delay time between events was conducted, with a particular focus on the diagnostic interval. The results, presented in Table 4.7, highlighted that the longest time difference between public and private healthcare services was observed between the events of ultrasound and biopsy, with an average of 33.3 days. Comparing the time intervals for public and private services, the study found that public services took a longer time for most diagnostic events compared to private services. These findings aligned with the expectations of the process expert, whose prior research had identified delays during the diagnostic interval as a significant factor contributing to breast cancer patients seeking treatment at a later stage. The results of the time delay analysis will be used in DES modelling to conduct scenario analyses aimed at estimating the potential impact of reducing delays on treatment initiation and patient outcomes.

Table 4.8 Time between events for Public and Private services

Interval	Events	Public Service	Private Service	Time different between Public and Private services
		Mean	Mean	Mean
Patient Interval	Symptom discovery > start consultation	88.3 d	65.9d	22.4d
Diagnostic Interval	Start consultation > mammogram	41.1d	30d	11.1d
	Mammogram> ultrasound	18.7d	12.2d	6.5d
	Ultrasound> biopsy	56.4d	23.1d	33.3d
	Biopsy> end consultation	20.6d	18.6d	2d
	End consultation> consultation in cancer	25.7d	29.4d	3.7d
	Consultation in cancer> biopsy report	20.8d	11.8d	9d
Treatment Interval	Biopsy report> treatment	39.7d	32.4d	7.3d

Final process models

As a final step, the final process map and associated parameters were documented to be integrated into the DES modelling process. The process maps, which provide a visual representation of the patient's journey from symptom discovery to treatment start, along with the parameters that were identified and measured during the study. Figure 4.12 depicts the final process maps for both public and private healthcare services, along with the associated parameters.

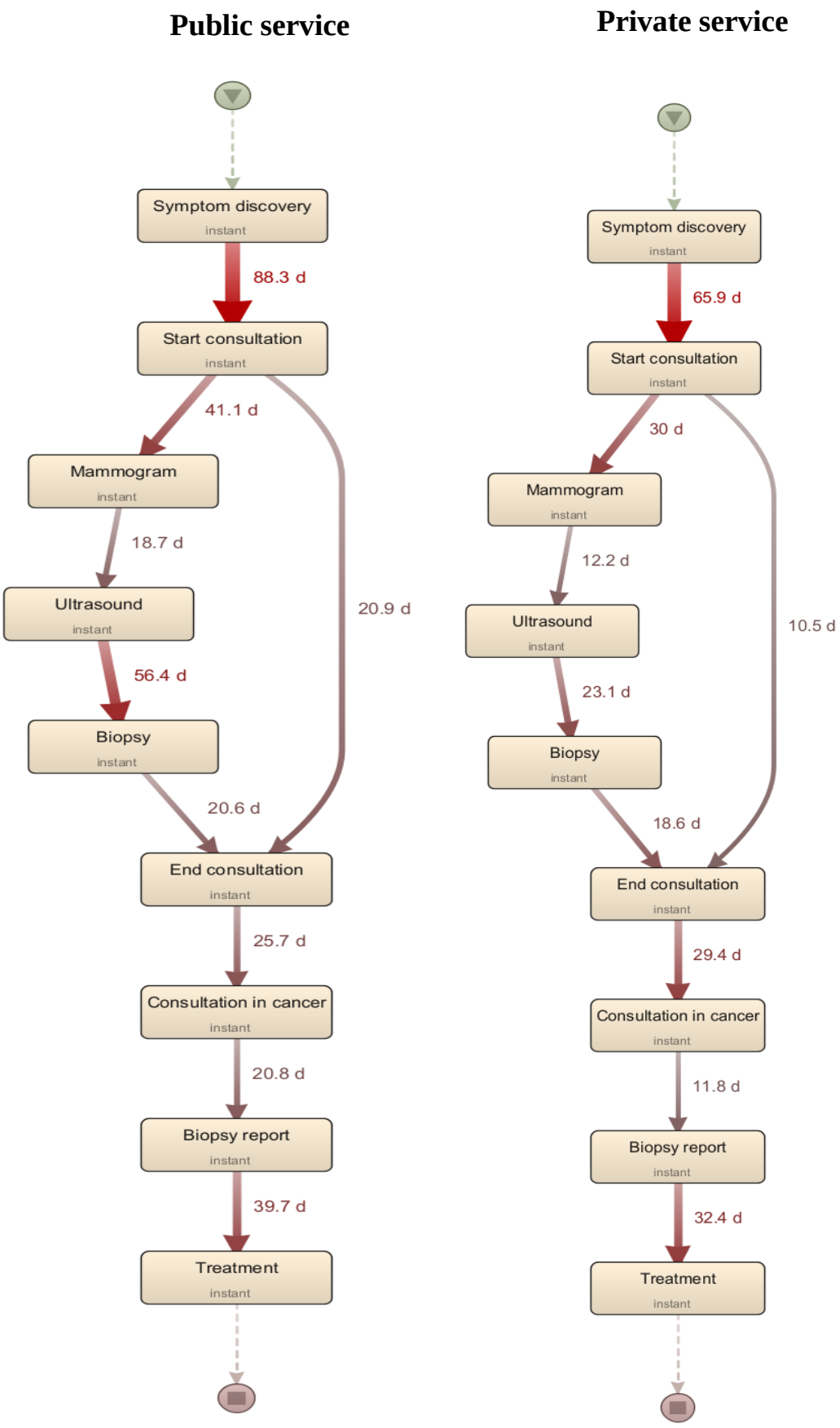


Figure 4.12 Process maps for public and private healthcare services

4.5.2.10 Step 6: Validation of the AS-IS Model

This step also corresponds to the validation activity in the framework and is operationalised through the review of flagged pathways with experts (see Section 4.5.2.8). Following abstraction, clinicians examined unusual or low-frequency pathways to confirm whether they represented meaningful clinical variations or artefacts of the data. Pathways were not excluded automatically; instead, expert judgment determined whether they should be retained or removed.

Through this validation, the AS-IS process model was confirmed as both empirically consistent with the event log and clinically robust. In particular, rare but clinically relevant variations were preserved, while atypical or non-standard behaviours (such as bypassing biopsy or skipping imaging steps) were excluded. This ensured that the final AS-IS model provided a credible representation of the real diagnostic pathways to be used in subsequent simulation activities.

Parameterisation

Parameterisation is a crucial step in discrete event simulation (DES) modelling. It involves identifying and defining the input parameters of the model, which are used to simulate the behaviour of the system being modelled. Although some input parameters can be derived from theoretical models or expert knowledge, many parameters are usually hard to estimate in a precise manner. This is the stage where process mining comes in to help in the generation of input parameters from actual data.

4.5.2.11 Step 7: Direct parameters generated by process mining

From process mining, we generated the process model along with its parameters, such as activity durations. These parameters can be used in later steps for further manipulation and refinement to meet the required specifications for analysis and simulation

A number of parameters can be derived directly from event logs, such as activity durations. These serve as input parameters for the Discrete Event Simulation (DES) model, ensuring an accurate representation of the process. In this case study, process mining was applied to analyse event logs and extract activity durations. Integrating these results into the DES model provides a realistic and data-driven representation of the process, enabling accurate simulation and scenario testing.

4.5.2.12 Step 8: Derived parameters

Nevertheless, some input parameters may need to be analysed further before they can be applied in DES modelling. For instance, the interarrival time distribution and the activity time distribution had to be obtained from the raw data in the event logs through statistical methods like fitting a probability distribution to the data.

Another parameter that needed additional analysis is routing percentages, which indicate the percentage of process instances that take a given path of activities. This parameter was obtained by the data analysis method from the logs of the event. Process flow, outlining the order of activities and their interconnections, had to be constructed based on the event logs through process mining tools like process discovery.

This section is dedicated to the process mining integration into the parameterisation stage of the discrete event simulation modelling. We detail the utilization of the process mining outcomes in obtaining the input parameters for the DES model. We present the different inputs derived from process mining as well as the algorithms and methods used to derive them. At the end of this chapter, the reader will have a definite idea of the process of incorporating process mining into the parameterisation stage of DES modelling, and the advantages to be derived from this integration.

Interarrival time

In this study, we used log-based process mining techniques to obtain the interarrival times for a discrete-event simulation (DES) model of a breast cancer case study.

Interarrival duration is the time between the occurrence of two consecutive events in a process. This was calculated using real data obtained from event log process mining. However, in order to do so, a simple manual calculation is necessary, which can be easily done using an excel formula. This calculation was demonstrated in Figure 4.13 and the algorithms is explained in the following paragraphs. Once the interarrival duration data is obtained, it can be entered into DES modelling. This allows for a more accurate representation of the process and can aid in the identification of potential areas for improvement.

The algorithm starts with setting the arrival time for the first case in the log to zero. This is because it is the first case coming into the system based on the given log, and therefore there is no time duration. For each subsequent case in the log, the algorithm calculates the time difference between the current case and the previous case using a formula. This time difference is commonly referred to as the interarrival time, and it represents the time elapsed between the arrival of two consecutive cases in the log.

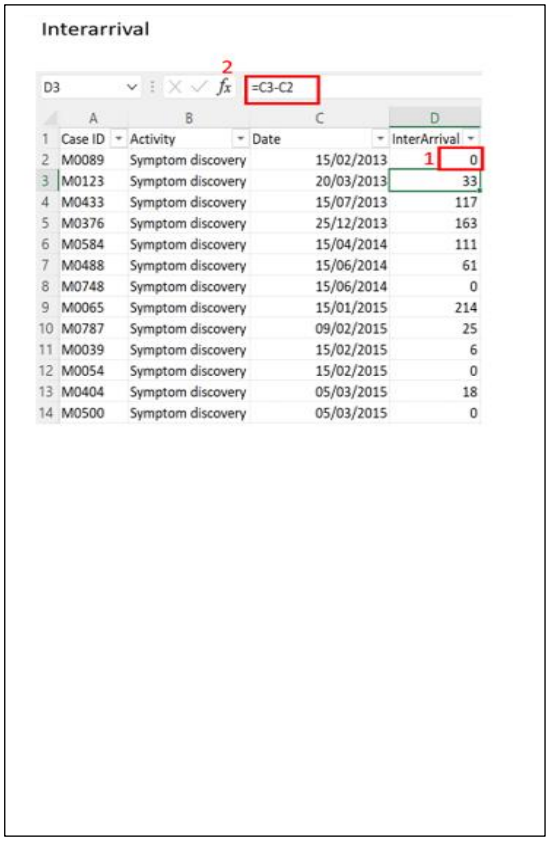
The formula used to calculate the interarrival time is as follows:

Interarrival time = Timestamp(current case) - Timestamp(previous case)

Once the interarrival time has been calculated, the algorithm sets the arrival time for the current case to the arrival time of the previous case and the calculated time difference. This ensures that the arrival times of cases in the simulation model are consistent with the arrival times in the log. The algorithm continues this process for all cases in the log until all cases have been processed.

Interarrival times were used directly to preserve the empirical arrival pattern, whereas inter-event durations (activity times) were modelled as empirical probability distributions to enable stochastic variation in DES. This approach allowed the variation in event-to-event timings to be reproduced across simulation replications.

While some patients may progress more rapidly through the pathway (for example in urgent cases), this relationship was not modelled explicitly as a patient attribute. Instead, such variation was incorporated implicitly within the fitted distributions, which captured the range of observed delays.



Algorithms Pseudocode:

1. Set the arrival time for the first case to 0.
2. For each subsequent case in the log:
3. Calculate the time difference between the current case and the previous case using a formula (Interarrival time = Timestamp(current case) - Timestamp(previous case)
4. Set the arrival time for the current case to the arrival time of the previous case plus the calculated time difference.
5. Continue until all cases in the log have been processed.

Figure 4.13 Interarrival calculation

Activity Probability Distribution

Parameterisation was required to transform the raw activity times from the event log into simulation-ready inputs. Process mining provided timestamps for each event, but these cannot be used directly in DES. Instead, activity durations were calculated by measuring the time intervals between consecutive events within the same case (process ID). These durations, together with transition frequencies, were then converted into probability distributions and transition probabilities, forming the key input parameters for DES modelling.

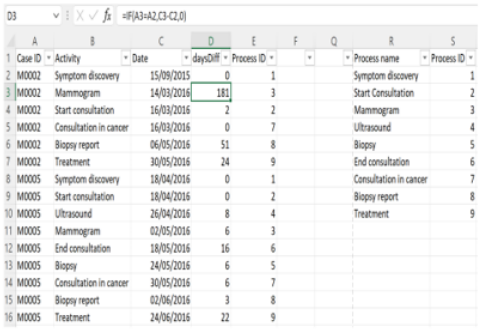
Activity durations were used in Simul8 to generate probability distribution profiles. These profiles represented the variability in time taken to complete each activity and allowed the simulation engine to randomly sample inter-event times. This ensured that the DES model captured both the average behaviour and the natural variability observed in the patient pathways, supporting realistic reproduction of the process while enabling analysis of alternative scenarios for improvement.

Because the observed data contained many outliers and did not conform to standard theoretical distributions such as Normal, Exponential, or Weibull, empirical distributions were adopted. The empirical approach was based directly on the frequency distribution of observed data, constructed by generating histograms and fitting a curve to the data. Outliers were reviewed carefully rather than removed automatically. Cases that were implausible or not relevant to the study objectives, such as extreme delays caused by data entry errors or administrative inconsistencies, were excluded. However, outliers that were clinically plausible and meaningful for understanding diagnostic delays were retained. This approach ensured that the variability of patient experience was preserved, while minimising the influence of erroneous values that could distort the model.

The main advantage of the empirical method is that it avoids assumptions about the underlying probability distribution and provides a closer fit to real data, particularly in the presence of outliers or skewness. A limitation is that large sample sizes are required to achieve stable distributions, and extrapolation beyond the observed range is not appropriate. To address the practical challenge of constructing empirical histograms, a

spreadsheet-based tool was developed to automate the creation of probability distributions in Excel. This tool allowed specification of data ranges, automatic generation of histograms, and definition of bins, making the empirical method feasible for use with large datasets.

Time interval between events



	A	B	C	D	E	F	Q	R	S
1	Case ID	Activity	Date	daysDiff	Process ID			Process name	Process ID
2	M0002	Symptom discovery	15/09/2015	0	1			Symptom discovery	1
3	M0002	Mammogram	14/03/2016	181	3			Start Consultation	2
4	M0002	Start consultation	16/03/2016	2	2			Mammogram	3
5	M0002	Consultation in cancer	16/03/2016	0	7			Ultrasound	4
6	M0002	Biopsy report	06/05/2016	51	8			Biopsy	5
7	M0002	Treatment	30/05/2016	24	9			End consultation	6
8	M0005	Symptom discovery	18/04/2016	0	1			Consultation in cancer	7
9	M0005	Start consultation	18/04/2016	0	2			Biopsy report	8
10	M0005	Ultrasound	26/04/2016	8	4			Treatment	9
11	M0005	Mammogram	02/05/2016	6	3				
12	M0005	End consultation	18/05/2016	16	6				
13	M0005	Biopsy	24/05/2016	6	5				
14	M0005	Consultation in cancer	30/05/2016	6	7				
15	M0005	Biopsy report	02/06/2016	3	8				
16	M0005	Treatment	24/06/2016	22	9				

Algorithms Pseudocode:

1. Assign a unique "Process ID" for each activity in the event log data.
2. For each process instance:
   a. Sort the activities by timestamp.
   b. Calculate the time difference between consecutive activities to obtain the time between events.
3. Use the process mining results to identify the main trajectories that appear most frequently.
4. Filter the time interval data to include only the main trajectories.
5. Use the filtered time interval data to create a probability distribution.
6. Incorporate the generated probability distribution into the DES model to simulate the interarrival times between events.

Figure 4.14 Calculation for activity distribution

Routing probability

The routing probabilities are calculated based on an analysis of the event log data. First, we find out the probability of occurrence for each path in a process by dividing the number of instances that followed a specific path to get the total number of those instances. These calculations can give an idea about the relative distribution of different paths on this dataset.

The frequency of each path is computed as the number of process instances that took that specific path. Thereafter the routing percentage is obtained by dividing frequency of each

path for total number of process instances. In Figure 4.15, for example, the proportion of patients moving from Biopsy Report to Treatment was calculated using the number of patients with a Biopsy Report as the denominator.

It is important to validate these derived percentages against the behaviour that was actually observed during analysis so as to confirm their accuracy. The purpose of this validation step is to ensure that the routing proportions derived do indeed reflect what was actually observed as process behaviour. Lastly, the obtained routing percentages act as input variables in the DES model.

	A	B	C	D	E	F	G	H	I
1	Case ID	Activity	Complete Timestamp	daysDiff	Process ID	Previous_T			
2	M0002	Treatment	30/05/2016	22	9	Biopsy report			
3	M0005	Treatment	24/06/2016	24	9	Biopsy report			
4	M0006	Treatment	08/07/2016	36	9	Biopsy report		Total Biopsy Reports	197
5	M0020	Treatment	04/07/2016	24	9	Biopsy report		**Routing Probability (BiopsyReport→Treatment)**	0.84264
6	M0023	Treatment	06/07/2016	29	9	Biopsy report			
7	M0028	Treatment	15/07/2016	28	9	Biopsy report			
8	M0029	Treatment	22/07/2016	34	9	Biopsy report			
9	M0032	Treatment	27/06/2016	47	9	Biopsy report			
10	M0033	Treatment	25/07/2016	28	9	Biopsy report			
11	M0034	Treatment	30/07/2016	26	9	Biopsy report			
12	M0039	Treatment	02/08/2016	46	9	Biopsy report			
13	M0041	Treatment	15/07/2016	28	9	Biopsy report			

Figure 4.15: Routing probability

4.5.2.13 Step 9: Parameter validation with expert

Derived parameters (durations, routing, arrivals) were checked for internal consistency and reviewed with clinical experts for face validity. Where discrepancies arose (e.g., implausibly long durations attributable to data entry), corrections followed the same expert-review.

4.5.3 Phase 3: Converting process mining results into DES modelling.

The aim of this section is to discuss the integration of process mining results into the development of a DES model for breast cancer pathways. Specifically, we examined how the conceptual model and parameters derived from process mining were incorporated into the DES model, together with the validation activities conducted to ensure model accuracy. By detailing this process, we aimed to demonstrate the potential benefits of integrating process mining with DES modelling for healthcare systems optimisation.

We focused on time-to-event modelling, as the available data captured diagnostic delays between activities. This approach highlights the duration between key events and their impact on the overall system, making it particularly suitable for healthcare pathways where delays are unpredictable.

In the following subsections, we elaborate on how the process mining outputs were translated and parameterised into simulation-ready inputs for DES modelling.

4.5.3.1 Step 1: Translate process map into a conceptual DES model

The process map generated from process mining was simplified and adapted into a conceptual DES model. The focus was on retaining the clinically meaningful diagnostic transitions and reducing unnecessary complexity. This produced a conceptual model aligned with the study objective of analysing diagnostic delays.

4.5.3.2 Step 2: Define routing logic and dependencies

The next step was to define the flow logic and dependencies between activities in the pathway. Using the process map as a guide, routing probabilities were specified for alternative paths, and dependencies such as the sequence between consultation, diagnostic imaging, and biopsy were captured from the process map and represented in the DES structure. These descriptions reflected the observed patient flow and guided how patients moved through the pathway in the DES model.

4.5.3.3 Step 3: Input parameters into the DES model

The final step was parameterising the DES model using values derived from process mining. Transition probabilities were calculated from the frequency of observed paths, while time-to-event distributions (e.g., consultation to mammogram, ultrasound to biopsy) were extracted from the event log. These parameters allowed the DES model to represent both the likelihood of different trajectories and the timing of events. The final process model adapted for DES is presented in Figure 4.12.

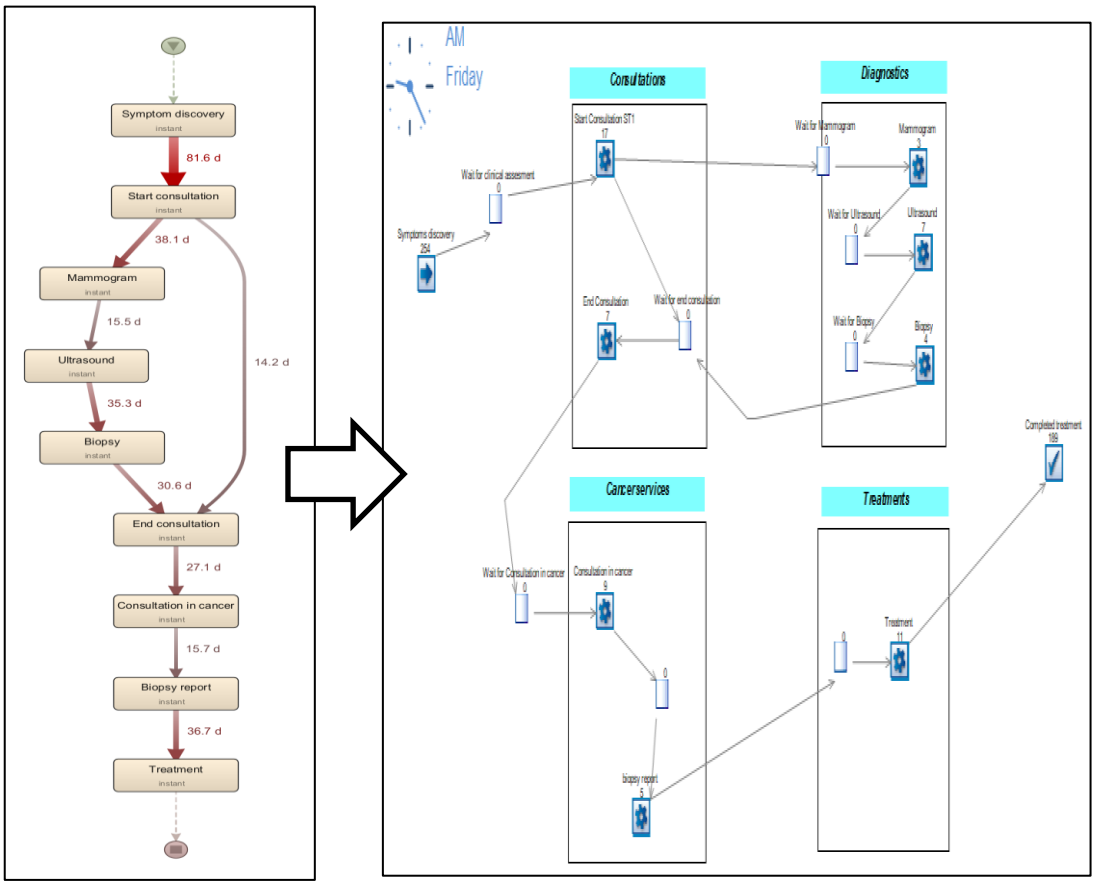


Figure 4.16 Final process model from process mining integrated into DES modelling.

Parameter Results from Process Mining and DES Modelling.

We compared results from DES modelling with results obtained from process mining.

Table 4.9 Comparing time interval results for public services.

Time to event	**DES**	**Process mining**
	Mean durations (CI) (days)*	Mean duration (CI) (days)
Symptom discovery > start consultation	89 (63.24,116)	88(62.34, 114.31)
Start consultation> Mammogram	40 (28.49, 73.61)	41(10.83, 71.32)
Mammogram> Ultrasound	18(3.45, 21.50)	19(2.97, 34.49)
Ultrasound> Biopsy	55(20.42, 80.54)	57(19.42, 93.35)
Biopsy> End consultation	19(7.46, 27.50)	21(6.31, 34.96)
End consultation> consultation in cancer	24(15.47, 27.55)	26(12.90, 38.51)
Consultation in cancer> Biopsy report	22 (16.45, 23.52)	21(12.15, 29.43)
Biopsy report> treatment	41(35.49, 47.51)	40(34.15, 45.30)

**The Discrete Event Simulation (DES) was conducted using 150 replications.*

Table 4.10 Comparing time interval results for private services.

Time to event	**DES**	**Process mining**
	Mean durations (CI) (days)*	Mean duration (CI) (days)
Symptom discovery > start consultation	64(43.45, 80.05)	65(45.36, 86.51)
Start consultation> Mammogram	28(17.32, 41.5)	30(13.93, 46.06)
Mammogram> Ultrasound	12(0, 21.92)	12(0, 25.17)
Ultrasound> Biopsy	21(10.50, 29.65)	23(13.32, 32.97)
Biopsy> End consultation	15(5.45, 25.67)	17(8.32, 28.87)
End consultation> consultation in cancer	32(15.66, 38.84)	30(18.01, 40.82)
Consultation in cancer> Biopsy report	16(10.57, 17.07)	12(10.03, 13.56)
Biopsy report> treatment	31(30.40, 38.55)	32(27.50, 37.26)

**The Discrete Event Simulation (DES) was conducted using 150 replications.*

4.5.3.4 Step 4: Simulation model

Scenario Analysis for Assessing the Impact of Diagnostic Delays

(Note: This scenario modelling does not include disease progression. Patients are assumed to remain at their initial stage throughout the pathway, so the analysis is limited to pathway efficiency i.e., changes in diagnostic intervals and initiation of treatment rather than survival estimation.)

We also conducted scenario modelling in DES to estimate the potential impact of reducing diagnostic intervals for breast cancer patients. The results of our study showed that shorter diagnostic intervals enabled patients to progress more quickly through the diagnostic pathway, leading to earlier initiation of treatment.

In the current scenario, based on patients who initially sought consultation at a public health centre, the overall average time spent in the diagnostic pathway was calculated using the proportions of patients in different stages of breast cancer (early stage, locally advanced, and metastatic) and the associated time intervals to treatment.

In the new scenario, assuming faster diagnosis and treatment, a proportion of patients initially classified as locally advanced or metastatic were assumed to be diagnosed earlier. This resulted in more patients starting treatment at earlier stages, reducing the average time to treatment.

In the early diagnosis scenario, we assumed that 15% of patients with locally advanced disease and 20% of those with metastatic disease would have been diagnosed at an earlier stage. These proportions were based on indicative evidence from diagnostic delay studies and expert input and were used for simulation purposes only.

Importantly, the models did not simulate transitions between stages of disease. Instead, staging was incorporated as a static variable at diagnosis, based on the collaborator's dataset. The proportions shown in Table 4.11 represent a redistribution effect when diagnostic delays are reduced, rather than predictions of disease progression.

In Table 4.11, the proportions of patients at each stage and the corresponding diagnostic intervals are presented for the current scenario, alongside the revised estimates for the new scenario with reduced delays. Additionally, the results highlight that reducing diagnostic intervals leads to earlier treatment initiation and shorter overall time in the system.

Table 4.11 Comparison of patient outcomes in the current and new scenarios (time in system analysis)

Stage	**Proportion of patients by stage in the Current Scenario**	**Proportion of patients by stage in the New Scenario**
Early Stage	0.389	0.482
Locally Advanced	0.464	0.415
Metastatic	0.146	0.103

(Note: Staging information was obtained from the collaborator's dataset and reflects stage at diagnosis (recorded at presentation to cancer services). The redistribution of patients across stages in the revised scenario results from shorter diagnostic intervals, which increased the proportion of patients diagnosed earlier. This analysis does not simulate disease progression but relies on observed associations between diagnostic delays and stage at diagnosis).

From our findings, it can be inferred that achieving shorter diagnostic intervals could dramatically improve patient outcomes, with the potential to increase the proportion of patients diagnosed and treated at earlier stages. This emphasizes the critical importance of timely treatment for breast cancer patients. Differences in patient outcomes can also be attributed to several factors, including accessibility of healthcare services, socioeconomic status, and regional variations in the availability of healthcare infrastructure. Such insights could be valuable in shaping public health policies and enhancing both the timeliness of treatment and the overall quality of breast cancer care in Mexico.

In addition, this approach assumes a fixed proportion of patients can be reclassified to earlier stages under improved diagnostic timelines. While this provides a useful proof of concept to explore the potential impact of earlier diagnosis on patient pathways, it remains a crude approximation. Therefore, conclusions should be interpreted with

caution, and future work should incorporate progression models to capture stage-shifting effects where data permits.

Model Scope: Exclusion of Survival Estimation

While the Discrete Event Simulation (DES) model developed in Chapter 4 offers insights into diagnostic delays and potential system improvements, it is important to clarify a key limitation: the model does not include disease progression logic and therefore cannot estimate survival outcomes.

No Disease Progression Modelled

The current model assumes that each patient enters the pathway at a fixed clinical stage (e.g., Stage I, II, III, or IV) and remains at that stage throughout the diagnostic and treatment process. It does not simulate how patients may transition from early to late stages as a consequence of process delays such as extended wait times for biopsy or consultation in cancer. Therefore, the potential clinical deterioration that can occur while awaiting care is not represented in the simulation.

Assumption Regarding Stage at Diagnosis

This study does not model disease progression over time. Patients are assumed to remain at the same clinical stage (e.g., Stage I, II, III, or IV) throughout their diagnostic and treatment journey. The analysis therefore focuses on differences in stage-at-diagnosis distributions between the public and private healthcare sectors. A key assumption is that patients in both systems present with similar symptom severity at the point of entry. The higher proportion of late-stage diagnoses in the public sector is attributed to longer diagnostic delays rather than more severe initial presentation.

Under this assumption, scenario analyses such as the “early diagnosis” intervention capture changes in pathway efficiency specifically reductions in diagnostic delays and earlier treatment initiation. These scenarios are not to be interpreted as survival modelling, as the DES model does not incorporate disease progression beyond stage at diagnosis.

This limitation could be addressed in future work by integrating a disease progression component. Doing so would allow for the exploration of how diagnostic delays may

influence patient outcomes and survival, particularly for late-stage cancers. However, the inclusion of such features was beyond the scope of the present study.

4.5.4 Phase 4: Validation

Sargent (2010) defines validation as the process of determining whether a simulation model is an accurate representation of the real-world system, within the context of its intended use. He categorizes validation into several distinct types. Conceptual model validation involves ensuring that the assumptions, process logic, and structure of the model correctly reflect the actual system being studied. This step is essential to confirm that the model is theoretically sound before implementation. Data validation focuses on verifying the accuracy, completeness, and appropriateness of the input data used in the model, ensuring that it faithfully represents real-world values.

Computerized model verification (distinct from validation) ensures that the model has been correctly implemented in the simulation software without coding errors or logical faults. Operational validation, also referred to as external validation, is the process of comparing the model's output to observed data from the actual system to determine how well it predicts or reflects real-world behaviour. Face validation involves presenting the model and its output to domain experts, who evaluate whether the model appears reasonable and credible based on their professional experience. This method is especially useful when empirical data for comparison is limited. Internal validation, also known as a consistency check, tests whether the model can reproduce outputs using the same dataset it was built from. While useful for identifying errors in logic or formulation, internal validation does not test the model's ability to generalize beyond the development data.

Although validation activities occur at multiple stages of the framework (e.g., conceptual validation in Phase 2 and internal consistency checks in Phase 3), Phase 4 is presented as a dedicated stage to consolidate external validation. While the framework illustrates scenario analysis within Phase 4, in this case study the scenarios were analysed earlier during Phase 3 (see Section 4.5.3) as part of the experimentation process. Accordingly,

Phase 4 focuses on validating the model outputs through expert review and literature comparison, ensuring model credibility before deriving insights.

In this study, we applied both internal and external validation to ensure the accuracy and credibility of the simulation model. Internal validation was conducted by using the same dataset, event log data from Mexican breast cancer patients to both develop and evaluate the model. We validated that the model successfully reproduced known outcomes from the dataset, such as the average patient interval of 88.3 days from symptom discovery to the first consultation, confirming internal consistency and correct implementation. To complement this, external validation was performed by comparing the model's outputs to real-world data from independent literature sources that were not used during model development. For example, the simulated diagnostic interval of 71.2 days was compared against findings from Unger-Saldaña et al. (2011), who reported delays of more than four months for the combined diagnostic and treatment intervals, primarily due to provider-related issues such as misdiagnosis, repeated consultations, and referral delays.

4.5.4.1 Expert Face to Face Validation

Presentation of Process Mining Results

We presented the process mining results to a clinical expert, with further explanation provided in Chapter 4, Section 4.5. The session involved a detailed review of the mined pathways for Mexican breast cancer patients, with a focus on key variations between public and private healthcare systems, pathway frequencies, and time-to-event intervals. Through pathway visualisations, the expert identified delays in public care particularly in diagnostics and treatment initiation while confirming the streamlined nature of private healthcare pathways. Common trends and outlier pathways were discussed, including those with extended timelines due to procedural inefficiencies or complex diagnostic needs. The oncologist's feedback was critical in validating the realism, completeness, and clinical relevance of the model. They ensured that all meaningful steps were represented and also evaluated the reasonableness of time intervals between key events using their clinical benchmarks. Discrepancies in the data, such as unexpectedly short or prolonged intervals, were flagged for further investigation to ensure alignment with real-world experiences. This expert input helped refine the model, ensuring it balances data-driven

insights with clinical realities, thereby strengthening its reliability and usefulness for Discrete Event Simulation (DES) modelling.

4.5.4.2 Literature Comparison on Delay Patterns

We compared the results from the process mining analysis with findings reported in the literature

Public Healthcare Delays

Research by Unger-Saldaña (2011) indicates that diagnostic delays in the Mexican public healthcare system commonly range between 8 to 12 weeks (approximately 2 to 3 months). These delays were attributed to administrative barriers, limited diagnostic capacity, and multiple consultations before referral for imaging and biopsy. This is consistent with the delays observed in the process mining analysis, where the diagnostic interval averaged 71.2 days (~10 weeks) in public services.

Private Healthcare Delays

Although data on diagnostic intervals in Mexico's private healthcare system are limited, existing literature suggests that patients generally experience faster access to diagnostic services in private settings compared to the public sector. This aligns with the time-to-event intervals observed in the private pathways from the process mining analysis, where the pathway from symptom discovery to diagnosis is notably quicker (Unger-Saldaña et al., 2021).

Public Healthcare Services

Table 4.12: Example of patient, diagnostic, and treatment intervals with durations obtained from process mining and literature sources (Public Services).

Interval Name	Description	Mean Duration from Process Mining	Duration details from Literature	References
Patient Interval	Time from symptom discovery to first medical consultation	88.3 days	Delays >1 month – Barriers: low awareness, socioeconomic constraints, and fear.	Unger-Saldaña et al. (2011), Bright et al. (2011)
Diagnostic Interval	Time from first medical consultation to diagnosis	71.2 days	Delays more than 4 months due to provider-related issues (Diagnostic + Treatment). – Linked to misdiagnosis, repeated consultations, and referral delays -6.6 months ≈ 198 days for diagnostic interval and 0.6 months ≈ 18 days for treatment interval (Bright et al.)	Unger-Saldaña et al. (2011), Bright et al. (2011)
Treatment Interval	Time from diagnosis to initiation of treatment	39.7 days		

Private Healthcare Services

Table 4.13: Example of patient, diagnostic, and treatment intervals with durations obtained from process mining and literature sources (Private Services).

Interval Name	Description	Mean Duration from Process Mining	Duration details from Literature	References
Patient Interval	Time from symptom discovery to first medical consultation	65.9 days	No specific data found.	
Diagnostic Interval	Time from first medical consultation to diagnosis	56.9 days	No specific interval durations found but literature suggests access to diagnostic services is generally faster in private healthcare settings due greater resource availability.	Unger-Saldaña et al. (2021);
Treatment Interval	Time from diagnosis to initiation of treatment	32.4 days	No specific data found	

Expert validation and literature comparison provided external confirmation that the model outputs were both clinically plausible and consistent with reported evidence, thereby consolidating Phase 4 as the dedicated validation stage of the framework.

4.6 Value of Process Mining in DES Model Development (Mexican Breast Cancer Case Study).

The following five points demonstrate how process mining contributed to the development of the DES model in the Mexican breast cancer case study. Each point is explicitly linked to the comparative discussion in Section 6.6, showing how the case study evidence grounds and exemplifies the broader analysis of traditional versus process mining approaches.

1. Revealing Clinical Pathway Variants

A key strength of process mining was its ability to reveal all observed pathway variants from the event log, providing an evidence-based view of real-world patient flows. This included both expected sequences and variations that may not be widely recognised or routinely documented, such as pathway deviations, less frequent routes, and branching patterns that were critical to understanding the full spectrum of how breast cancer care was delivered in Mexico.

In the private healthcare sector, analysis of the raw event log revealed that a notable proportion of patients accessed diagnostic procedures such as mammograms or ultrasounds directly, without a preceding consultation or formal referral. While this non-standard pattern was not anticipated in the initial conceptual model, the final process model presents an abstracted form to improve clarity while retaining the pathways relevant for simulation.

In the public sector, process mining identified cases where patients underwent multiple visits to general practitioners before being referred for diagnostic investigations. This looping behaviour, indicative of rework or delays in progressing through the system, added a layer of complexity that would be difficult to detect from summary statistics or protocol-based descriptions alone.

These findings provided valuable insights into the diversity of patient journeys and informed decisions on how to structure the simulation model to reflect this variation

appropriately. By identifying and analysing distinct care patterns such as direct access to diagnostics or repeated pre-diagnostic visits, process mining supported the inclusion of clinically and operationally relevant pathway variations. This ensured the DES model could simulate care processes that better aligned with actual patient experiences and account for differences in timing and service use across sectors enabling more realistic projections and targeted scenario analyses, particularly when exploring the impact of reducing diagnostic delays or improving access equity.

In the Mexican breast cancer case study, process mining generated an evidence-based view of patient pathways, revealing important variations that extended beyond the standardised descriptions often used in modelling. In the private sector, some patients accessed diagnostic procedures such as mammograms directly without prior consultation, while in the public sector, others experienced multiple consultations with doctors before eventually being referred for diagnostics. These insights informed the development of the DES conceptual model by highlighting variations that needed to be considered. They provided more than descriptive observations, as they directly guided modelling decisions to ensure that the DES structure reflected both expected and unexpected patient journeys.

This demonstrates the added value of process mining, in line with the discussion in Section 6.6, by showing how empirical evidence complemented expert knowledge to capture pathway variations that may not typically be identified in traditional DES model development based solely on expert recall

2. Evidence-Based Simplification and Pathway Filtering

In Chapter 4, the initial process model generated from the full event log of the Mexican breast cancer case study revealed a highly complex “spaghetti model.” This raw representation captured the full diversity of patient experiences including rework loops, skipped steps, and rare deviations but was not suitable for simulation due to its lack of clarity and overwhelming complexity. Traditionally, simplification of such models would rely heavily on clinician judgment to decide which pathways to retain. Process mining added value by providing an empirical, transparent, and repeatable method for making

these simplification decisions, while ensuring that the final abstracted model retained the most relevant behaviours for simulation.

First, variant analysis enabled the identification of dominant and exceptional pathways based on observed data rather than assumptions. This moved the modelling process beyond relying solely on clinicians' expectations of a 'typical' patient journey. For example, rare but extreme sequences such as a direct jump from Start consultation to Biopsy after a 17-week delay, or a feedback loop between Biopsy and Ultrasound were identified in the raw model but abstracted in the final representation for clarity. These patterns, while not widely recognised by clinicians, were objectively evaluated for their relevance to the simulation scope.

Second, the use of frequency and performance-based filtering (e.g., with Fuzzy Miner) allowed simplification to be guided not only by how often a path occurred, but also by how much delay or inefficiency it contributed. This added a quantitative layer of justification that complemented expert-led pathway selection by providing a data-driven basis for decision-making. For example, transitions with excessive delays were flagged and reviewed for their operational impact, not just their frequency. This ensured the DES model focused on sequences that were both frequent and significant to system performance.

Third, process mining made it possible to visualise and compare pathway structures across cases, supporting informed discussions with clinical experts. In cases of ambiguity such as the decision to retain the Ultrasound → Biopsy path despite its moderate frequency expert judgment was integrated, but only after the data highlighted it as potentially meaningful. This evidence–expert dialogue would not have been triggered in a traditional workflow and ensured the model reflected both empirical reality and clinical relevance.

In the Mexican breast cancer case study, process mining supported evidence-based simplification by providing a structured view of patient pathways and highlighting patterns that were most relevant for modelling. This allowed complexity to be reduced without losing clinically meaningful behaviour.

As discussed in Section 6.6, process mining offered a data-driven basis for simplification, while expert input confirmed which pathways were most appropriate to retain. This combination made the abstraction process more transparent. Compared with traditional DES model development, which relies largely on clinician judgement, process mining ensured that important variations were less likely to be overlooked.

3. Determining Level of Detail for Simulation

A core challenge in building discrete event simulation (DES) models is deciding the appropriate level of detail to include. Too much detail produces complex models that are inefficient to run; too little risks oversimplifying important dynamics. In Chapter 4, process mining added value by providing objective, data-driven guidance on which events and variations warranted detailed modelling, and which could be simplified. Rather than relying solely on clinician input which often reflects idealised or standardised pathways process mining enabled us to examine actual care sequences recorded in the event log. This included time durations, frequencies, and transition patterns across the full patient cohort, giving a factual basis for determining which steps in the pathway needed to be modelled explicitly.

For example, in the public sector pathway, process mining revealed a substantial time gap between biopsy and specialist consultation. While clinicians might describe this as part of a single diagnostic phase, the data showed that the interval contributed significantly to overall system delay. As a result, these steps were represented separately in the DES to capture the impact of this delay on patient flow. In other cases, such as Mammogram followed by Ultrasound, the activities occurred in a consistent order across most pathway variants. Process mining confirmed their stable sequencing, so both activities were retained as distinct steps in the DES model to reflect the diagnostic workflow.

Process mining also revealed repeated instances of certain tests (e.g., re-imaging) that occurred in practice but had little influence on pathway structure or total time in the system. In these cases, they were abstracted into a single averaged event, reducing complexity without losing meaningful behaviour. Finally, visualising and quantifying

variant frequencies helped decide which exceptions to exclude or merge, resulting in a balanced level of detail that preserved fidelity to patient journeys where it mattered most, while abstracting where added complexity had little value.

In developing the DES conceptual model, a key challenge was deciding how much detail to include without making the model either overly complex or too simplistic. Process mining supported this by quantifying delays, frequencies, and rework patterns in the event log, helping to identify where detailed representation was important and where simplification was appropriate.

As discussed in Section 6.6, process mining provides an evidence-based foundation by using event log data, while expert knowledge offers clinical insight and validation. Together, these perspectives support decisions on the appropriate level of detail. Compared with traditional DES model development, which relies mainly on expert judgement, process mining adds a data-driven complement that makes it less likely important delays and variations would be overlooked.

4. Data-Driven Cross-Checking and Refinement of Expert Knowledge

A key expectation of this research was that process mining would complement traditional, expert-driven pathway design by comparing observed care delivery with clinician expectations and refining the resulting simulation model with these insights. In Chapter 4, process mining demonstrated this value by confirming areas of alignment while also identifying differences, leading to a representation of the discrete event simulation (DES) model that was better aligned with observed practice.

Clinicians provided essential input on the expected care process, describing a structured referral pathway beginning with Start consultation, progressing through diagnostic activities such as Mammogram, Ultrasound, and Biopsy, and culminating in Cancer consultation before treatment. This description aligned with institutional protocols and was reflected in the majority of cases. However, process mining enabled a deeper, data-driven interrogation of these assumptions, revealing variations in actual care delivery that would have been difficult to detect through interviews alone. For example, analysis of

the event logs showed that some patients revisited Start consultation multiple times before advancing to diagnostic imaging, while others skipped intermediate stages entirely moving directly from initial consultation to biopsy or treatment. These patterns were not mentioned during early discussions, likely because they fall outside standard expectations or occur too infrequently to be recalled accurately.

Rather than dismissing these as anomalies, the deviations were quantified and assessed for their frequency and potential impact. Where they occurred at non-trivial rates, these deviations were noted during model development and informed decisions on how pathways were represented in the DES but were not explicitly depicted as separate branches in the final conceptual model diagram. This allowed the simulation to reflect real-world variability in patient journeys, rather than a simplified "ideal" flow, and supported more realistic estimation of delays particularly in the public sector, where variability was more pronounced.

Visualising the full set of observed pathways also informed stakeholder discussions and helped refine model assumptions. For example, when clinicians were shown sequences involving repeated Start consultations or diagnostic backtracking, they acknowledged such cases did occur but had not featured in their original conceptualisation. This prompted constructive dialogue and consensus on which behaviours should be explicitly represented in the model structure a process that would have been unlikely to emerge from expert consultation alone.

Process mining provided an evidence-based view of the care pathways, which was then cross-checked with expert input to refine and enrich the model structure. The event log highlighted variations that were not initially emphasised by clinicians, complementing the structured pathways they described.

As discussed in Section 6.6, this combination of empirical evidence and expert knowledge ensured that the DES model reflected both real-world practice and clinical insight, enhancing its relevance and interpretability. This type of evidence–expert refinement is usually difficult to achieve in traditional DES model development, which depends mainly on expert consultation.

5. Parameter Estimation for Early Modelling

Process mining was used to extract initial values for activity durations and related timing parameters directly from the event log. These values provided a practical starting point for building an early-stage DES prototype, allowing the model to be populated quickly with timing estimates grounded in observed care pathways.

However, it is important to note that these outputs are descriptive summaries typically means or medians and are limited to what can be directly observed in the log. Process mining does not provide confidence intervals, quantify uncertainty, or perform the more rigorous statistical validation needed for robust modelling. For this reason, the estimates were treated as provisional inputs to support early model development rather than definitive parameters.

In later stages of modelling, these preliminary values must be replaced with rigorously validated inputs obtained through formal statistical methods. This includes fitting appropriate probability distributions, estimating variance and confidence intervals, and explicitly incorporating uncertainty where relevant. Thus, process mining functioned as a rapid, evidence-based way of parameterising the initial model, complementing but not replacing comprehensive statistical analysis.

Process mining was used to extract initial parameter values, such as activity durations and timing estimates, directly from the event log. This provided a rapid and transparent starting point for early model development, before more formal statistical validation was applied.

As discussed in Section 6.6, traditional parameterisation approaches typically use real-world data where available, drawing on literature values or clinician estimates only when such data are lacking. Process mining complemented this by deriving values systematically from the event log, ensuring that early parameters reflected local practice while still requiring further statistical refinement for robustness.

4.7 Discussion

The discussion section provides a comprehensive analysis of the potential of integrating process mining techniques in DES modelling applications for breast cancer patient pathways. The use of data-driven process models has enabled the identification of key delays in the diagnostic process that contribute to treatment delay and ultimately impact patient outcomes.

The results of our study suggest that the application of process mining allows improving the accuracy of DES models, and, as a result, obtaining more reliable estimates of patient delays and pathway performance in general. Additionally, the integration of process mining also enabled us to have a detailed view of the breast cancer patient pathways since it was able to outline the most frequent patient pathways and differences in patient trajectories. The revelation that a major gap exists between the diagnostic and treatment emphasizes the necessity of an enhanced health policy.

4.7.1 Data processing

The framework we developed was divided into four main phases. In the first phase, we obtained data on the overall process, including patient demographics, clinical information, and timestamps for each event in the process. This dataset contained missing values, inconsistencies, errors, which can affect the accuracy of the analysis. This data required significant data preprocessing efforts, such as data cleaning, data transformation, and data enrichment. Data preprocessing was crucial for the accuracy of the results, but it was time-consuming and required specialized skills.

In order to improve the integration framework with these challenges, several solutions were implemented. Given secondary data was used, we asked the data owner to cross-reference the data from multiple sources and check the accuracy and completeness of the data. By doing so, the data owner helped to identify any missing or incorrect information and make the necessary corrections. In addition, incorporating domain expertise was considered crucial in identifying the most important variables to be included in the analysis and to focus on the key factors that affect the delay of diagnosis and treatment

of breast cancer. This step played a vital role in reducing the dimensionality of the data and improving the accuracy and interpretability of the results.

Getting a clear event definition is important, the challenge of using secondary data is to understand each event in the dataset is very dependent on expert knowledge. In addition, it is very important for researchers to understand in depth the whole process so that researchers can detect any unusual behaviour in the process model and get confirmation from process experts. It is important to provide events with clear names is very crucial so that they are easy to understand. In the case study, the initial naming in the dataset is not clear, so this caused a lot of confusion about the actual definition. For that reason, researchers need to seek confirmation from process experts repeatedly.

Moreover, visualisation techniques were used to detect outliers, missing values, and inconsistencies quickly in the dataset. Graphical representations of the data were created to identify patterns and trends, and potential errors or anomalies were easily spotted. This helped to save time and effort in data cleaning and preprocessing and improved the accuracy and reliability of the analysis. Histograms were used for this purpose. These techniques are particularly useful when dealing with large datasets, and when the data is complex and difficult to interpret.

We also faced challenges when creating the event log due to the varied formats and structures of the data sources. Extracting and transforming the data into a standardized format for analysis was a complex task. The data we worked with had differing levels of detail, the timestamps have varying levels of granularity. Standardizing the timestamps across all event logs was time-consuming and required careful attention to detail.

4.7.2 Conceptual modelling

The second phase involved the application of process mining techniques to create a data-driven process model for breast cancer patient pathways that depicted both common trajectories and variations in patient journeys.

During the process of creating an initial process map, a significant challenge that was encountered was the presence of many exceptions and deviations from the expected process flow. The low quality of the dataset contributed to this challenge. Despite these challenges, efforts were made to ensure that the resulting process map and parameters were still accurate and informative.

To address these challenges, again we employed data cleaning to improve the quality and consistency of the dataset. This involved identifying and resolving missing or incorrect data and filtering out irrelevant or incomplete data. Furthermore, we conducted manual inspection and validation of the initial process map with the help of subject matter experts, who relied on their experience and expertise to validate the map. By doing so, we were able to identify and correct any inaccuracies and gaps in the initial process map.

Due to the nature of missing and incomplete data detection and rectification during pre-processing, for this dataset, data cleaning and validation became an iterative process. This included multiple iterations of validating and correcting both data as well as process model with expert. However, despite these difficulties, the desired process map and parameters were obtained using process mining approaches.

4.7.3 Selection of algorithms

We tested several process mining algorithms before deciding to use Fuzzy Miner for our analysis, see the previous section for detailed comparisons and evaluation criteria. We used the algorithms within ProM Software and Disco software (Built-in Fuzzy Miner) to generate the process map and ultimately selected Fuzzy Miner due to its suitability for modelling complex, unstructured, and variable healthcare processes.

Fuzzy Miner was chosen because it does not enforce a rigid logical structure on the process. Instead, it analyses the frequency and strength of relationships between activities, making it especially appropriate for healthcare datasets where patient journeys are rarely uniform. Unlike more structured algorithms such as Alpha Miner or Inductive Miner, Fuzzy Miner interprets the process "as-is," which aligns well with the dynamic and often non-linear nature of clinical workflows.

Additionally, Fuzzy Miner includes thresholding features that allow abstraction levels to be adjusted. This enables us to present simplified high-level views or more detailed pathways depending on the needs of different stakeholders. Its flexibility and visual clarity make it highly useful for communicating findings with clinicians and decision-makers. The use of Fuzzy Miner also simplified the analysis process and made it more accessible to users without advanced data analysis or programming expertise. Disco's intuitive interface allows for rapid generation and interpretation of visual process maps, enhancing both usability and stakeholder engagement.

In light of these advantages, we recommend exploring multiple algorithms where appropriate and comparing their outputs to ensure that the selected process map accurately reflects the data characteristics and analysis objectives. However, given our case study's complexity and variability, Fuzzy Miner emerged as the most effective and practical choice.

4.7.4 Data abstraction

The initial process map generated was too complex. We used process expert knowledge to identify important processes in consideration with the research objectives and problems being investigated. Moreover, we used subgrouping to focus on a specific subset of the process that was most relevant to the research objectives. In this case study, we conducted subgrouping of different service providers which is public service and private services to see the delays that occur in diagnostics intervals.

In addition, Disco uses evaluation metrics from process mining to evaluate and validate the resulting process map. The software uses fitness metrics and precision metrics to determine the accuracy of the process model. Additionally, the software considers the simplicity and coherence of the model to ensure it is easy to understand and maintain, and that the process flow follows the expected order of activities. These metrics help to ensure that the process map is accurate and reliable, even when derived from a low-quality dataset. The fitness metric is used to measure how well the process map matches the actual process data. The precision metric is used to measure the accuracy of the process map in predicting the outcome of a process instance. The simplicity metric is

used to measure the complexity of the process map, and the coherence metric is used to measure the consistency of the process map, such as whether the process flow follows the expected order of activities.

4.7.5 Translating into DES model

In the third phase, we integrated the process model into a DES simulation model of breast cancer patient pathways and identify opportunities for improvement.

Process models generated from process mining can be very complex, with many different activities and paths. This can make it difficult to translate the model into a DES model, as it requires simplifying the process while still maintaining its essential characteristics. Process mining generates a process model based on event logs, which may not be in the format required for DES modelling. Data transformation may be necessary to convert the event log data into a format that can be used in the DES model.

It is important to validate the accuracy of the process model generated from process mining before using it as input for DES modelling. Validation involves checking the model against the actual process to ensure that it accurately reflects the real-world process.

However, the methodology also has some limitations and challenges. One of the main challenges is the data quality and completeness, which can affect the accuracy and generalizability of the process models and simulation results. In addition, the methodology requires a significant amount of data preparation, analysis, and modelling expertise, which can limit its applicability and scalability in some contexts.

4.7.6 Recommendation

We would recommend that machine learning methods be applied for preprocessing data in future work and automate the detection and treatment process of missing values. Using the strength of machine learning algorithms, it is possible to learn from patterns and relationships in the data, allowing predictions to be made about missing values based on what information is present. Sophisticated models of imputation can be developed using machine learning algorithms. In data cleaning and preparation, automating the preprocessing of data through machine learning is a strategy that saves on time. Nonetheless, when using any automated preprocessing methods, it is important to validate the results thoroughly for accuracy and reliability.

In addition, if primary data are utilized, we suggest creating a standardized protocol for collecting information to guarantee its consistent and accurate collection. This protocol ensures that the data being collected is high quality and consistent, thereby minimising errors and inconsistencies in the data. Healthcare providers can achieve standard data collection by setting rules and regulations of how to collect the required information.

Specialized software may be required to handle such large datasets and also because running large datasets can be time-consuming and require significant computational resources. Furthermore, the technical expertise of the analysts is crucial in ensuring that the appropriate algorithms and techniques are used to analyse the data effectively. It is essential to have analysts who are skilled in using data analysis tools, such as Python or R to process the data quickly. Based on our experience implementing the dataset, we needed to extract data from the database before manipulating it in R programming to ensure that the analysis was well-organized and easy to perform.

Moreover, we have learned valuable lessons that we will use as a checklist for future integration of this type of dataset. One of the critical elements of the checklist is the need for expert opinion. As seen in this case study, process mining requires a deep understanding of the process being analysed, and without expert opinion, it can be challenging to accurately model the process and derive meaningful insights from the data.

4.7.7 Linking to the Kidney Case Study

The Mexican breast cancer case study provided the first implementation of the framework, demonstrating how event log preparation, process mining, abstraction, and expert validation could be combined to generate a conceptual model suitable for DES. Key lessons from this study included the need for careful event log cleaning to address inconsistencies, the use of pathway abstraction to manage complexity, the role of clinical expert input in interpreting patient trajectories, and the importance of algorithm selection.

These lessons directly informed the kidney case study in Chapter 5. While applied in a different clinical context, the same principles were carried forward. In particular, pathway abstraction was again used to focus on the critical elements of the care process, and Fuzzy Miner was adopted as the process mining algorithm to simplify complex pathways while retaining clinically meaningful transitions. This continuity demonstrates how methodological refinements from the breast cancer study strengthened the design and implementation of the kidney study.

CHAPTER 5

CONCEPTUAL MODEL AND PARAMETERS RESULTING FROM PROCESS MINING WITH THE UK KIDNEY DATASET

5.1 Chapter Overview

This case study builds on the lessons from the breast cancer study in Chapter 4, where event log cleaning, pathway abstraction, algorithm selection, and expert validation were shown to be important for developing conceptual models for DES. The same principles are applied in a different clinical context of peritoneal dialysis. In particular, pathway abstraction is used to focus on the critical elements of the care process, and Fuzzy Miner is adopted as the process mining algorithm to simplify complex pathways while retaining clinically meaningful transitions. The methods we used in Chapter 5 are grounded in the methodological experience from Chapter 4, and this chapter applies the proposed step-by-step framework to derive the conceptual model and its parameters from a peritoneal dialysis dataset, with the aim of assessing and refining the framework based on the experience gained. The process model generated through process mining is then compared with the existing conceptual model developed by our collaborator in Chapter 6.

5.2 Background

This case study focuses on a specific type of treatment for kidney patients known as peritoneal dialysis. Within this context, we explore the implications of two distinct techniques used for catheter insertion in PD patients: medical and surgical methods.

Kidney disease, known as renal disease in the medical field, is a condition marked by a decline in the kidney's normal function. These vital organs serve pivotal roles within the body, undertaking various essential functions. They act as filters, eliminating waste products, excess fluids, and toxins from the blood circulation. Furthermore, they maintain electrolyte equilibrium, participate in blood pressure regulation, produce hormones vital

for red blood cell production, and contribute to the preservation of healthy bones(Braun & Khayat, 2021).

Dialysis is a treatment approach used to treat kidney failure. It encompasses two main methods: hemodialysis and peritoneal dialysis. Hemodialysis employs a specialized machine called a dialyzer, which mimics the function of a natural kidney by filtering the blood. In contrast, peritoneal dialysis offers a home-based alternative utilizing the peritoneum, a thin membrane surrounding the abdominal cavity, to naturally filter and cleanse the blood(Masakane et al., 2015).

5.2.1 Introduction to Peritoneal Dialysis

Peritoneal dialysis (PD) is a form of kidney dialysis that makes use of the peritoneum, which is the lining found within the abdominal cavity and acts as a filter for waste material and fluids in excess to be removed from blood. PD is often performed at home, granting patients much more freedom and a sense of flexibility than other dialysis methods (Teitelbaum & Burkart, 2003).

In PD, a catheter is surgically inserted through a small opening to the patient's peritoneal cavity. The catheter is used to fill the abdominal cavity with a dialysate, which is further responsible for removing waste and unnecessary fluid from the blood. The dialysate sits in the peritoneal cavity for some time, usually a few hours, and during this period it sponges waste products and unnecessary fluids out of the blood. The patient then discards the used dialysate and refills his or her abdomen with fresh dialysate to restart the process (Rockney & Kleeman, n.d.).

Peritoneal dialysis (PD) catheter insertion can be done through two different approaches: medical and surgical. The medical approach involves inserting the catheter using a minimally invasive technique under local anaesthesia. This method typically involves using a stylet to guide the catheter into the abdomen, without the need for an incision. This technique is often referred to as the percutaneous method. The surgical approach involves making a small incision in the abdomen and tunnelling the catheter through the subcutaneous tissue and into the peritoneal cavity(Wright et al., 1999). This method is known as the open surgical method and is typically done under general anaesthesia.

Both methods have their own advantages and disadvantages, and the choice of method depends on various factors, such as the patient's anatomy and medical history. The introduction of the percutaneous technique of insertion, as well as other methods for Medical PD catheter placement, has made it possible to perform this procedure at the patient’s bedside or in an outpatient setting, which is much more preferable. The surgery is related to higher risk of complications, such as bleeding and infection, and longer recovery time(Tullavardhana et al., 2016). However, it may not be suitable for all patients and may not be as effective as surgical PD catheter insertion in terms of catheter survival and long-term outcomes.

On the other hands, surgical PD catheter insertion such as laparoscopic or open surgery is an invasive procedure that involves more recovery time and could be associated with a higher risk of complications. Nevertheless, it is considered to be more efficient in regard to catheter survival and overall long-term results.

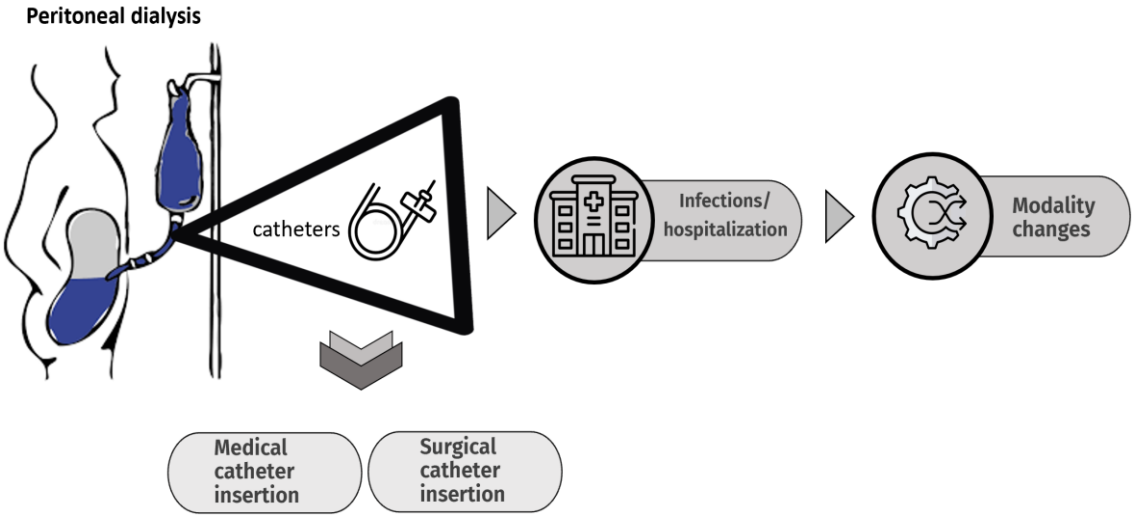


Figure 5.1 Peritoneal Dialysis

5.3 Aim of the case study

The use of this dataset serves several objectives. Firstly, it aims is to generate process mapping for patients undergoing Peritoneal Dialysis (PD) by employing process mining techniques. This process mapping provides a visual representation of the sequential steps and activities involved in the PD treatment process, allowing for a better understanding of the patient pathways and the overall flow of the treatment.

In addition, the dataset enables an examination of the difference between the medical catheter insertion technique and the surgical catheter insertion technique in relation to the time taken to initiate PD dialysis.

Another aspect of interest is examining the time to infection among patients who undergo medical catheter insertion versus surgical catheter insertion. By analysing the infection rates and durations for each group, we can gain insights into the potential impact of the insertion technique on the occurrence and timing of infections, thus informing clinical decision-making.

To validate the results and assess the value of process mining, the process models and parameters derived from process mining were compared with the results obtained from the existing model structure. This comparison allows for an evaluation of the completeness and accuracy of the process models generated through process mining, providing key points into the potential benefits and limitations of utilizing process mining techniques in healthcare process analysis.

Lastly, documenting the challenges and lessons learned from this case study contributes to the improvement of the integration framework developed. By identifying and addressing the challenges encountered during the data analysis and process mapping, valuable insights can be gained, and future studies can benefit from this knowledge.

5.4 Dataset

This section aims to provide a comprehensive explanation of the secondary dataset, which specifically pertains to the treatment of peritoneal dialysis (PD) for kidney patients. The dataset includes a wide range of variables that capture various aspects of the PD treatment process, patient demographics, clinical parameters, and outcomes. These variables encompass information such as patient characteristics (age, gender, race), medical history, treatment modalities, infections, and hospitalisation data.

5.4.1 Overview of the dataset

The present study utilized a secondary dataset provided by a collaborator, comprising data of 802 patients. All data collected for this study has undergone anonymization procedures to protect patient privacy, ensuring that the original identities of the patients

remain unknown to the researchers. Ethical approval for the study was obtained from the Ethics Unit at the University of Sheffield (Project: 048991), and the collaborators obtained their own ethical approval for data collection.

Subjects were recruited from 45 PD units in the UK, selected from the 72 units reporting to the Renal Registry. Among these units, 20 participated in the International Peritoneal Dialysis Outcomes and Practice Study (PDOPPS). The recruitment process spanned a 30-month inclusion window, and each centre selected 20-30 sequential patients based on its size. The time frame for recruitment is between January and March 2015.

Follow-up assessments were conducted every 3 or 4 months until the end of the study minimum follow-up 12 months using standardized unit and patient questionnaires. The questionnaires, developed by the international PDOPPS team in collaboration with the International Society for PD, captured information on clinical practices, service organisation, and patient characteristics. The baseline patient questionnaire covered 19 domains, including quality of life, depression, satisfaction with care, assistance in performing PD, and infusion and drain pain. Additional survey instruments specific to the UK Catheter Study were also employed.

The dataset includes various patient outcomes of interest, such as the occurrence of catheter-related events following insertion. It also tracks the recovery from such events after undergoing corrective procedures, including successful re-insertion. Patient mortality rates are recorded, and patient-reported outcomes, including health-related quality of life, satisfaction with care, PD assistance, and PD training, are documented.

The initial study aims to examine how patient- and centre-related factors influence the choice of catheter insertion pathways for PD patients. It also investigates the impact of these factors and pathways on patient outcomes, specifically comparing the occurrence of and recovery from PD catheter-related adverse events and mortality between individuals who underwent surgical catheter insertion and those who underwent medical catheter insertion. Additionally, the study aims to evaluate the health economics associated with these pathways.

5.4.2 Variables in the dataset

This dataset contains a large number of files and variables, so the focus of this section is only on the relevant files and variables. The data consists of 1948 variables categorized into five main files: census, patient modality history, catheter insertion worksheet, medical training file, and infection file. Further explanation about the related variables in each file can be found in section 5.5.1.2. Table 5.1 below presents the demographic information of the patients in the dataset.

Table 5.1 Demographics information (Redacted)

This table and its detailed description have been removed from the publicly available version of this thesis to maintain data confidentiality.

5.5 Process mining

The objective of this section is to generate a process model and parameters using process mining. Specifically, we have generated a process model for patients receiving peritoneal dialysis, from catheter insertion to the first PD dialysis, including any infections and hospitalizations. We then compared the time to PD dialysis and the time to infections for patients who received catheter insertion through medical and surgical means, respectively.

The results of the process mining analysis are presented in the form of a process model, which includes detailed visualisations of the flow and activities of the process and the parameters.

5.5.1 Data preparation

In process mining analysis, the preparation phase is a crucial step that aims to gather accurate and reliable data in the appropriate format, in the form of event logs. In this case study, a secondary dataset was obtained from a collaborator, and to ensure its suitability for analysis, it underwent several important steps. These steps included data collection and extraction, selection of relevant variables, data cleaning, verification of event definitions with process experts, combination of multiple events, and creation of event

logs. The thorough checking and formatting of the data allowed for a more precise analysis and reliable results.

5.5.1.1 Data collection and extraction

Data collection and extraction were carried out by obtaining a secondary dataset from a collaborator. We accessed multiple relevant files, including PDOPPS and UK Catheter Study, which were shared through an encrypted private university drive to understand the data structure and variables it contained. These files were provided in a .csv format. After understanding and discussing with the process expert (i.e., a nephrologist), we extracted the necessary data and imported it into the R programming environment for data manipulation. This involved placing the data into data files and standardizing the format.

5.5.1.2 Selection of relevant variables

After gaining an understanding of the dataset, the initial selection of variables was undertaken by the research team and refined in consultation with clinical experts to ensure relevance for conceptual modelling. Out of 1,948 available variables across multiple files, 17 were chosen because they contained the essential information needed to construct an event log: case identifiers, activity labels, and timestamps. These variables were sufficient to support the generation of process maps and parameters for conceptual model development. The remaining variables, such as demographic details or outcome measures, were not directly relevant to representing care pathways and were therefore excluded at this stage.

Table 5.2 Relevant variables

File	Variable	Descriptions
Census	CatheterInsertionDate	This variable contains information about the date when the catheter was installed for peritoneal dialysis.
	FirstPDTrainingDate	This variable captures the date of the first training session that a patient undergoes before starting peritoneal dialysis. It represents the initial education and training provided to the patient to familiarize them with the PD procedure.
	FirstPDDialysisDate	This variable refers to the date when the patient initiates peritoneal dialysis. It marks the start of the patient's regular dialysis sessions using the peritoneal dialysis method.
	LastPDDialysisDate	This variable signifies the most recent or final date on which the patient underwent peritoneal dialysis. It represents the most recent session of PD dialysis recorded in the dataset.
	DeathDate	This variable indicates the date of death of the patient. It records the specific date when the patient passed away.

	ReasonStopped	This variable includes a code, specifically "Code 23: Received Transplant," which serves as a reference to extract the date of a kidney transplant. It denotes that the patient stopped peritoneal dialysis due to receiving a kidney transplant. The date associated with this reason can be extracted for further analysis or tracking purposes.
Patient Modality History	ModalityStartDate	This variable captures the start date for each type of modality the patient received, such as peritoneal dialysis (PD), hemodialysis (HD), or a kidney transplant. It represents the date when the patient began a specific modality as part of their treatment.
	ModalityStopDate	This variable indicates the date when a particular modality was stopped or discontinued for the patient. It signifies the end date of the specific modality, such as PD, hemodialysis, or a kidney transplant.
	Modality type	This variable categorizes the different types of modalities that a patient may have received,such as peritoneal dialysis (PD), hemodialysis(HD), or a kidney

		transplant. It provides information about the specific treatment modality that was administered to the patient.
Catheter Insertion Worksheet (CIW)	A1_Date_AccessProcedure	This variable represents the date when the access procedure for catheter insertion was performed. It signifies the specific date when the procedure to establish access for catheter placement took place.
	D2_InsertionTechnique_Open	This variable indicates the use of an open technique for catheter insertion
	D2_InsertionTechnique_Lap	This variable signifies the use of a laparoscopic technique for catheter insertion
	D2_InsertionTechnique_Perc NonFluro	This variable indicates the use of a percutaneous technique without fluoroscopy for catheter insertion.
	D2_InsertionTechnique_Perc Fluro	This variable suggests the use of a percutaneous technique with the assistance of fluoroscopy for catheter insertion.
Infectio n	EventType	This variable provides information about the type or nature of the infection event

	StartDate	This variable captures the start date of the recorded infection event
Medical Training (MQ Training)	G1_FirstDateTraining	This variable captures the date of the first PD training session for the individual.

5.5.1.3 Data cleaning

Data cleaning was performed to handle missing data, correct errors, and handle extreme data to ensure that the data is accurate and suitable for generating the process map. Table 5.3 shows the types of data errors, and the methods used to handle these errors in this case study.

This case study did not have many missing values because the dataset had good quality data. This could be attributed to the robust and highly effective data collection protocol utilized by our collaborator in previous efforts.

Table 5.3 Data cleaning

Data Issue Category	Data Issue	Solution
Data Outlier	There is 1 ID with extreme start date	Data cleaning by removing it.
Data Entry Error	There are 109 IDs that have a date with modality start date as 01/01/00.	Upon reviewing the data, we requested the data owner to investigate the issue at hand. The data owner has since reported back, stating that the identified IDs belong to patients who

		underwent HD dialysis. However, as our current focus is on PD dialysis, we have decided to separate these IDs and place them in a different group.
Duplicate data issue	There are 7 IDs that have repetitive rows.	To address this concern, by examining each ID individually and resolving the repetitive rows.
Missing values	There are 19 rows with Modality start date with Cell with “.” (missing values)	We reached out to the process owner to inquire about these missing dots. The process owner suspects that the "." (dot) represents a missing value of 0, specifically in cases where the patient started on HD before transitioning to PD. Consequently, we have made the decision to group these patient IDs under the HD category.
Invalid data	There are 13 rows where ModalityStartDate is after ModalityStopDate (ModalityStartDate > ModalityStopDate)	We have made the decision to remove these 13 rows from the dataset.
	There is 1 row with FirstPDDialysisDate> LastPDDialysisDate	We have made the decision to remove this 1 row from the dataset.

Logical inconsistency data	There are 3 rows with LastPDDialysisDate> DeathDate	We adjusted the LastPDDialysisDate to match the DeathDate for these three rows. By making the LastPDDialysisDate equal to the DeathDate, we align the information in the dataset.

5.5.1.4 Naming events and consolidating multiple events.

The dataset used for this case study had clear and easily understandable variable names. Therefore, not much renaming was required, and all names used were taken directly from the existing variables.

In the context of the process mining analysis, peritonitis and exit site infections were named under one event "Perit_Exit" as they were both types of infections that could occur in patients undergoing peritoneal dialysis. These infections shared similar characteristics in terms of their symptoms, treatments, and impact on the patient's overall health. Grouping peritonitis and exit site infection under one event allows for a more efficient and accurate analysis of the process of peritoneal dialysis.

In addition, catheter event and hospitalization were consolidated under one event named "CE_Hospitalization." A catheter event, such as insertion or removal of a catheter, can often lead to hospitalization for patients undergoing peritoneal dialysis due to complications or infections that may arise. By consolidating events, a more comprehensive view of the process is obtained, including the frequency and impact of catheter-related complications and the resulting hospitalization.

5.5.1.5 Simplification of events

In this case study, we have implemented several event simplifications due to the complexity of the patient flow process. Modality changes represent a critical aspect of patient trajectories, as they are likely correlated with infection development. Patients with end-stage renal failure often experience multiple transitions between peritoneal dialysis

(PD) and hemodialysis (HD), both in hospital and at home, along with hospital admissions due to catheter infections and re-insertions. These transitions are reflected in Figures 5.4, 5.5, and later in Figure 6.7, where the process variations are evident.

To manage this complexity, we considered only the first modality switch, as our calculations indicate that, on average, patients experience just one switch. Table 5.5 presents the minimum and maximum number of modality switches observed in the dataset, along with their corresponding distributions.

Table 5.4 Modality switches

Metric	Value
Min.	1 switch of modality
Max.	8 switches of modalities
Median.	1 switch of modality
Mean.	Approximately 1.4 switches

The table highlights key findings:

Minimum modality switches: The lowest complexity level recorded is a single switch.

Maximum modality switches: The highest number of switches observed is 8, representing a rare but extreme case.

Median number of switches: A median of 1 indicates that the majority of patients experience only one modality switch.

Mean number of switches: The mean, approximately 1.4, suggests that most patients undergo minimal modality changes.

To further contextualize this, the distribution of modality switches within the dataset is as follows:

- 77% of patients had 1 modality switch
- 17% had 2 switches
- 4.5% had 3 switches
- 1.4% had 4 switches
- 0.1% had up to 8 switches

Although a small subset of patients experienced multiple modality transitions, these cases were relatively uncommon. We acknowledge that, in real-life clinical settings, patients may cycle between PD and HD multiple times, influenced by complications such as catheter infections requiring hospital admission and re-insertion. However, our decision to focus on the first switch was based on the need to balance analytical clarity with clinical relevance.

Given that the primary objective of this study is to examine the time to PD initiation for two groups with different catheter insertion techniques (surgical and medical) and to assess the time to infection, including higher numbers of modality switches would introduce unnecessary complexity. Incorporating extreme cases, such as the patient with eight modality switches, could add variability that detracts from the primary objective, making it more challenging to identify meaningful patterns in PD initiation and time to infection.

Additionally, to maintain clarity in process mapping, we have streamlined the representation of infection events. Specifically, for peritonitis and exit site infections, only the earliest recorded infection date was used in constructing the process map. This approach was also applied to catheter exchanges (CE) and hospitalizations, where only the earliest event was considered. While we acknowledge that recurrent infections and hospitalizations are significant clinical concerns, this simplification ensures a more focused analysis of the initial transition to PD and its associated risks.

5.5.1.6 Generating an Event Log from Process Data

Generating an event log from process data involves creating a data storage structure that contains relevant information about the process, such as the Case ID, activities, and date the activity was performed. This structure is typically formatted in a standard.csv file, which is the minimum required information for process mining. Figure 5.2 shows example of the event log for the kidney dataset.

ID	Activity	Date
105002082	CatheterInsertionDate	30/06/2016
105002082	FirstPDTrainingDate	08/08/2016
105002082	FirstPDDialysisDate	09/08/2016
105002082	LastPDDialysisDate	16/12/2017
105002082	CentreHD_Start	16/02/2017
105004026	CatheterInsertionDate	24/08/2016
105004026	FirstPDTrainingDate	20/09/2016
105004026	FirstPDDialysisDate	22/09/2016
105004026	LastPDDialysisDate	25/04/2017
105004026	CE_hosp	28/10/2016
105004026	CentreHD_Start	10/02/2017
105004026	CentreHDStop	21/04/2017

Figure 5.2 Example of event log for Peritoneal Dialysis

Once the necessary data is entered into the event log format, it is ready to be loaded into process mining software for analysis. To generate an event log, the process data is first collected and reviewed to ensure accuracy and completeness. Any inconsistencies or errors in the data were identified and resolved before the data is entered into the event log format.

5.5.2 Process discovery

In this phase of the study, our primary focus was to conduct process discovery and generate an accurate and informative process model using the event logs. The objective was to gain a deeper understanding of the pathways followed by PD dialysis patients, from the initial catheter insertion to the subsequent steps involved in obtaining PD dialysis. Additionally, we aimed to capture any potential infections resulting from

catheter events and the possibility of hospitalizations due to complications that might require modality changes.

To achieve this, we utilised a process management software called Disco. We loaded the event logs into the software and employed various analysis techniques to identify the flow and behaviour of the process. In this case study, we directly applied the learning from the algorithm comparison undertaken in Chapter 4 and used the Fuzzy Miner algorithm, as it was found to be the most effective for simplifying complex healthcare pathways while retaining clinically meaningful detail. The resulting process model provided a clear and visually intuitive representation of the PD patient pathways, infections, and potential modality changes. This visual representation allowed us to examine the sequential order of activities, understand the interdependencies between these activities, and assess the time taken to complete each activity.

Throughout the analysis, we focused on a set of 12 key activities that were prominent in the study. These activities included 'Catheter insertion', 'First PD Training', 'First PD dialysis', 'Last PD Dialysis', 'Perit_Exit', 'CE_hosp', 'CentreHD', 'HomeHD', 'No Dialysis', 'Transplant', and 'HybridPD+HD'. Furthermore, we made use of specific start and stop events for some activities, such as 'CentreHD_Start' and 'CentreHD_Stop', in order to capture the duration of patient involvement within specific modalities.

The process mining analysis revealed common patterns in the PD patient pathways. Typically, the pathways originated with 'Catheter Insertion', followed by 'First PD Training', 'First PD dialysis', 'Last PD Dialysis', and subsequently, the occurrence of 'Infections' indicated by 'Perit_Exit' or 'CE_hosp'. Notably, modality changes were observed in the pathways, which encompassed various possibilities such as 'CentreHD Dialysis', 'HomeHD Dialysis', 'No Dialysis', 'Transplant', or a hybrid approach involving a combination of PD and HD dialysis.

Clinically, patients undergoing peritoneal dialysis (PD) initiate the process with the insertion of a PD catheter through a medical or surgical procedure, typically performed by a surgeon or physician. Subsequent to catheter placement, patients undergo comprehensive education and training on managing the PD catheter and performing dialysis exchanges. Following this, patients begin PD therapy at specialized centres or clinics. Complications, such as infections, may arise during PD treatment, necessitating

hospitalization or adjustments in catheter care. If complications persist, consideration may be given to transitioning to alternative treatment modalities, such as hemodialysis or kidney transplantation. Unfortunately, in some cases, persistent complications or a decline in health may lead to a deterioration in the patient's well-being and potential death.

Moreover, it is important to highlight that by applying a 10% frequency threshold for abstraction in the process model, we ensured that the model effectively represented the clinical visualisation of PD patient pathways. This approach struck a balance between capturing the necessary level of detail and avoiding unnecessary complexity. By doing so, we provided a clear yet clinically meaningful overview of PD patient pathways while retaining both the most common trajectories and those clinically significant to our study objectives. This abstraction enabled us to understand the overall flow and key treatment steps, including modality transitions, without overwhelming complexity.

Secondly, the validation through clinical validation and verification further strengthened the reliability and accuracy of the process model. By comparing the process model with real-world clinical data and insights, we ensured that the model accurately represented the observed patient pathways. This validation process involved experts in the field, i.e a nephrologist who confirm the alignment of the model with the actual clinical practices and experiences. The validation process provided confidence in the accuracy and reliability of the process model and its ability to effectively represent the clinical visualisation of the PD patient pathways.

Throughout the analysis, we discovered and identified 50 variations in patterns within the PD dialysis patient pathways demonstrated the complexity and diversity of the treatment process. By capturing these variations, we gained a comprehensive understanding of the different pathways patients could take, considering factors such as catheter insertion, infections, and potential modality changes. This comprehensive understanding allowed us to explore the intricacies and nuances of PD dialysis, providing valuable insights into the various factors and potential challenges that patients may encounter during their treatment journey. The identification of these patterns and variations provided a holistic view of the patient pathways, facilitating a deeper understanding of PD dialysis and informing potential improvements or interventions to enhance patient outcomes.

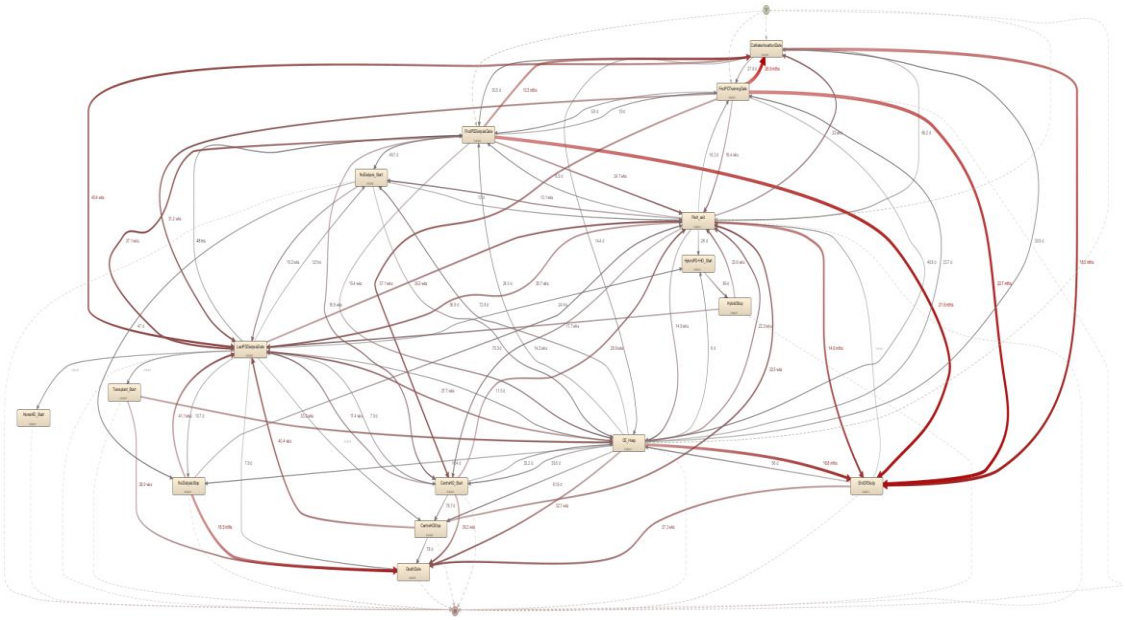


Figure 5.3 Initial process model for overall PD patients

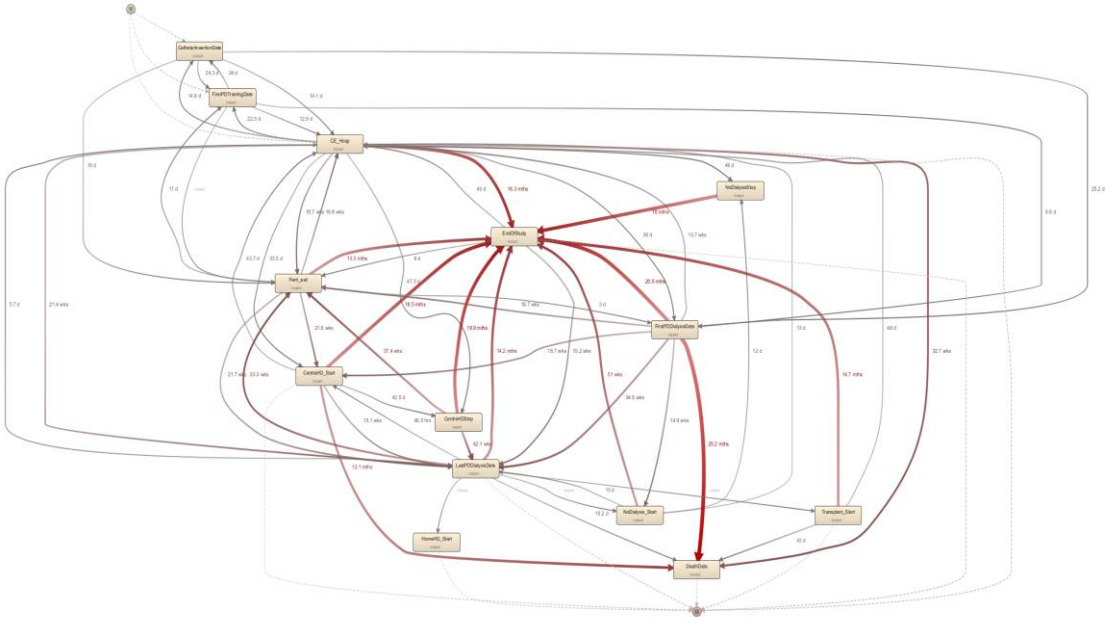


Figure 5.4 Initial process model for PD patients with medical catheter insertion

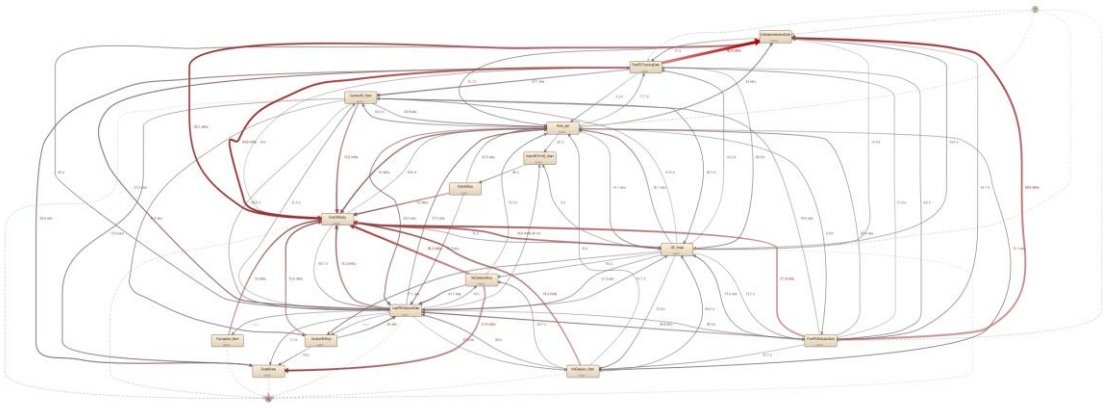


Figure 5.5 Initial process model for PD patients with surgical catheter insertion

5.5.3 Process analyses

In process analyses, we conduct checking to ensure alignment with the actual processes recorded in the event logs. Our performance analysis involves evaluating the processes represented in the models, with a focus on metrics such as time durations. We employed expert validation to discern the underlying reasons behind any deviations detected in the process models. This collaborative approach with subject matter experts enhances the reliability of our findings, providing a comprehensive understanding of the intricacies within the processes. In the next subsections, we delve into the specifics of how we conducted the assessment of the generated process models.

5.5.3.1 Simplifying the process model

Simplifying a complex process model in process mining is a collaborative effort where we work with experts in PD Dialysis, i.e a nephrologist. The process begins with a process model that has evolved from the analysis of event logs, featuring complex pathways. We prioritize critical sequences and frequently occurring steps, allowing us to simplify the model by emphasizing the most impactful elements.

Next, we remove unnecessary pathways, retaining only the most critical decision-making points that significantly influence the overall process flow. We further simplify by

aggregating similar events based on expert advice into higher-level representations. This approach provides a more concise view of the main process without delving into unnecessary detail.

Validation of the simplified model is a critical step for us. We strive to strike a balance between simplicity and maintaining the integrity of the representation, and we validate this with experts.

5.5.3.2 Assessment of the process model

We conducted an assessment of the process model generated by process mining to gain a deeper understanding of the PD pathway. To evaluate the process models, we assessed them based on three key criteria: fitness, precision and generalization. The fitness criterion measured the closeness of the process models to the real PD pathway and also considered any variations and deviations. To measure fitness, we compared the process models with the views of professionals in the health care field, namely, nephrologists to determine whether the model accurately reflected their experiences.

The criterion for precision judged the level of detail and the correctness of the process models. This included an evaluation of how accurately the models represented the particular complexities of the PD pathway with any potential variations or deviations. To measure precision, we performed a detail analysis of the level of detail that was included in these models and how well they represented the specific steps and activities of the PD pathway. These steps included everything from the introduction of the catheter and the PD training to the first PD dialysis, infections, and the risk of changing the modality. We were focused mainly on the identification and ordering of the key stages of PD patient pathways within these models. The initial process model is shown in figure 5.4 and figure 5.5.

In addition, we evaluated the models in terms of depicting the connections and relationships that exist between different stages of the PD pathway. This included the assessment of the appropriateness of the process mining technique to reflect the dependencies and interactions between various steps, such as modality changes and infections.

Generalization is the process of reducing the level of detail in an event data set in order to have a more understandable and manageable representation of a process. It includes summarization of the data, without losing any important information, patterns, and structure. This study employs a 10% data abstraction strategy, which is based on the inputs provided by clinical expert. This approach guarantees the preservation of key information about the peritoneal dialysis PD patient pathways' intricacies while reducing the amount of data. The abstraction process is meant to abstract the key factors that are necessary to represent the path of a PD patient through PD training, infections and modality changes. The attributes and elements needed to represent patient pathways accurately should be determined by clinical experts who are knowledgeable about the complexities of peritoneal dialysis management. When data abstraction is used with clinical knowledge, the generalized process model that emerges is an effective reflection of PD patient pathways.

Our evaluation of the process models allowed us to distinguish between the PD pathways for the medical and surgical catheter insertion subgroups.

5.5.3.3 Subgroup analysis of different catheter insertion techniques in process mining.

We sub-grouped the patients in the initial overall process model into two categories: medical catheter insertion and surgical catheter insertion. This allowed us to generate two separate process models to achieve the case study objectives.

The first objective of the assessment was to compare the time it took for patients to begin peritoneal dialysis after receiving medical or surgical catheter insertion. To achieve this, we analysed the process models generated for each subgroup to identify any differences in the time it took for patients to progress through the steps involved in the PD pathway. We focused on measuring the time to event, which in this case was the time it took for patients to begin peritoneal dialysis after receiving catheter insertion. We compared the time to event between the medical and surgical catheter insertion subgroups to identify any significant differences. By identifying any differences, we could also develop strategies for improving patient outcomes.

The second objective of the assessment was to compare the time to infection after first PD Dialysis in patients who received medical or surgical catheter insertion. We analysed the process models generated for each subgroup to identify any variations in the time it took for patients to develop infections.

The results of the assessment revealed significant differences in the time it took for patients to begin peritoneal dialysis after receiving medical or surgical catheter insertion. Specifically, the process model for patients who received medical catheter insertion showed a shorter time to PD dialysis compared to the process model for patients who received surgical catheter insertion.

In terms of the second objective, the results of the assessment showed a significant difference in the time to infection after PD dialysis for patients who received medical or surgical catheter insertion. Specifically, according to the process model, patients who received medical catheter insertion developed infections sooner after catheter insertion compared to patients who received surgical catheter insertion.

It is important to note, however, that a shorter time to infection may indicate a higher risk of infection, as it suggests that patients who received medical catheter insertion developed infections more quickly. The risk of infection is influenced by various factors, including patient characteristics, surgical techniques, and adherence to infection prevention protocols. Nevertheless, the results of this assessment provide valuable insights for developing strategies to mitigate infection risks in patients undergoing surgical catheter insertion. These strategies may include improving catheter insertion techniques and enhancing infection prevention practices.

Table 5.5 Comparison of time-to event between medical and surgical catheter insertion methods

Events	Mean time for Medical Catheter Insertion (days)	Mean time for Surgical Catheter Insertion (days)
Catheter Insertion > First PD Training	24.3	31
First PD Training > First PD Dialysis	6.6	4.8
First PD Dialysis > Perit_Exit	130.9	177.8
Perit_Exit > Last PD Dialysis	151.9	205.1
First PD Dialysis > Last PD Dialysis	241.5	205.1
Last PD Dialysis > Centre HD Start	1.9	12.2
Centre HD Start > Centre HD Stop	42.5	93.8
First PD Dialysis > CE_hospital	95.5	107.8

Table 5.5 shows the time between different events in the PD pathway, along with the mean time in days for each event for each catheter insertion group. The table provides an overview of the time it takes for patients to progress through the PD pathway and allows for a comparison between patients who received medical or surgical catheter insertion.

Time to PD dialysis: The time between Catheter Insertion and First PD Dialysis is an important indicator of how quickly patients are able to progress to PD dialysis. Patients who received surgical catheter insertion had a shorter mean time (31.0 days) between Catheter Insertion and First PD Training compared to patients who received medical catheter insertion (24.3 days). However, patients who received medical catheter insertion

had a shorter mean time (6.6 days) between First PD Training and First PD Dialysis compared to patients who received surgical catheter insertion (4.8 days). This suggests that medical catheter insertion may be associated with a faster time to PD dialysis compared to surgical catheter insertion.

Time to infection Perit_Exit: Patients who received medical catheter insertion had a shorter mean time (130.9 days) between First PD Dialysis and Perit_Exit compared to patients who received surgical catheter insertion (177.8 days). This suggests that medical catheter insertion may be associated with reduced time to infection compared to surgical catheter insertion.

Time to infection CE_hospital: Patients who received medical catheter insertion had a shorter mean time (95.5 days) from First PD Dialysis to CE_hospital compared to patients who received surgical catheter insertion (107.8 days).

5.5.4 Final process map

Figure 5.6 and Figure 5.7 depicts the final process map for PD patients pathways having the medical and surgical catheter insertion, along with the associated parameters.

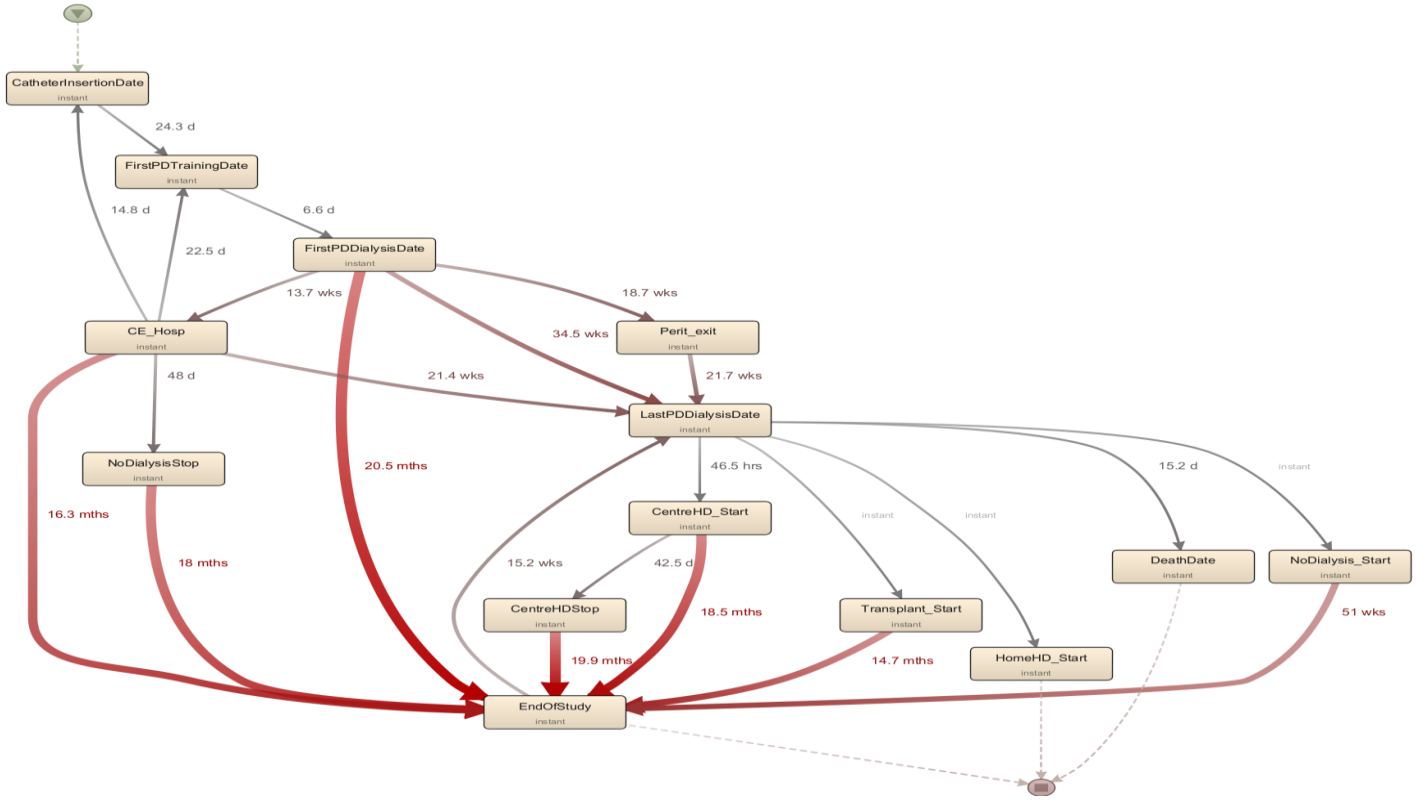


Figure 5.6 Final process model for PD patients with medical catheter insertion

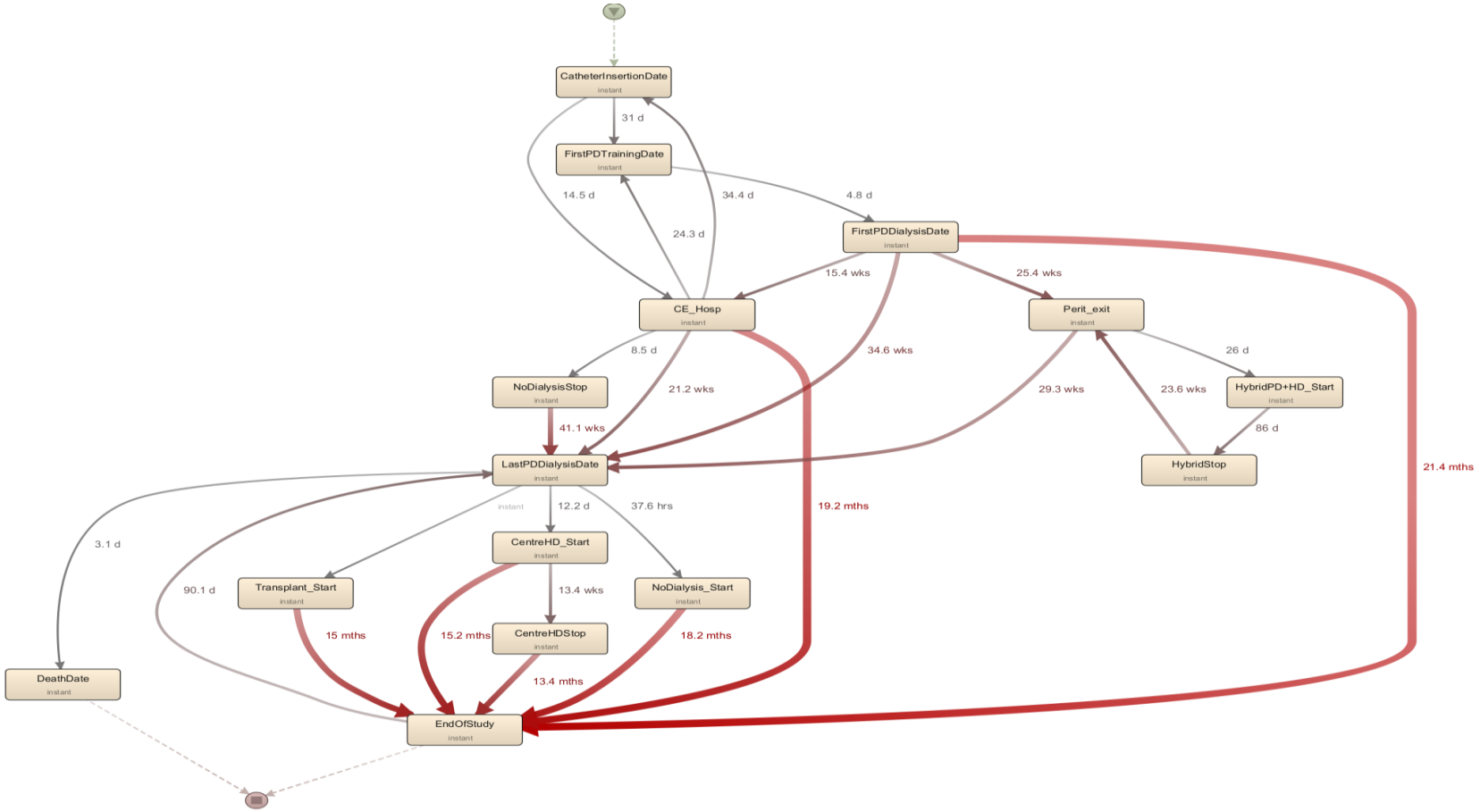


Figure 5.7 Final process model for PD patients with surgical catheter insertion

5.6 Results Interpretation and Clinical Significance

We found that patients who received medical catheter insertion had a shorter time to peritoneal dialysis. Medical catheter insertion techniques, such as percutaneous, are recognised for their less invasive nature compared to surgical methods. These procedures often entail a needle puncture through the abdominal wall to position the catheter into the peritoneal cavity. In contrast to surgical insertion methods that may involve more tissue trauma and disruption, medical techniques often result in less tissue damage. Due to their minimally invasive approach, patients undergoing these techniques may encounter quicker recovery times and a reduced likelihood of complications. Consequently, the healing process at the catheter site is generally expedited due to the typically faster healing time at the catheter insertion site. This accelerated healing enables patients to initiate peritoneal dialysis sooner after the catheter placement, contributing to a reduced wait time before starting PD treatments. These advantages contribute to shortening the interval before commencing peritoneal dialysis (PD), facilitating a smoother transition to dialysis for patients.

With less discomfort and a shorter recovery duration, patient who opt for medical techniques may find themselves better positioned to commence peritoneal dialysis earlier. This expedited recovery plays a significant role in facilitating a swift transition to PD treatment.

Availability and scheduling at healthcare facilities also play a crucial role in determining the time to commence peritoneal dialysis. Medical catheter insertion procedures might be more accessible or have shorter waiting times for scheduling compared to surgical procedures in certain healthcare settings. This accessibility could significantly contribute to reducing the interval before initiating peritoneal dialysis for patients who opt for medical catheter insertion methods.

The choice of catheter insertion technique can also be influenced by individual patient factors. Variables such as overall health status, anatomical considerations, and the presence of comorbidities may impact the selection of a catheter insertion method. Patients with fewer comorbidities or a more suitable abdominal anatomy for medical catheter insertion may experience a quicker transition to peritoneal dialysis, further contributing to the variation in the time interval before starting PD.

Our study revealed that patients who underwent medical catheter insertion were less likely to develop infections compared to those who received surgical catheter insertion. However, when infections did occur in medical catheter insertion patients, they developed more quickly after the procedure.

While surgical procedures are often considered more controlled and precise, they do not necessarily lead to better infection outcomes. In contrast, our findings suggest that patients undergoing medical catheter insertion may require closer post-procedure monitoring, as early detection of infections could enable prompt intervention and reduce the severity and duration of complications.

The findings highlight the importance of monitoring and early detection in managing infections post-catheter insertion, advocating for more attentive care for patients who undergo medical catheter insertion to promptly identify and manage any potential infections that may arise.

5.7 Value of Process Mining in Conceptual Model Development (UK Kidney Case Study)

1.Revealing Clinical Pathway Variants Based on Insertion Technique

In Chapter 5, process mining was instrumental in uncovering and quantifying the real-world variation in patient pathways based on catheter insertion technique medical versus surgical. Although both techniques aimed to initiate peritoneal dialysis, they followed structurally distinct trajectories, each with unique transitions and system dynamics. These differences were not based on clinical assumptions but extracted directly from event log data and formed the basis for building a more representative and evidence-based DES conceptual model.

For medical catheter insertion (Figure 5.6), the pathway tended to be more streamlined. Patients moved from CatheterInsertionDate to FirstPDDialysisDate with fewer intermediate transitions. Many patients transitioned directly into routine dialysis states or exited the pathway without major interruptions. The model also captured the empirical sequence of events such as movement to CE_Hosp and common exit points (e.g., flows to EndOfStudy), providing a clear view of the operational flow for this group.

In contrast, the surgical insertion pathway (Figure 5.7) was more complex and featured a wider variety of routes. The pathway included detours such as temporary transitions to HybridPD+HD_Start with a downstream HybridStop that then connects to Perit_exit, additional operational diversions related to surgical logistics, and more re-entry routes into CentreHD_Start or NoDialysis_Start states. A notable feature is the prominent direct transition from FirstPDDialysisDate to EndOfStudy; HybridStop does not connect directly to EndOfStudy in the final model.

While clinician input was essential for defining expected care pathways, process mining complemented this by exposing variations that clinicians had not routinely described. These included return flows, dropout routes, and modality switches, all of which were operationally significant. Incorporating these patterns ensured the DES conceptual model reflected the heterogeneity of patient experiences, rather than a simplified single pathway.

Process mining identified and quantified pathway differences between medical and surgical insertion techniques, highlighting deviations such as modality switches and early dropouts. These findings informed the conceptual model structure, ensuring they were taken into account during its development.

As discussed in Section 6.6, this showed how process mining can complement expert-driven approaches by capturing less common but important variations that might not otherwise be emphasised. It is unlikely that such pathway variants would have been fully identified through traditional DES model development, where pathway structures are usually based on expert consultation and summary data rather than detailed event logs.

2. Evidence-Based Simplification and Pathway Filtering

One of the key challenges in developing a DES conceptual model from real-world healthcare data is managing the complexity and heterogeneity of patient journeys. In the UK kidney case study, the raw event log derived from routine electronic health records contained a large number of unique patient traces, each reflecting a different sequence of events surrounding catheter insertion and subsequent follow-up care. These included standard steps as well as rare deviations, rework loops, and occasional inconsistencies, resulting in an initial process model that was highly complex and difficult to interpret directly.

To address this, process mining provided a structured and empirical method for simplifying the model while preserving clinical relevance. Using the variant analysis and frequency filtering features in Disco, patient traces were grouped by behavioural similarity, enabling the identification of dominant pathway patterns.

This analysis captured all pathway variants recorded in the event log, reflecting both common workflows related to medical or surgical insertion and less frequent deviations in follow-up care. Rather than arbitrarily selecting or generalising patient flows, this approach enabled pathway selection to be grounded in observed evidence

Each retained variant was both frequent and clinically meaningful. For example:

- Variant showed a streamlined medical insertion followed by prompt training and dialysis initiation.
- Variant reflected a surgical insertion followed by a longer period before PDdialysis initiation, with additional recovery time
- Variant involved patients undergoing multiple rounds of pre-dialysis assessments due to post-procedure complications before progressing to dialysis.

These variants were retained in the conceptual model because they represented key structural and operational differences between patient groups. In contrast, infrequent or clinically inconsistent sequences such as those caused by administrative anomalies or one-off delays were excluded, maintaining focus and clarity.

This evidence-based simplification supported more transparent modelling decisions. Process mining made it possible to articulate why certain pathways were selected and how they represented the broader care process, ensuring the conceptual model remained tractable while still analytically robust. In this way, process mining helped strike a balance: retaining sufficient detail to represent key variations in patient care, while simplifying the structure enough for practical modelling and stakeholder engagement.

Process mining identified representative pathways directly from the data, showing which variants were both frequent and meaningful. This provided evidence-based basis for simplification, while expert knowledge complemented this by helping to confirm which pathways were most clinically appropriate.

As discussed in Section 6.6, this combination of empirical evidence and expert input supported a more transparent abstraction process, preserving essential variation while keeping the model practical for simulation. This offered a more explicit, data-driven basis for simplification than is usually available in traditional DES model development.

3. Determining Level of Detail from Observed Pathway Variations

In this case study, process mining was used to support the development of a conceptual model for peritoneal dialysis (PD) pathways by providing empirical insights into actual patient flows, rather than relying solely on clinical knowledge or expert-derived "standard" pathways. While clinician input was essential for defining expected care processes, process mining complemented this by offering objective, data-driven evidence on the timing, variability, and structure of patient pathways, highlighting clear differences between medical and surgical catheter insertion groups.

A key challenge in conceptual modelling particularly when used to inform future DES is deciding how much detail to include. Process mining addressed this by revealing where detailed representation was clinically justified and where simplification was appropriate. For example, the event logs showed significant delays in the surgical group between CatheterInsertionDate and FirstPDTrainingDate (mean: 31 days), and between FirstPDTrainingDate and FirstPDDialysis (mean: 4.8 days), linked to post-operative recovery. These delays were longer and more variable than in the medical group, insights that would likely be underestimated in expert-only models. As a result, these stages were retained as distinct elements in the conceptual model to reflect their impact on access to treatment.

In contrast, for medical insertions, First PD Training and First PD Dialysis occurred in relatively close succession (mean gap: 6.6 days). Although temporally close, these steps were retained as separate elements due to their clinical relevance. This ensured the model captured essential distinctions in patient care rather than merging steps simply because they were close in time.

Process mining also revealed rework loops and multiple pre-dialysis interactions in surgical cases such as clinical reassessments indicating that patient pathways were not strictly linear. These loops were not explicitly identified in initial clinician discussions but were evident in the event log, and they were incorporated into the conceptual model to better reflect real patient journey.

Process mining quantified delays, variability, and rework in surgical pathways and confirmed stable sequencing in medical pathways. These insights guided which steps were retained as distinct elements and which could be simplified, ensuring that the conceptual model remained clinically relevant and representative of observed practice.

As discussed in Section 6.6, combining empirical evidence from process mining with expert input supported more balanced decisions on the appropriate level of detail, helping the model remain both interpretable and analytically useful. Traditional DES model development, which depends mainly on expert consultation, would have been less likely to highlight these discrepancies between expected and observed care pathways.

4. Data-Driven Cross-Checking and Refinement of Expert Knowledge

In the UK kidney case study, process mining complemented clinical expertise by cross-checking expert input against observed event data, helping to refine and enrich the conceptual model of peritoneal dialysis (PD) services. Initially, the clinical team described the PD care process as relatively standardised, particularly for the period following catheter insertion. For example, it was generally expected that patients would progress quickly from catheter insertion to training and then to dialysis initiation, with minimal variation between cases. These expectations influenced the early structure of the conceptual model, which initially reflected a simplified and linear flow of care.

However, process mining revealed a more complex and variable reality. By analysing timestamped event data for both surgical and medical catheter insertions, the analysis highlighted differences in timing and variability that were not apparent in the initial clinical description.

In the medical insertion group, patients tended to transition more quickly between CatheterInsertion, First PD Training, and First PD Dialysis. The event data showed relatively short and consistent time gaps, with an average of 24.3 days from insertion to training and 6.6 days from training to dialysis. This pattern confirmed a more streamlined pathway for patients undergoing medical catheter insertion.

By contrast, in the surgical insertion group, the average time from catheter insertion to training was longer, at 31 days, and displayed greater variability. These delays were often linked to factors such as clinical coordination, post-operative recovery, or wound care considerations. This highlighted a more complex trajectory for surgical patients, one that required explicit representation in the conceptual model.

The findings prompted structural adjustments: stages with consistent delays or variability were retained as distinct elements, ensuring the model reflected differences between surgical and medical pathways rather than assuming a uniform trajectory. Process mining also revealed pathway variations such as rework loops and transitions into other modalities that were not initially emphasised in clinical discussions. These included episodes of infection, catheter-related events, or shifts to haemodialysis, which were incorporated into the conceptual model. Their inclusion ensured the model captured both common pathways and less frequent but clinically important scenarios.

Process mining provided an evidence-based view of patient pathways and was then cross-checked with clinical input, confirming areas of alignment while also revealing differences in delays, variability, and modality changes. This allowed the conceptual model to incorporate both observed data and expert knowledge, improving its accuracy and representativeness.

As discussed in Section 6.6, combining empirical evidence from process mining with expert review offered a more complete view of patient pathways, helping the model to reflect both expected flows and observed variability. In traditional DES model development, which is generally based on expert consultation, such differences between expected and observed pathways would have been less likely to be highlighted.

5.8 Discussion

In the UK Kidney case study dataset, there are numerous variables compared to the Mexican Breast Cancer dataset. The storage and management of the dataset are more structured, stored in different files, introducing some complexity to data extraction. The dataset includes a data dictionary that facilitates understanding of variables, but detailed explanations from the data owner, in this case, the nephrologist, are needed. These explanations should cover the selection process for relevant variables and why they are essential. Additionally, collaboration with the data owner is crucial for understanding the data structure and extraction methods.

Initially, data profiling was conducted on all variables to check for missing data. This was learned from the experience of the first case study with the Mexican Breast Cancer dataset. The profiling revealed that the handling and storage of data in this dataset were well-structured, reducing errors and missing data. This enhanced the data quality, serving as input for conceptual model development and parameterisation for modelling purposes. The data protocol used proved effective in ensuring accurate data storage and entry, reducing errors during data cleaning. With good-quality data, data cleaning became easier and possibly required fewer iterations.

The dataset mostly contains complete data compared to the Mexican Breast Cancer dataset, which had many incomplete entries. This completeness advantage speeds up the data cleaning process, reducing the need for extensive reiteration.

In selecting variables, expert advice was sought, especially for cases like the different handling of transplant activity dates for kidney patients stored in the data structure. Advice from experts, such as nephrologists, was crucial in understanding the structure of the datasets and the relationships between variables in different files. Unlike the Mexican dataset, where all variables were stored in a single file, this dataset has two main files: PDOPPS and UKCath. The UKCath dataset is nested within PDOPPS. Less time was spent on data cleaning for this dataset due to its good data quality, focusing more on collaboration with nephrologists to understand the dataset's structure and the relationships between variables in different files, which, when compared to the Mexican dataset. There

are the same variables that appear in different files. Therefore, we need advice from the data owner, a process expert, to determine which files' variables should be used in such cases.

Understanding the data structure for data extraction heavily relies on advice from experts, especially when dealing with numerous files and complex data storage structures. Because this dataset is well-organized, data cleaning does not take much time. Naming variables and simplification are also quite straightforward because the dataset uses standard data naming that is easy to understand, and we can name activities using near-label names, as it is already clear.

We have simplified the event Peri_Exit and CE_Hops, as mentioned in section 5.52. This is based on suggestions from nephrologists, our collaborators. The aim is to make the patient's journey, specifically time to infection, clearer by reducing complexity in the conceptual model.This simplification was done by combining both Perit and Exit site events into Perit_Exit, and catheter-related events and hospitalizations into CE_Hosp. Many efforts were made to analyse clinical pattern trajectories for patients undergoing PD treatment to highlight differences between two patient groups, those with medical and surgical catheter insertions. Our collaborating nephrologist emphasized differences in time to PD dialysis, time to infection, and variations in pathways for patients with medical and surgical catheter insertions.

Many discussions took place with our collaborator nephrologist to find a clear way to represent these differences in the conceptual model. For example, initially, only one event captured PD dialysis. However, our collaborator wanted to know the time between FirstPDDialysis to LastPDDialysis, so we created two events to capture the time between them and to show whether infection occurred between FirstPDDialysis and LastPDDialysis or beyond. Additionally, we wanted to know how long patients spent in centre HD before switching modality back to PD dialysis. Furthermore, we created another event, EndOfStudy, to show if there were patients still on PDDialysis at the cutoff date for this study.

Selecting the level of data abstraction for this dataset was quite challenging to present appropriate clinical pathways while not missing any important clinical insights. We tested different levels of data abstraction, and after several tests and consultations with experts, we agreed that a 10% level of data abstraction can display sufficient clinical understanding for the pathways.

For future work, leveraging advanced machine learning techniques and data visualisation techniques can improve data pre-processing and cleaning. Opportunities lie within the dataset, which has good data, providing a more specific process model for better clinical understanding, potentially revealing hidden patterns for better-informed decision-making. When the parameters and conceptual model are correct, there is potential for prediction capabilities, allowing the forecast of potential process outcomes.

Finally, we want to highlight the importance of validating the results for the conceptual model and parameters with experts and existing models. Expert validation provides insights into the clinical relevance and appropriateness of the chosen variables, ensuring that the model aligns with real-world clinical scenarios.

CHAPTER 6

COMPARISON OF TRADITIONAL SIMULATION MODELLING APPROACH VS (FULLY AUTOMATED AND EXPERT DRIVEN) PROCESS MINING APPROACHES

6.1 Chapter Overview

We conducted a comparative analysis of three different approaches for developing discrete event simulation models: traditional simulation modelling, expert-driven process mining, and fully automated process mining. The expert driven process mining approach in chapter 4 was compared to a fully automated process mining approach using the same dataset, while the expert driven process mining approach in chapter 5 was compared against conceptual model and parameters derived using traditional approaches in DES modelling (without process mining). The comparison encompassed the results of process mining (i.e. conceptual models, pathways, and parameters) as well as the time/resources required and interpretability. Finally, we implemented a fully automated process mining approach on a big dataset to understand opportunities and challenges.

6.2 Introduction to Approaches

The section compares different methodologies: Traditional Simulation Modelling and Fully Automated and Expert-Driven Process Mining Approaches. Traditional Simulation Modelling relies on established methods, potentially involving manual inputs, while the latter employs modern automation and expert-driven techniques. The section sets the stage for a detailed comparison, aiming to evaluate opportunities and challenges associated with these approaches.

6.2.1 Traditional Simulation Modelling Approach

Traditional discrete event simulation (DES) is a modelling technique widely applied in various domains, including healthcare, to examine complex systems and processes. This method primarily relies on expert knowledge to develop the simulation model. The following gives the steps involved in structure a traditional discrete event simulation model.

The first stage of the design, which is Problem Identification and Scope Definition primarily focuses on defining that research question to be answered by use of simulation. At the same time, this stage includes defining the scope and limits of the simulation model, more specifically stating which parts of healthcare system are going to be simulated. After this, the Gathering Requirements phase requires detailed elicitation of relevant data, stakeholder requirements and information necessary for development of an authentic simulation model that can reflect reality. Specification of the necessary variables and sources of data makes it possible to conduct a simulation based on real-world components, thus increasing its utility.

The Model Conceptualization and Design step involve stakeholders in a collaborative manner to create the schematic framework of the simulation. This stage involves the identification and definition of entities, events, and activities within system hence forming a basis for further development of the simulation model.

From here, the aspect of Parameterisation and Assumption Setting entails coming up with parameters as well as assuming based on data availability, literature reviews or expert opinions. These parameters determine how the simulation behaves, and assumptions complement any missing data that render it suitable for application in a particular healthcare setting. Moving to this simulation stage, Model Development; the theoretical design now becomes a model that is functioning using requisite software or programming languages. A traditional DSS models a system with the help of discrete events that take place at given points in time thus giving an elaborate definition on how such systems should function.

One of the important steps is Validation and Testing, this step ensures that the model mimics a real healthcare system adequately, and hence it guarantees simulation's validity. Lastly, Refinement as well Iteration constitutes a continuous process that is grounded on validation outcomes and stakeholder's complaints. Changes are made to the simulation model over a series of iterations for improvement on accuracy and relevance. This is a continual process of improvement that helps to ensure close fit with observations from the real world and effectiveness at addressing the research question for which it was developed.

Table 6.1: Steps in traditional simulation modelling approaches.

Step	Description
1.Problem Identification and Scope Definition	Define the research question and establish the scope and boundaries of the simulation model.
2.Gathering Requirements	Collect necessary data, stakeholder needs, and relevant information for modelling.
3.Model Conceptualization and Design	Collaboratively design the conceptual framework, identifying entities, events, and activities within the healthcare system.
4.Parameterisation and Assumption Setting	Set parameters and assumptions based on available data, literature, and expert opinions.
5. Model Development	Translate the conceptual design into a functioning simulation using appropriate software or programming languages.

6.Validation and Testing	Rigorously test and validate the simulation model against real-world data or observed behaviours to ensure accuracy.
7.Refinement and Iteration	Iterate on the model based on validation results and stakeholder feedback to enhance accuracy and relevance.

6.2.2 Expert Driven Process Mining Approach

Expert-driven process mining approach involves the active participation of subject matter experts in conjunction with process mining techniques to create simulation models. This approach was implemented in the case studies in Chapters 4 and 5.

The first step is to determine objectives for integrating process mining with DES modelling, which sets the ground for each of the following stages. A clear protocol is setup in the data preparation stage for collection and processing of information. This involves collection of suitable event log data from different sources, carrying out data profiling to determine characteristics of the data and implementing processes for cleaning the collected data so as to ensure that it is up-to-standard. Missing value imputation is conducted as appropriate with support from domain experts. This is followed by selection of the relevant variables, and formatting data for feeding into analysis tools with checks on timestamp granularity in order to ascertain consistency.

Conceptual modelling converts the event log into a structured form, with key components being ID's, activity labels and timestamps. Suitable process mining algorithms are chosen that automatically create process maps and correlation. A complexity assessment is performed in order to determine the critical steps and paths throughout the process. Data abstraction is used to make the data more manageable by removing irrelevant information and reducing its complexity, thereby allowing further analysis. The process model can then be sub-grouped as appropriate, and the outcomes compared with benchmarks. Transitioning to the parameterisation phase, there is identification of activity related parameters and subsequent quantitative analytical work that requires calculation in order

to draw out insights. Another validation and benchmarking is done after this stage to validate the results obtained as well ensure that they are reliable.

The transition to a DES model requires that the process model is translated into simulation-ready form. This involves the specification of activity logic and dependencies, parameterisation to specify the simulation, as well as automation of repetitive activities and real-time updates for increased model efficiency and flexibility. Execution is then performed on this simulation, and its behaviour and results are tested for accuracy.

The model validation carried out using comparative analysis, where the DES model is compared against process mining results and expert opinions. If any disparities are noted, these are corrected to ensure that the model does adjust enough for accuracy. In order to carry out a scenario analysis, the DES model is prepared with typical data, scenarios are generated, simulations are run for these scenarios and results obtained from analysing them.

While Table 6.1 outlines the general steps of traditional simulation modelling approaches, Table 6.2 expands these steps into a more detailed framework tailored for an expert-driven process mining approach. The broad stages in Table 6.1, such as requirements gathering, conceptual modelling, parameterisation, and validation, are retained but unpacked into finer-grained activities that explicitly integrate process mining methods. For example, “Gathering Requirements” in Table 6.1 is represented by multiple steps in Table 6.2, including defining a data protocol, data gathering, data profiling, cleaning, and formatting. The terminology therefore differs between the two tables: Table 6.1 uses general language consistent with traditional simulation modelling, whereas Table 6.2 adopts more specific terminology drawn from process mining literature to capture the technical steps required for integration.

Table 6.2:Steps in expert-driven process mining approach.

Step	Description
1. Define clear objectives	Clearly define the objectives for integrating process mining into DES modelling.
Data Preparation	
2. Data Protocol	Define the protocol for data collection and processing.
3. Gather Relevant Data	Identify sources and collect event log data from various systems.
4. Data Profiling	Analyse collected data to understand characteristics, structure, and quality.
5. Data Cleaning	Remove outliers, correct inconsistencies, and reformat data for accuracy.
6. Missing Data Handling	Impute missing values where necessary and consult experts when necessary.
7. Data Extraction	Select relevant variables and extract data for further analysis.
8. Data Formatting	Format data for compatibility with analysis tools and ensure data consistency.
Conceptual Modelling	
9. Event Log Structure	Create a structured event log with IDs, activity labels, and timestamps.
10. Algorithm Selection	Select appropriate process mining algorithm(s) for pathway discovery.
11. Process Mapping	Automatically generate process maps from event logs to visualise activity flows. Compare models produced by different algorithms to evaluate suitability.
12. Complexity Assessment	Assess process complexity by identifying key steps and paths.
13. Data Abstraction	Simplify models through frequency/time-based filtering to retain key pathways.
14. Subgroup Modelling	Develop subgroup-specific process models where needed.
15.Validation and Benchmarking	Evaluate the results and compare them against benchmarks to validate the model.
Parameterisation	
16. Direct Parameter Identification	Identify parameters that can be obtained directly from the event log, such as activity durations, frequencies.
17. Derived Parameters	Derive secondary parameters through specific calculations.
18.Validation and Benchmarking	Evaluate the results and compare them against benchmarks to validate the model.
Transforming to DES Model	
19. Model Translation	Translate the process model into a DES model.
20. Activity Logic and Dependencies	Define the logic and dependencies between activities.
21. Input parameters into the DES model	Specify the parameters required to configure the simulation. Incorporating automation for repetitive tasks and enabling real-time updates improves efficiency and adaptability.
22. Run Simulation	Execute the simulation and verify its behaviour.
Model Validation	
23. Comparative Analysis	Compare DES outputs with process mining results and expert judgement to confirm plausibility.
24. Discrepancy Resolution	Identify and resolve discrepancies between DES outputs, process mining insights, and literature evidence.
25. Scenario Analysis	Define and design scenarios for experimentation.
26. Improved decision making	Use the simulation results to support better decisions.

6.2.3 Fully Automated Process Mining Approach

The fully automated process mining method for event simulation modelling involves the use of algorithms and data-driven techniques to generate detailed and data-based process models, which are then transformed into DES (Discrete Event Simulation) models. The entire process is executed without expert input or opinion.

In fully automated process mining conceptual modelling, parameterisation and translation into discrete event simulation (DES) modelling are automated processes. This implies that tasks like algorithm selection, process map generation, parameter recognition and DES model translation from process mining results are done without human intervention.

It starts with having clearly defined goals and a data protocol for uniformity. Relevant information is collected and then data profiling and cleaning are done to increase its accuracy. An event log structure is defined automatically for process representation. The event log is used to automate the selection of algorithms and process mapping. Complexity assessment and data abstraction are automatic, and process models for subgroup analysis is formed.

The parameter identification and quantitative analysis are automated for accurate depiction. Model translation into a DES model, activity logic and sequencing are automated. The input parameter and running simulation, both are fully automated, which can ensure the repeatability of analyses.

Table 6.3: Steps in fully automated process mining approach.

No.	Step	Description
1	Define clear objectives	Clearly define the objectives of the simulation process.
2	Data Protocol	Establish a protocol for handling data, ensuring consistency and adherence to standards.
3	Gather Relevant Data	Collect relevant data for the simulation, ensuring it reflects the real-world process.
4	Data Profiling	Analyze and profile the collected data to understand its characteristics and quality.
5	Data Cleaning	Cleanse the data by addressing inconsistencies, errors, and missing values.
6	Missing Data Handling	Implement strategies to handle missing data effectively.
7	Data Extraction	Extract relevant data for the simulation process.
8	Data Formatting	Format the data to meet simulation requirements.
9	Event Log Structure	Establish the structure of the event log for process representation.
10	Algorithm Selection	Automatically select algorithms suitable for process mining.
11	Process Mapping	Automatically generate process maps based on the event log.
12	Data Abstraction	Automatically abstract data to capture essential process features.
13	Validation and Benchmarking	Skip expert validation and benchmarking as the process is fully automated (no expert input).
14	Model Translation	Automatically translate the conceptual model into a discrete event simulation (DES) model.
15	Activity Logic and Sequencing	Automatically define activity logic and sequencing for the DES model.
16	Input Parameters	Automatically handle input parameters required for the simulation.
17	Run Simulation	The process of running the discrete event simulation.
18	Scenario Generation	Generate scenarios for further analysis.

6.3 Expert Driven Process Mining Approach vs Fully Automated Process Mining Approach: Comparative Analysis using the Mexican Breast Cancer Dataset

The expert driven process mining approach in chapter 4 was compared to a fully automated process mining approach using the same dataset. The expert driven process mining approach is briefly described first (the reader is referred to Chapter 4 for further details) and a detailed description of the fully automated process mining approach is then presented. Then, a comparative analysis aims to evaluate and contrast the outcomes and methodologies of both approaches.

6.3.1 Reiteration of Expert Driven Process Mining Approach

The case study using Mexican Breast Cancer dataset, documented in Chapter 4, aimed to reduce delay in breast cancer diagnostic procedures, implemented the "expert-driven process mining approach " for developing DES models. This method actively engages subject matter experts (i.e an oncologist) alongside process mining techniques to construct simulation models.

The initial phase involved collaborating with domain experts to map out breast cancer treatment processes. Expert insights were instrumental in managing the dataset, which underwent preprocessing and analysis. Process mining analysis using Disco software resulted in comprehensive process models that captured the complexities of diagnosing and treating breast cancer. These process models were manually converted into a DES model in Simul8 software using expert input, enabling simulation-based analysis of different scenarios to assess the impact of diagnostic delays and treatment initiation times.

This approach allowed us to extract vital information from historical data related to breast cancer diagnosis, treatments, and patient pathways. By integrating the insights provided by experts with process mining analyses, we were able to simulate potential operational improvements in breast cancer care, specifically by reducing diagnostic intervals and enabling earlier initiation of treatment.

The process of generating a conceptual model, parameterisation, and validation for DES modelling is the same as described in Chapter 4. As such, this approach is not described in this section and the reader is referred to Chapter 4 for further details.

6.3.2 Fully Automated Process Mining Approach

Fully automated process mining approach was used with the same Mexican Breast Cancer dataset (described in Chapter 4) to develop a DES model. Automated algorithms were deployed to discover patterns and autonomously generate process models. These automated models facilitated DES modelling, enabling simulation-based evaluations of breast cancer treatment pathways without direct expert input. The objective is to compare this model with the one produced in Chapter 4, which employed expert-driven process mining and to study any challenges and opportunities that arise.

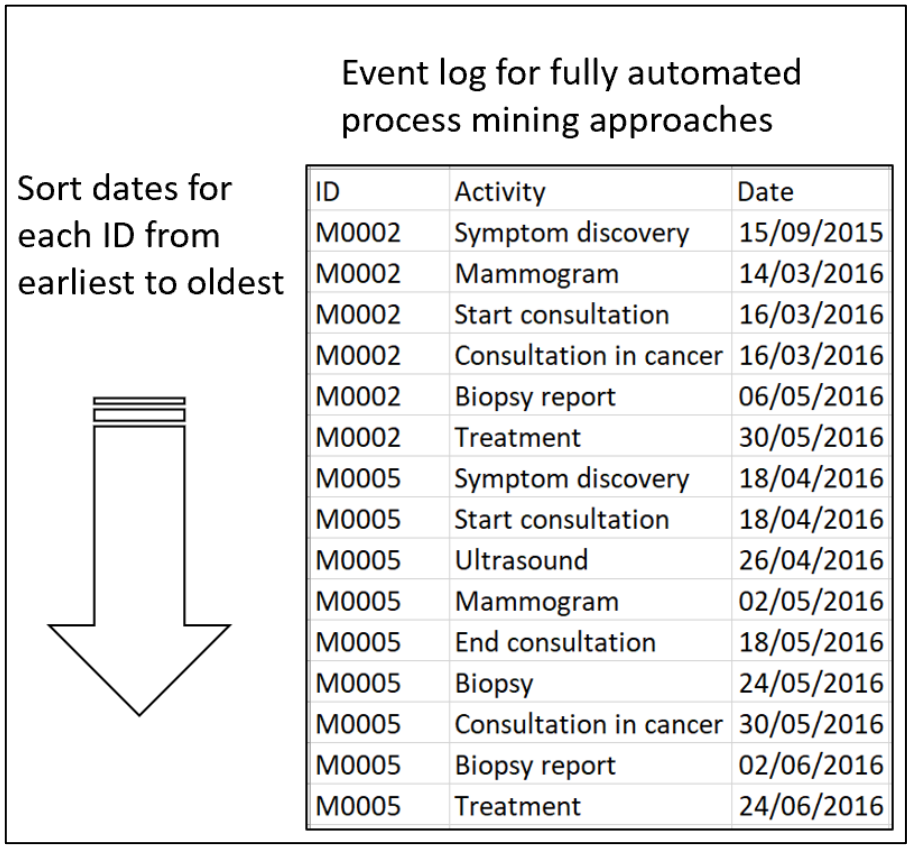
We employed Simul8 Professional, generously sponsored by Simul8, for PhD research on simulation modelling. The conceptual model and parameters were generated entirely through automated processes without expert input. Simul8 offers tools for automated process mining in DES modelling. Simul8 Professional was chosen for its integrated support for both process mining and DES modelling, enabling a seamless transition from discovered process models to simulation-based evaluations. While process mining tools like Disco, ProM, and Celonis lack direct simulation functionalities, and DES software such as Arena, AnyLogic, and FlexSim lacks built-in process mining capabilities, Simul8 Professional uniquely facilitates both steps within a single automated pipeline, eliminating manual intervention (Simul8, 2024). However, differences in software implementations of process mining algorithms may lead to variations in discovered models and simulation outcomes(Augusto et al., 2019).

The following sections elaborate on the development of our DES model using this fully automated process mining approach.

6.3.2.1 Conceptual Modelling

Event Logs as Input

We use the same cleaned event log as in the Mexican Breast Cancer case study in Chapter 4. The processes of data preparation and data cleaning are similar to those described in Chapter 4, with the exception that data formatting for event log creation requires arranging the start date from ascending to descending order because it is a requirement of the software; otherwise, the data cannot be read correctly.



ID	Activity	Date
M0002	Symptom discovery	15/09/2015
M0002	Mammogram	14/03/2016
M0002	Start consultation	16/03/2016
M0002	Consultation in cancer	16/03/2016
M0002	Biopsy report	06/05/2016
M0002	Treatment	30/05/2016
M0005	Symptom discovery	18/04/2016
M0005	Start consultation	18/04/2016
M0005	Ultrasound	26/04/2016
M0005	Mammogram	02/05/2016
M0005	End consultation	18/05/2016
M0005	Biopsy	24/05/2016
M0005	Consultation in cancer	30/05/2016
M0005	Biopsy report	02/06/2016
M0005	Treatment	24/06/2016

Figure 6.1: Event Log- Sorted Chronologically from Earliest to Latest Date

Conceptual model discovery

Event log data is loaded directly into the Simul8 tool. Built-in algorithm within Simul8 is used for process discovery. This algorithm analyse event logs to identify patterns, dependencies, and relationships among activities. They automatically construct process models representing the flow of activities, transitions, and dependencies within the process.

The resulting process model remains complex, although upon our observation, there are opportunities for simplification in the process model. The common pathways generated by the Simul8 approach are the same as those we generated in Chapter 4 using expert-driven process mining.

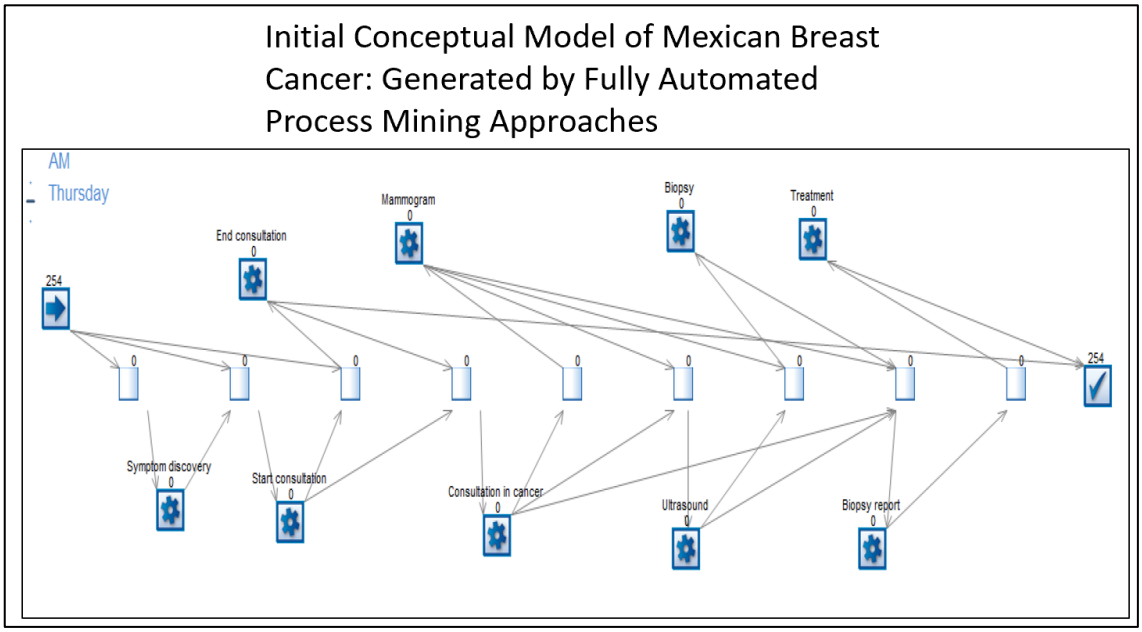


Figure 6.2: Conceptual model using a fully automated process mining approach*.

*The same process model is obtained as in Chapter 4 after using the final cleaned event log. This is because, for this dataset, we performed multiple iterations of cleaning even after producing the initial conceptual model. We had to reiterate the cleaning process due to hidden data errors initially identified as false negative data errors. False negative data errors refer to instances where erroneous data is mistakenly classified as correct or valid during the data cleaning process. In this context, it means that certain errors—such as incorrectly formatted timestamps were not initially detected as issues because they

appeared to be in a valid format. These undetected errors only became apparent after generating the process model and reviewing it with domain experts.

6.3.2.2 Parameterisation

Time to event duration

In setting input parameters for DES modelling using this integration technique, fully automated integration tools can directly provide input parameters from process mining results to DES modelling, particularly for arrival times and routing probabilities. However, for the time-to-event distributions of each activity, Simul8's automated process mining feature initially assigns an "Average" distribution by default. This distribution assumes a Normal probability distribution with a standard deviation of 25% of the mean.

This default approach is not suitable for modelling time-to-event distributions, as real-world event durations rarely follow a Normal distribution. Instead, they often exhibit skewed patterns better represented by distributions such as Exponential, Weibull, or Lognormal (Law, 2014). Therefore, we manually adjust the probability distributions to ensure greater accuracy, following the methodology outlined in Chapter 4.

Rather than relying on the default "Average" distribution or fitting a generic line, we define probability distributions based on historical data. Specifically, we construct histograms from observed time-to-event durations and use the frequency distribution of these data points to determine appropriate probability distributions. This data-driven approach ensures that the model more accurately reflects real-world process variability, leading to more reliable simulation outcomes.

Figure 6.3 illustrates how the default "Average" distribution is replaced with a frequency-based probability distribution derived from actual data. By making these adjustments, we ensure that the DES model is grounded in empirical observations rather than arbitrary assumptions, leading to more reliable and realistic simulation outputs.

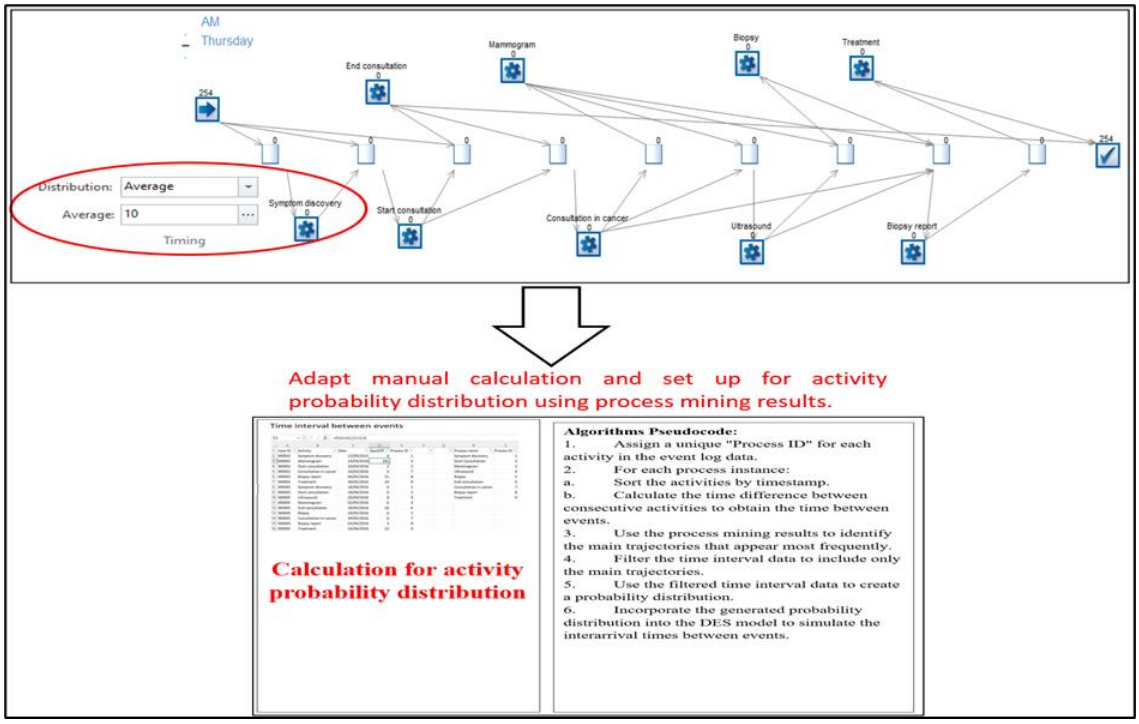


Figure 6.3: Fully Automated Process Mining: Transforming Default Average Time to Event into a Frequency-Based Probability Distribution

Interarrival time

For fully automatic process mining, interarrival time is calculated automatically and set as scheduled arrival in Simul8. The parameters provided align consistently with those obtained through process mining. In contrast, when employing expert-driven process mining, the interarrival time must be calculated based on results from process mining, as explained in Chapter 4. In this chapter, we detail the process of calculating interarrival time from process mining and subsequently setting it in DES modelling using Simul8. The sole difference lies in the units of time; in Simul8, these can be converted to standard time units.

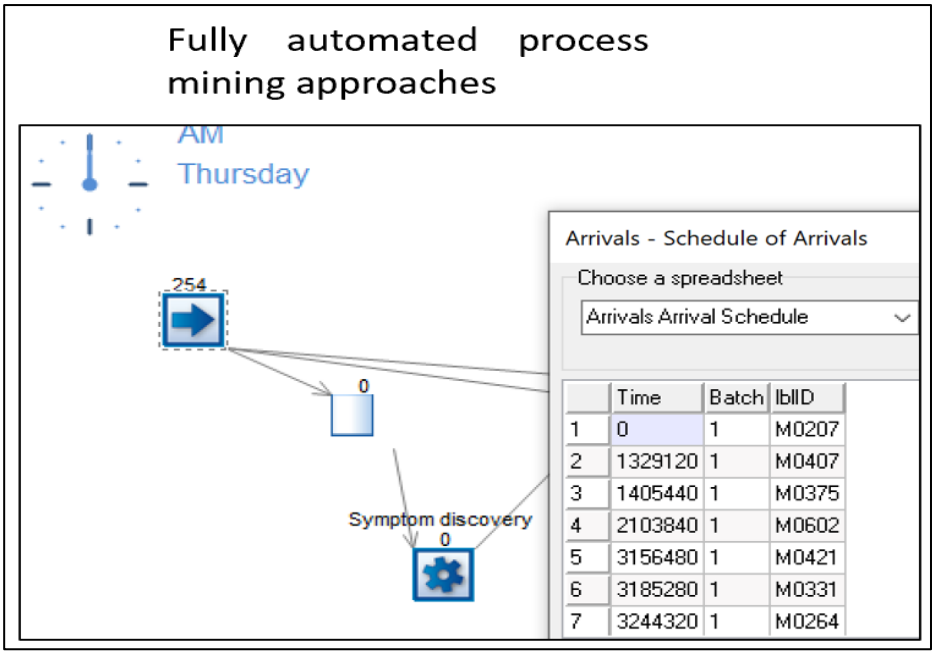


Figure 6.4: Setting Up Interarrival Time for Fully Automated Process Mining.

Routing probability

Routing probability for fully automatic integration is calculated automatically, providing percentage routing out and is directly set up in the Simul8 environment.

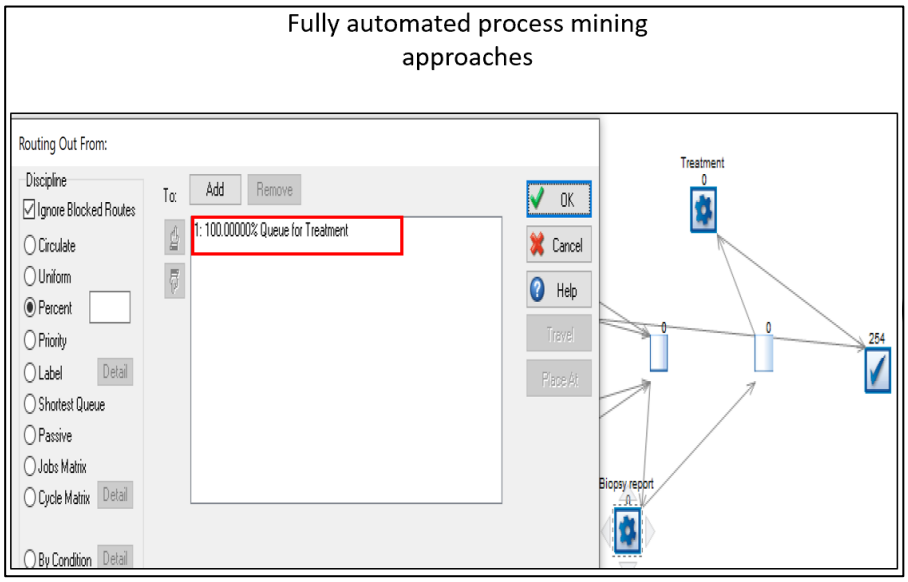


Figure 6.5: Setting Up Routing Probability for Fully Automated Process Mining.

6.3.2.3 Validation and Scenario Analysis

In the context of fully automated process mining methods, validation means comparing simulation models' output results with even logs data. Through comparing the simulated and observed real-world, we can validate the consistency of these models.

Validation is concerned with comparing the similarities and differences of the simulated behaviour to those witnessed in real world. The aim is to evaluate whether the system dynamics are adequately represented by the simulation models. Such degree of closeness between the simulated outcomes and the recorded events in event logs would imply that simulation is accurate and reliable.

Validation offers an opportunity to determine the validity of automated process mining algorithms in capturing the key attributes of real-world processes. Differences or variations between the simulated and observed data can be used to indicate the sections where one has to make improvements or even adjustments on the simulation models. The iteration of validation can be used to improve the models and make them closer to the way actual processes work.

The analysis of the scenarios is similar to the approaches based on experts in process mining as described in Chapter 4.

6.3.2.4 Summary of steps taken for fully automated process mining approaches

Through fully automated process mining approaches, we found that several steps in the integration process have been simplified, such as translating the process mining conceptual model and parameters into the DES model. However, model validation can only be performed by the modeller to ensure that settings and configurations are correct, as validation with process experts cannot be carried out due to the absence of expert opinions. Automatic integration is faster and more efficient in handling large datasets. It offers real-time or near-real-time integration of process data, making it suitable for dynamic and evolving systems.

6.3.2.5 Algorithm Transparency and User Control

Simul8's automated process mining feature uses proprietary, customized algorithms to analyse event logs and generate process models. However, the specific algorithms applied by the software are not publicly disclosed, and the process lacks transparency. The model presented to users is a simplified version of the full process, and user control over model generation is limited. User interaction in Simul8 is primarily restricted to modifying the event log before uploading it. After model generation, the software does permit some manual editing, such as adding or removing pathways or parameters that may be missing or misrepresented. However, default parameter values are automatically pre-set, and time distributions are typically based on simple averages.

The initial "spaghetti diagram" in Figure 4.3 was generated by visualizing all pathways from the dataset without any filtering. Simul8 then automatically simplifies this model using its internal algorithm to produce the refined version shown in Figure 6.2. Despite these capabilities, Automated Simul8 lacks key functionalities commonly available in specialized process mining tools such as Disco and ProM:

- **Event Frequency-Based Filtering:** Users cannot set thresholds to filter out infrequent transitions. All transitions are included regardless of significance, and infrequent transitions are not highlighted.

- **Variant Filtering:** The software does not support prioritization or exclusion of process variants. All variants are shown together, and users cannot view or isolate the most frequent pathways.

Unlike Simul8, Disco and ProM offer much greater transparency and user control over both algorithm selection and process model simplification.

Disco Software:

- **Algorithm Transparency**: Built-in use of Fuzzy Miner, which handles unstructured data by analysing activity frequency and strength of relationships without imposing rigid structure.

- User Controls for Simplification:
    - Thresholding slider to adjust model detail
    - Variant filter
    - Performance filter
    - Start/End point filter
    - Attribute filter
    - Timeframe filter

ProM Software:

- Algorithm Choice: Supports multiple algorithms including:
    - Alpha Miner
    - Heuristic Miner
    - Fuzzy Miner
    - Inductive Miner
- User Controls for Simplification & Filtering:
    - Event log filters (e.g., duplicate tasks, start/end events)
    - Variant filter
    - Performance analysis tools

These advanced functionalities in Disco and ProM enable users to explore, filter, and refine models interactively, promoting clearer insights and greater adaptability to different process mining objectives.

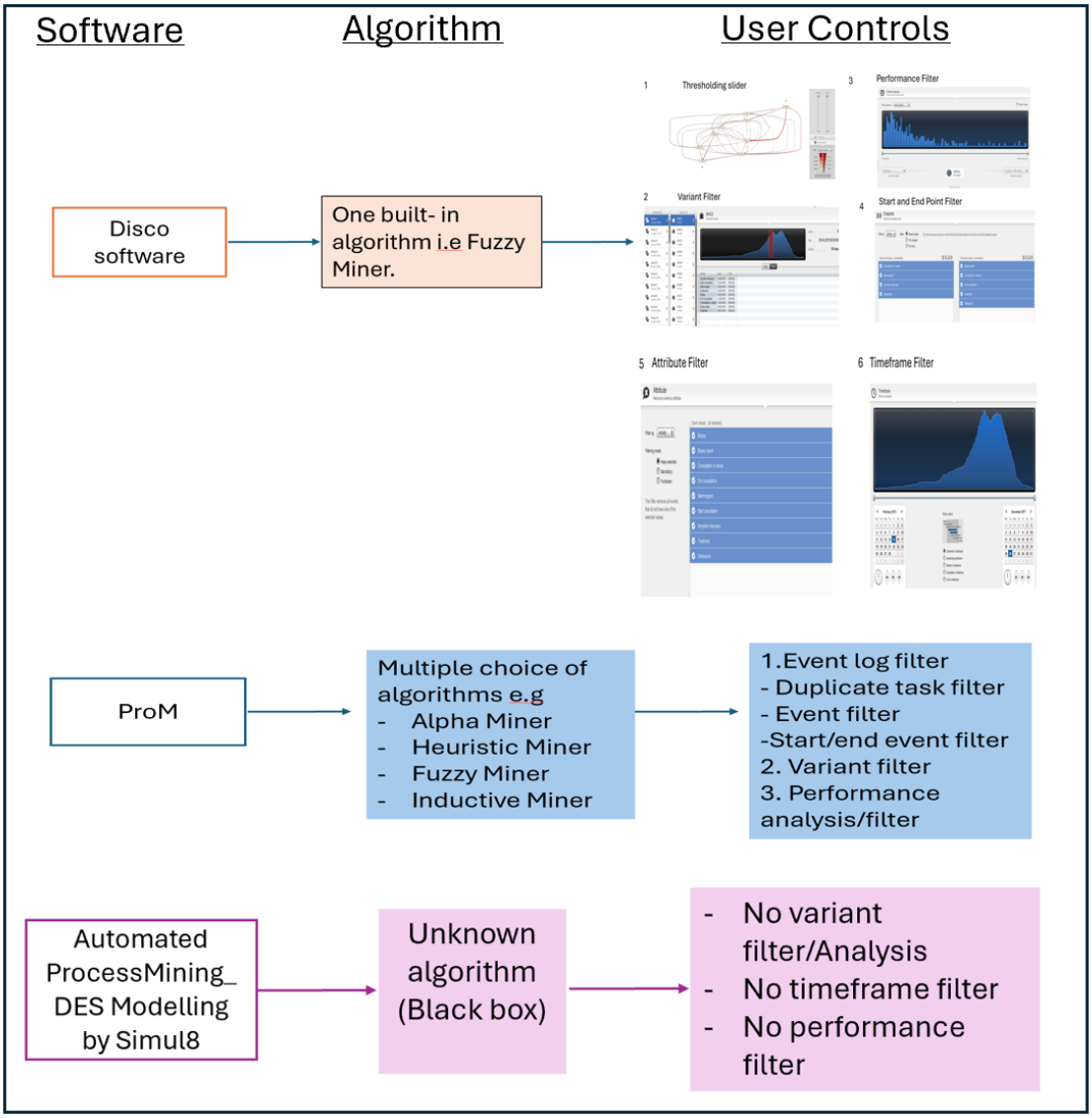


Figure 6.6 Algorithm Transparency and User Control

6.3.3 Comparative Analysis of the Results of the Expert Driven Process Mining Approach vs Fully Automated Process Mining Approach

This comparative analysis aims to evaluate and contrast the outcomes and methodologies of both approaches. The expert-driven approach relied on human expertise for data curation, feature selection, and process understanding, fostering detailed models. In contrast, the automated approach prioritized efficiency, employing algorithms for rapid model generation but potentially lacking nuanced insights captured by experts.

The same colour schemes in both figures indicate similar pathways originating from the same events in the process. The solid lines represent pathways that are consistently present in both the fully automated and expert-driven process mining models. In contrast, the dotted lines highlight pathways that are exclusive to one approach, appearing in either the automated or expert-driven conceptual models.

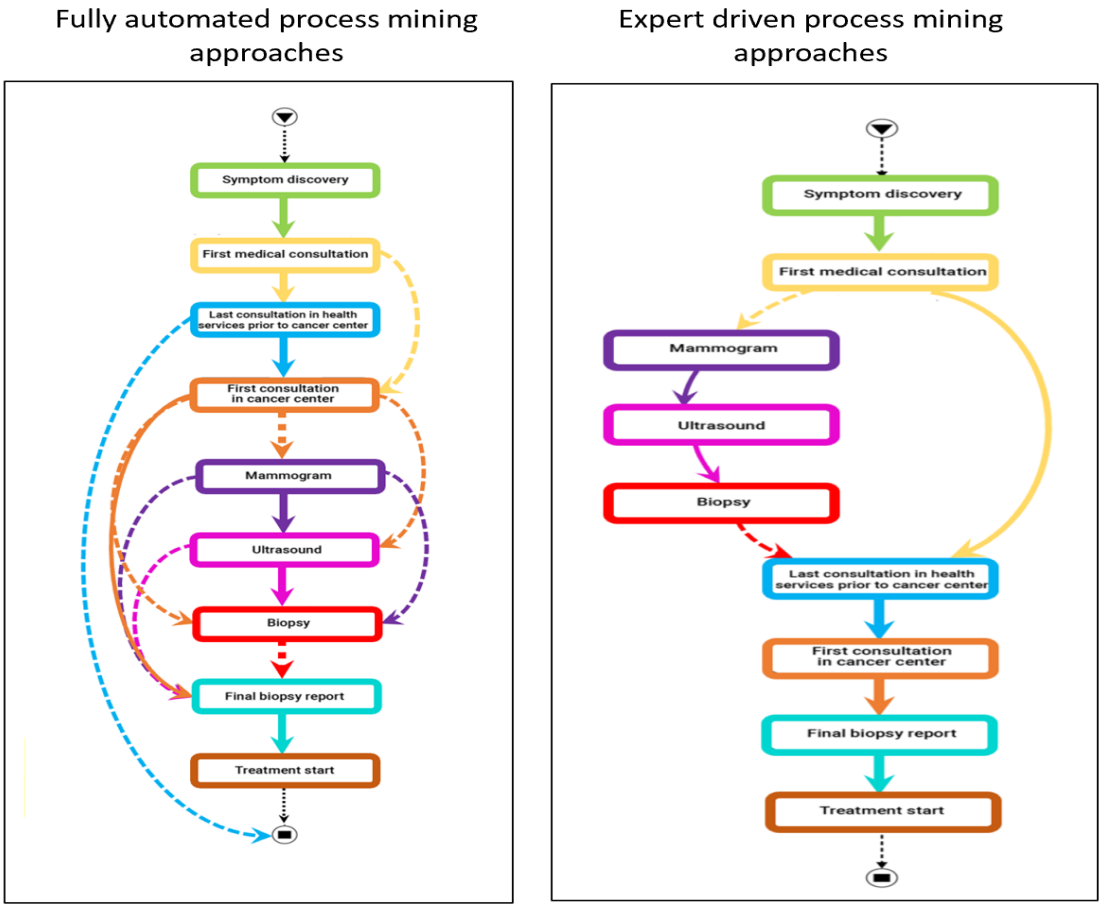


Figure 6.7: Mexican Breast Cancer Conceptual Model Generated by Fully Automated and Expert-Driven Process Mining Approaches.

Events Differences

We found that the events resulting from the process mining with expert-driven process mining for Mexican breast cancer cases are similar to the events present in the conceptual model generated by fully automated process mining. This is because both conceptual models use the same input event log from the actual process.

While both process mining and fully automated process mining rely on the same event log data, their approaches to interpreting and utilizing this information differ significantly. Process mining, especially when incorporating expert opinion, facilitates additional insights, refinements, and adjustments in the conceptual model. Expertise in this context influences the interpretation and categorization of events, potentially offering a more profound understanding of process flows beyond what raw event logs provide. Conversely, fully automated process mining operates solely on the input data without human intervention, potentially resulting in a conceptual model based purely on automated interpretations of event log data. As a consequence, these models might lack the insights and depth of understanding that human expertise can provide in refining and comprehending complex process flows.

Table 6.4: List of Event Conceptual Models Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study.

Fully automated process mining approaches	Expert driven process mining approaches
1 event: Symptoms discovery	1 event: Symptoms discovery
1 event: First medical consultation	1 event: First medical consultation
1 event: Mammogram	1 event: Mammogram
1 event: Ultrasound	1 event: Ultrasound
1 event: Biopsy	1 event: Biopsy
1 event: Last consultation in health centre	1 event: Last consultation in health centre

1 event: First consultation in cancer centre	1 event: First consultation in cancer centre
1 event: Final biopsy report	1 event: Final biopsy report
1 event: Treatment start	1 event: Treatment start

Pathways Differences

We revisited the process map generated through fully automated process mining and compared it with expert-driven methods. Figure 6.7 illustrates the conceptual model resulting from fully automated process mining using a cleaned event log at iteration 0 (initial cleaned event log without further data cleaning iterations). Initially, we observed significant differences in the pathways generated by the fully automated model. Specifically, the automated approach incorrectly positioned diagnostic events such as mammograms, ultrasounds, and biopsies after the first consultation in the cancer centre instead of before it. The automated model produced the following process sequence:

Start → Symptoms Discovery → First Medical Consultation → First Consultation in Cancer Centre → Mammogram → Ultrasound → Biopsy → Biopsy Report → Treatment.

This discrepancy arose due to timestamp inconsistencies in the dataset, which persisted despite data cleaning efforts. Even though we used a cleaned event log, hidden data errors remained, leading to incorrect event sequencing. With fully automated process mining, the absence of expert guidance prevented the correction of these errors, resulting in an inaccurate conceptual model. The findings indicate that some hidden timestamp errors persisted even after initial cleaning. These errors were subtle and did not violate basic data integrity checks but were only detected when analysing the logical flow of the process model.

Without an expert-driven reference model, identifying sequencing errors in the automated process model would be challenging. However, alternative data validation techniques could help detect such issues. First, cross-validation with domain experts should be conducted at multiple stages of data cleaning and model validation rather than solely after model generation. Second, logical consistency checks should be implemented to ensure that key diagnostic steps occur before treatment consultations (e.g., mammograms should not appear after cancer consultations).

These findings do not suggest that automation is inherently unreliable but rather highlight the risks of blindly relying on automated outputs without validation. While automated process mining is valuable for identifying broad trends in large datasets, its accuracy depends on data quality and logical consistency.

However, some minor variations in pathway representation may be due to differences in process abstraction between the two approaches. The expert-driven model, using Fuzzy Miner with expert guidance, allowed for manual refinement and filtering, whereas the fully automated process mining in Simul8 relied solely on algorithmic interpretation, but its algorithm is not published. Without expert validation, these sequencing errors in the automated model would have gone undetected.

Thus, while some differences in pathway representation may be nuanced interpretations, the incorrect event ordering in the automated model was a critical issue that required expert correction.

Nevertheless, the pathways with the highest proportion of patients were present in both conceptual models generated by fully automated and expert-driven process mining approaches (See Table 6.5).

Table 6.5: Pathways Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study.

Pathways	**Expert driven process mining approaches**	**Fully automated process mining approaches**	**Proportion of patients going through the pathway**
Symptoms discovery→First Medical Consultation	✓	✓	74.6 %
First Medical Consultation→ Last Consultation in Health centre	✓	✓	33.7%
Mammogram→ Ultrasound	✓	✓	52.5%
Ultrasound→ Biopsy	✓	✓	26.4%
Last Consultation in Health centre → First Consultation in cancer	✓	✓	47.1%
First Consultation in cancer → Biopsy report	✓	✓	75%
Biopsy report → Treatment	✓	✓	84.1%

The table below represents pathways that differ between the two conceptual models (i.e., the conceptual model from fully automated process mining and expert-driven process mining).

Table 6.6: Pathways that Differ Generated by Fully Automated and Expert-Driven Process Mining Approaches in the Mexican Breast Cancer Case Study.

Pathways	**Expert driven process mining approaches**	**Fully automated process mining by Simul8**	**Proportion of patients going through the pathway**
First Medical Consultation → Mammogram	✓	X	34.3%
Biopsy→Last Consultation in Health centre	✓	X	25.6%
First Medical Consultation → First Consultation in Cancer	X	✓	3.68%
First Consultation in Cancer → Ultrasound	X	✓	1.36%
Mammogram →Biopsy	X	✓	7.36%
Ultrasound→Biopsy Report	X	✓	5.43%
Biopsy→Biopsy Report	X	✓	5.04%

Parameters Differences

The parameters for inter-arrival time, routing probabilities, and time-to-event duration are the same as those obtained in expert-driven process mining, as detailed in Chapter 4.

6.4 Expert Driven Process Mining Approach vs Traditional Approach to Simulation Modelling: Comparative Analysis using UK Kidney case study.

The case study on the UK Kidney, documented in Chapter 5, initially employed an expert-driven process mining approach to produce a conceptual model and parameters. In this chapter, we present the traditional conceptual model developed by our collaborator for health economics modelling, utilizing DES (Discrete Event Simulation) modelling as described in Section 6.1.1. We compare the conceptual model produced by expert-driven process mining in Chapter 5 with the traditional conceptual model developed by our collaborator.

6.4.1 Expert-Driven Process Mining Approach

This approach involved the utilization of an expert-driven process mining approach to generate a conceptual model and parameters. This method leveraged the expertise of nephrologists in the field to extract meaningful insights from real-world data related to the UK Kidney case.

In Chapter 5, we generated a conceptual model and model parameters using expert-driven process mining. These process maps served as visual representations of the sequential steps and activities involved in the peritoneal dialysis (PD) treatment process. This visualisation aimed to enhance the understanding of patient pathways and the overall flow of the treatment.

The dataset used in this chapter allowed for an examination of the differences between two catheter insertion techniques: medical catheter insertion and surgical catheter insertion. Specifically, the analysis focused on the time taken to initiate PD dialysis and the time to infection for each technique. By analysing infection rates and durations for

each group, the chapter aimed to provide insights into how the catheter insertion technique might impact the occurrence and timing of infections. The findings from this analysis were intended to inform clinical decision-making.

The process model generated through process mining was then compared with the existing traditional conceptual model.

6.4.2 Traditional Conceptual Model by Collaborator

Our collaborator developed a traditional conceptual model specifically tailored for health economics modelling using DES (Discrete Event Simulation) modelling. This model served as a benchmark against which the outcomes of the expert-driven process mining approach could be compared.

In Chapter 5, we utilized the UK Kidney dataset. This dataset possesses an established conceptual model developed by our collaborators for their health economics modelling using Discrete Event Simulation, published in(Fotheringham et al., 2023). We conducted a Q&A session with our collaborator to understand how they came up with this conceptual model, enabling us to grasp the process and leverage it for comparative analysis of approaches. The discussion in this section is also drawn from insights provided in the paper which our collaborators shared with us about their Discrete Event Simulation model for health economics. Several important steps were involved in this process, such as:

6.4.2.1 Designing conceptual model

In the development of the conceptual model using the traditional method, to comprehensively address the intricacies of surgical catheter insertion and evaluate the cost-effectiveness of combined or dual pathways, the initiative commenced by collaboratively defining a research question. The collaborative efforts extended to gathering requirements, entailing the meticulous collection of relevant data on patient characteristics, outcomes, and associated costs.

The creation of the conceptual model was a process that involved discussions and interactions among different stakeholders such as the modellers, clinicians, and statisticians. The investigation focused on picturing the health states, transitions, and crucial events in surgical catheter insertion. The discussion revolved around the discrepancies between the model and the discrete event framework and the complexities of the patient journeys.

They used the analogy of a health state diagram with a start and end date that use for understanding the visual representation of different pathways. Special attention was paid to certain elements, including the PD highlighting its importance in the model.

In regard to the return of patients to catheter insertion health state, linear regression in the model was used. The search included the concept of patients from the catheter insertion to the first PD, last PD, or peritonitis, and the opportunity of returning to the catheter insertion after the hospitalization. The term “catalytic event” was introduced to describe multiple patient journeys, such as personality dialysis, exercise, and peritonitis.

They also focused on the transplant route and the portrayal of health states, HD and death in the haemodialysis centre. The lack of concrete transitions, for instance, HD to death, was justified pointing to the fact that these transitions were guided by the external data because of the short time horizon.

The then conceptual model was constructed using sequences of regressions, a subclass of graphical models, to depict the associations between centre and patient-related characteristics and catheter insertion technique and their influence on the likelihood of catheter-related events. They selected appropriate factors to incorporate into a multistate model of the patient trajectory from the catheter insertion through catheter events, modality changes, transplantation, and death, to estimate the effect of the catheter insertion technique on the rate of catheter-related events, controlling for relevant factors.

The selection of the final model was a joint effort, considering its ability to accurately represent the decision-making process and align with the objective.

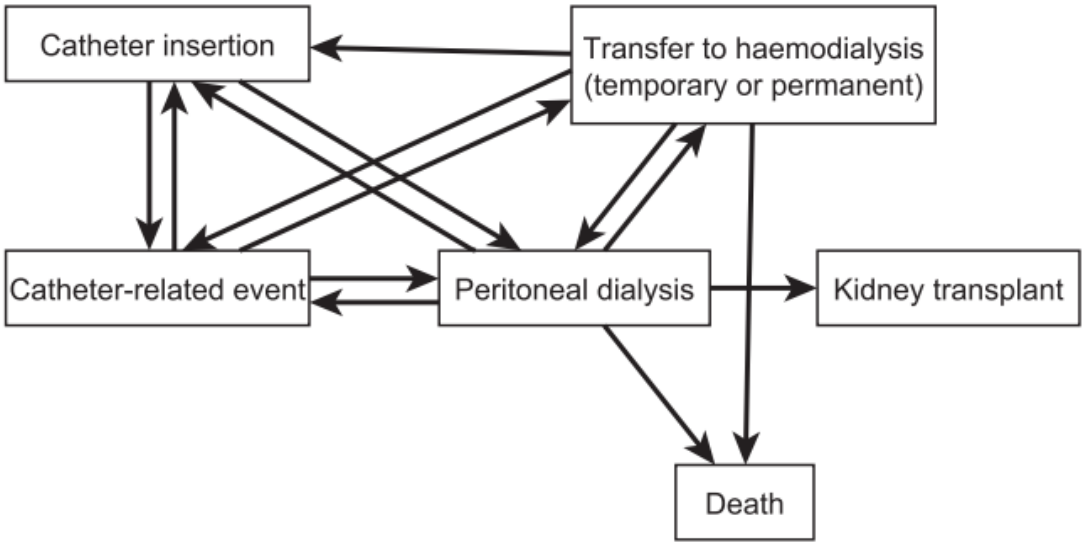


Figure 6.8: Conceptual model presented in the paper 'Catheter Event Rates in Medical Compared to Surgical Peritoneal Dialysis Catheter Insertion' based on the traditional method(Fotheringham et al., 2023).

6.4.2.2 Parameterisation Estimation

Parameterisation and assumption setting of parameters based on available data, literature, expert input and potentially data analysis from the UK-wide, multicentre prospective cohort study. The study likely involved gathering and analysing data on patient clinical history, dialysis unit-level characteristics and practices, and the subsequent occurrence of adverse catheter-related events within 12 months of follow-up. These parameters were then incorporated into the DES model to simulate the patient journey from catheter insertion through catheter events and various outcomes. Emphasis was placed on the comparative effectiveness and costs of medical and surgical pathways, with clear assumptions guiding the simulation and maintaining transparency.

6.4.2.3 Model implementation

As the initiative transitioned from conceptualization to implementation, our collaborator played a role in transforming the model into a functioning discrete event simulation. This involve coding the simulation using a software and programming languages, incorporating the relevant factors identified in the study to simulate the patient journey from catheter insertion through catheter events, modality changes,

transplantation, and death. They then used the data and insights from the study to inform the structure and functionality of the simulation model.

6.4.2.4 Analysis and Validation

The analysis and validation were carried out using statistical and simulation software to process and analyse the data from the UK-wide, multicentre prospective cohort study. For the sequences of regressions and multistate model, statistical analysis would have been conducted to establish associations between patient-level and centre-level factors with catheter insertion technique and their impact on the occurrence of catheter-related events. Validation of the model involved comparing simulated outcomes to observed data and existing clinical knowledge to ensure that the model accurately represented the patient journey and outcomes related to catheter insertion and events. Additionally, sensitivity analyses have been performed to assess the robustness of the model's findings.

Multiple iterations were collaboratively conducted, refining the model based on validation outcomes and feedback from the stakeholder. Central to the inquiry was a joint decision question that evaluated the cost-effectiveness of combined or dual pathways for surgical catheter insertion within the discrete event modelling context.

Throughout this collaborative process, the stakeholder, particularly statisticians and health economic modelers, engaged in dynamic interactions. They contributed valuable insights, including hazard ratios, which were y integrated into the discrete event model. Regular communication and collaboration between statisticians and health economic modelers ensured a cohesive approach, resulting in a comprehensive and robust decision model for surgical catheter insertion pathways.

6.4.3 Comparative Analysis of the Results of the Expert Driven Process Mining Approach vs Traditional Simulation Modelling Approach

Initially, our comparison focuses on the conceptual model in terms of the events, pathways produced, and parameters generated. Refer to the diagram shown below:

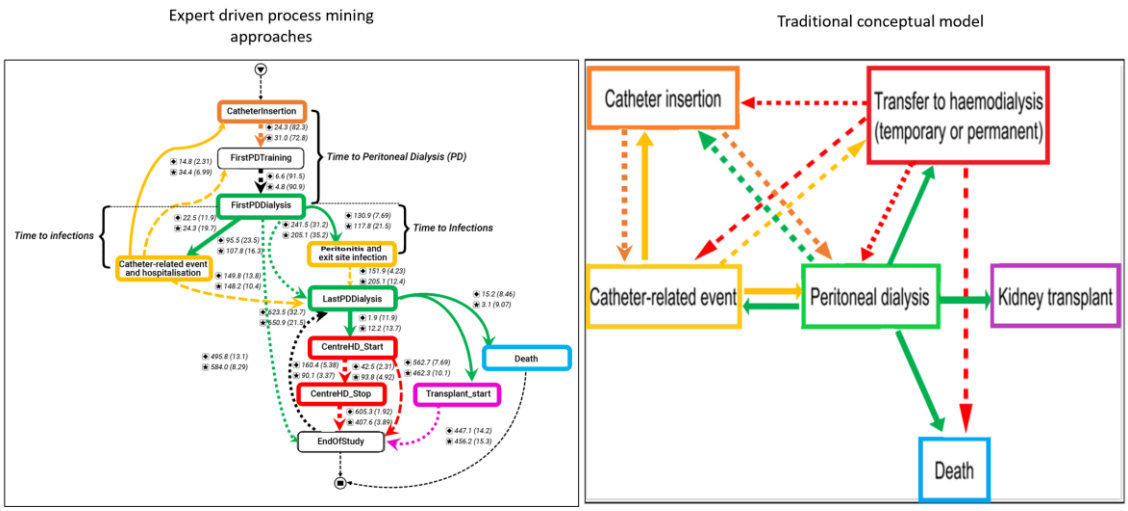


Figure 6.9: Conceptual Model by Expert-Driven Process Mining and Traditional Method

Events Differences

The same colour scheme in the conceptual model represents identical events that occurred within the models.

Table 6.7: List of Event Conceptual Models Generated by Expert-Driven Process Mining Approaches and Traditional Method using the UK Kidney Case Study.

Conceptual model by traditional method	Expert driven process mining approaches
1 event: catheter insertion	1 event: catheter insertion
1 event: peritoneal dialysis	2 events: First PD Dialysis and last PD Dialysis
1 event: catheter-related events	2 events: catheter-related events and hospitalization, peritonitis and exit site infections
1 event: transfer to HD	2 events: centre HD start and centre HD stop
1 event Kidney Transplant	1 event Transplant
I event death	1 event Death and 1 event end of study.

It can be seen that in expert-driven process mining approaches, the events are fine-grained. For instance, in case of Peritoneal Dialysis events, there is only one event available in the conventional concept model. However, in expert-driven process mining, it is detailed into two events: First PD Dialysis and Last PD Dialysis, respectively, thus capturing the time spent on PD Dialysis. Expert-driven process mining is the type of a process mining approach that takes into account not only data-driven information but also domain knowledge offered by experts. This method provides the ability to obtain a holistic understanding of processes using expert opinions as well as patterns discovered through data analysis. This approach is also good at capturing intricate events within each process step that requires contribution from expert in the domain. On the other hand, this needs a keen calibration of how detailed it should get to avoid unwarranted complexity. This also offers a room for improvement and adjustments, as the iterative refinement is based not only on expert opinions but data insights that tend to change over time.

On the other hand, the traditional approach bases solely expert judgments without incorporating insights from data analysis. This strategy is knowledge-centric in which the

model is constructed based on domain experts' extensive knowledge and experience. While that prevents possible bias of model input data, there is a risk of subjective biasing the model. While the ability to make good use of experts is a major strength, it might be at the expense of missing patterns that would have been clear from data-driven approaches.

The decision to apply expert-driven process mining over the traditional one also depends on the situation associated with the modelling task. While some processes may rely more on the subtle insights obtained through the use of a combination of expert knowledge and data analysis, others might necessitate greater dependence on expert opinions. Another important drawback that should be treated very attentively is the risk of oversights in the traditional method, especially when some events or patterns are not adequately acknowledged by experts. In certain situations, a hybrid method that relies on expert opinions supplemented by careful consideration of the data available could present an equitable and beneficial solution, utilizing both methods' advantages.

Pathways Differences

The solid lines represent pathways that exist in both models (expert-driven process mining and the traditional approach), meaning they are common across both methods. The dotted lines indicate pathways that appear in only one model either in the expert-driven process mining model or the traditional conceptual model, highlighting differences in how each approach represents the process.

In the right figure (traditional conceptual model), one HD event named 'Transfer to HD' has four dotted lines, implying that there are no corresponding pathways in the left figure (expert-driven process mining) for Transfer to HD, namely: Catheter Insertion, Catheter-related event, Peritoneal dialysis, Death. In contrast, the expert-process mining approach has divided the HD event into HD start and HD last in order to record the time the patient was in HD before changing modality.

With this example, we can say that although the traditional model incorporates complexity by depicting a transition and state network, process mining does more by

depicting the complex variations within an activity. In the case of the current study, the process mining analysis shows the complexity that is inherent in the process and is not captured by the 14 transitions of the traditional conceptual model.

In addition, concerning process variants, the pre-established model structure implicitly represents variations through transitions, for instance, from 'Temporary/Permanent Transfer to HD' to 'Kidney transplant.' On the other hand, process mining directly identifies and emphasizes these variants within individual activities. This fine-grained perspective helps to appreciate the different ways a process may unfold, despite the fact that it may occur in a single state or activity.

Furthermore, the conventional conceptual model used sequences of regressions to delineate the relationships between the patient and the dialysis unit-level characteristics and the catheter insertion technique and its effect on the incidence of catheter-related events. On the other hand, process mining achieves this automatically through the use of built-in algorithms in the software.

The main differences between these strategies are the level of granularity of the pathways and the balance of complexity and simplicity. The expert-driven process mining often provides a more flexible and more detailed representation, which reflects the subtlety of actual processes. On the other hand, the classic model can present a less complicated, perhaps more expert-driven approach that could miss some intricacies.

Parameters Differences

We study the difference between the parameters identified by process mining and the parameters estimated using traditional statistical methods. We have performed a comparison based on performance measures, particularly the average time for "time to PD from catheter insertion" and "time to infection," which includes time to Perit_exit infection.

The traditional model estimates each transition time individually using the method of maximum likelihood that assumes a Weibull distribution and semi-Markov property,

where the transition probability from one state to another depends on the time since entering the current state (from catheter insertion).

Process mining uses real event log data, but assumptions are still required at various stages of analysis. Its main advantage is that the generated parameters are grounded in empirical records, making them more reliable than parameters derived solely from hypothetical models. These parameters were then validated by clinical experts. For example, the estimated average time to PD dialysis was 30.9 days for patients undergoing medical catheter insertion, compared to 35.8 days for those undergoing surgical insertion. Clinical experts confirmed the plausibility of these results, which are consistent with existing knowledge that patients with medical catheter insertion typically start PD dialysis earlier than those with surgical insertion.

Furthermore, the average time to infection (Perit_Exit) is 130.9 days in the medical group and 177.8 days in the surgical catheter insertion group. This could means that the patients in the medical group develop infections earlier than those in the surgical group. In the same way, the median time to infection (CE_hosp) is 95.5 days in the medical setting and 107.8 days in the surgical catheter insertion setting. These results demonstrate the same trend of patients contracting infections earlier in the medical group compared to the surgical group. Validation from a clinical expert confirms their accuracy after examining the patient pathways and drawing from their experience.

The process mining technique can provide information regarding the duration and frequency of events, which can be further analysed to derive parameters such as time-to-event probability, routing probability, and inter-arrival time. In contrast, the traditional model may not provide these specific details.

6.5 Fully Automated Process Mining using a Big Dataset

In this subsection we discuss about our attempted to implement process mining in DES modelling using a large synthetic dataset of breast cancer called Simulacrum. However, we had to stop at the conceptual modelling stage because there were no experts available who were knowledgeable about the process being analysed. Simulacrum is a big dataset containing 226,406 cases of breast cancer and data on chemotherapy treatment known as Systemic Anti-Cancer Therapy (SACT), which includes information on the types and number of treatments received. The dataset is a type of synthetic dataset that is generated using algorithms to mimic the characteristics of real-world data.

6.5.1 Challenges without expert input

This section explores the feasibility of applying fully automated process mining on a large-scale, synthetic dataset without human intervention. While automation successfully generated a conceptual model, challenges emerged in variable selection, process interpretation, and validation due to the dataset's complexity and the absence of expert knowledge.

Although fully automated approaches can extract process models from event logs, expert input remains essential for:

Selecting relevant variables: Process mining generates all possible pathways without filtering for clinical relevance. Without expert guidance, it was difficult to determine which event sequences were meaningful versus which were artifacts of data structuring. Despite reviewing literature on chemotherapy processes, selecting appropriate variables remained challenging and risked misinterpretation.

Interpreting ambiguous pathways: Some event sequences did not align with expected clinical workflows, making it unclear whether they represented real-world variations or data inconsistencies.

Identifying missing or misrepresented process steps: The synthetic dataset lacked metadata explaining certain transitions between activities. Expert input could have helped determine whether these were real-world process gaps or data limitations.

Additionally, the process map generated from the dataset was highly complex, making it difficult to extract useful patterns for DES modelling. A process model without filtering and expert validation may introduce computational challenges, leading to unrealistic simulations. If all pathways were included, the model could become too complex and resource-intensive, reducing interpretability.

Another challenge was handling the dataset's size and computational demands. Given its scale, we had to store the data in a SQL database, extract relevant subsets, and manipulate them using R programming for structured analysis. Standard process mining tools struggled with the dataset size, requiring specialized computing resources and software to process it efficiently.

Despite these challenges, fully automated process mining provided valuable insights, particularly in generating an initial conceptual model without manual curation. The automation successfully captured the core elements outlined in the Table 6.3. However, fully automated process mining cannot evaluate clinical reasoning, prioritize pathways, or flag inconsistencies tasks that typically require expert input.

Thus, the conclusion is not that fully automated process mining is ineffective, but rather that automation alone is insufficient for interpreting highly complex and domain-specific event logs. The most effective approach may involve a hybrid model, where automation handles initial process discovery, while experts refine and validate key pathways to enhance model accuracy and usability.

6.6 General Comparison of the Different Approaches

6.6.1.1 Technical Aspects

Methodologies for conceptual model

Expert-Driven Process Mining relies on human expertise to interpret the complexity of processes, drawing heavily on the knowledge and experience of domain experts. In data preparation and preprocessing, the role of human experts is critical: they identify and extract relevant event log data, analyse its characteristics, and clean inconsistencies based on their detailed understanding of the process. Decisions about handling missing data are

also expert-driven, typically made in consultation with domain specialists to ensure accuracy. This approach is valuable because it produces comprehensive process models that capture nuanced process behaviours, but it can be resource-intensive and subject to variability depending on the skill of the experts involved.

Fully Automated Process Mining, by contrast, relies on algorithms to automate process modelling and thereby reduces dependence on human expertise. Data preparation and preprocessing in this method are carried out using predefined algorithms for data collection, profiling, cleaning, and model extraction. Automation provides efficiency and scalability, particularly in handling large datasets. However, this approach may sacrifice some of the contextual depth and interpretative insights that human experts provide. Its effectiveness depends heavily on the quality and flexibility of the algorithms employed.

Traditional Discrete-Event Simulation (DES) involves manually creating models from scratch, based on predefined rules that represent system logic, which provides detailed control over the modelling process. In traditional DES, the preferred source of input data is always the real-world system being studied, and this is used wherever possible. When system-level data are incomplete or unavailable, published literature or expert opinion may be used as supplementary sources. This approach offers fine-grained control, particularly when modelling complex rules and interdependencies between activities, but it is time-consuming and can become subjective if the rules are not clearly defined.

With respect to algorithm selection, process mapping, complexity assessment, and data abstraction, each approach has its own considerations. Both expert-driven and fully automated process mining approaches automate process mapping, but the degree of human involvement differs. Expert-driven approaches are well suited to managing complex processes and support detailed abstraction, while fully automated approaches emphasise efficiency and scalability. Traditional simulation, on the other hand, enables manual control over complexity and abstraction. Validation against benchmarks is essential across all approaches to ensure that the resulting process models are reliable and accurate. Ultimately, the choice of approach depends on the complexity of the process being studied, the expertise available, and the desired balance between automation and human input.

Parameter Estimation

In the context of Expert-Driven Process Mining, determining activity-related parameters is a group effort that requires domain experts' participation. These experts provide knowledge about processes and process mining results, ensuring that parameters reflect real-world process dynamics. Validation is conducted through multiple approaches, including expert reviews, statistical validation, and benchmarking against clinical standards. While expert validation (face validity) provides valuable insights, it is inherently subjective and should be complemented with statistical validation, where model outputs are compared with independent data not used in model development (Sargent, 2010). Benchmarks derived from clinical standards or established best practices also serve as an additional layer of validation to enhance model credibility.

In Fully Automated Process Mining, activity parameters are initially calculated from data using algorithms and statistical methods. By default, Simul8 assigns average-based distributions to time-to-event durations. However, recognizing the limitations of this, we manually adjusted the parameters by applying a frequency-based distribution derived from historical data. While this method improves accuracy, fully automated approaches may still overlook variations captured in empirical data, which are often essential in simulation modelling.

In traditional Discrete Event Simulation (DES), parameters are typically derived from historical data, which is used to fit probability density functions (PDFs) whenever available. DES modellers prioritize empirical data over published literature or expert opinion, which are only used as last resorts when historical data is unavailable. The approach involves quantitative analysis, where simulation parameters are manually set based on fitted distributions. Sensitivity analysis and comparisons with real-world data are standard validation techniques, allowing for refined parameter optimisation. While this method provides greater control over parameter assumptions, it can be more resource-intensive compared to fully automated approaches.

Techniques for validation

In an Expert-Driven Process Mining, validation is a complex procedure involving expert judgment and domain knowledge. The methodology incorporates a comparative analysis by mapping the DES model with process mining outcomes and expert advice. This correspondence permits an integrative assessment of the degree to which the model reflects the complexity of actual processes. The inconsistencies are sorted out by expert opinions, making use of the deep-rooted knowledge of domain experts to calibrate the model. As this practice introduces subjectivity, it creates an opportunity to incorporate subtle details and complexities of the context into the simulation.

As data-driven approach is applied in Fully Automated Process Mining, the validation step almost similar but without expert input. A statistical analysis of the DES model is made by comparing its outputs and those of the real world with process mining results. Discrepancies are found by statistical analysis, an automated and time-saving device for correcting the model. The focus is on quantitative measurements to trade off some qualitative observations for scalability and objectivity. This approach is especially applicable when large datasets and efficiency are important factors.

Validation approach of the traditional Discrete Event Simulation comprises both quantitative and qualitative elements. Sensitivity analysis as well as comparison of the model with empirical results can be viewed as comparative analysis. Sensitivity analysis allows identification of discrepancies and manual adjustment of parameters on the basis of expert judgments results in the fine-tuning of the model. This method provides a high level of control on the simulation and validation process but is labour intensive. It aligns the quantitative precision and qualitative insights offered by experts from the domain.

6.6.1.2 Logistic and Resource Aspects

Skill Sets

Expert-Driven Process Mining is dependent on experts who are individuals with a profound knowledge of the processes being modelled. The critical competency required for this strategy is domain knowledge and process analysis. They have to assess

complicated processes and cooperate efficiently in order to generate quality data. This is a collaborative process that requires communication skills as the domain experts provide qualitative insight that helps shape the modelling results.

Fully Automated Process Mining requires a technical-level skill set, one that is based on proficiency in data science, algorithm designing and statistical analysis. A group of people working with this methodology should have proficiency in managing big data sets, building algorithms for process mining, and conducting statistical analysis to check the model. It is important to be proficient in programming languages, data manipulation and statistical tools. This strategy is well-suited for situations involving level of data richness, with model development and validation being focused on automation and quantitative analysis.

A traditional Discrete Event Simulation would require another skill set that emphasizes skills regarding simulation. Those who participate in this way require a firm grounding in simulation theory, an understanding of modelling techniques as well as the capacity to appropriately characterize system dynamics. Knowledge of the fundamentals of discrete event simulation, good modelling tool skills and ability to interpret and verify the results of a simulation are essential. This approach allows for a fine-grained modelling process and is applicable in cases where comprehensive simulation and verification are critical.

Time Commitments

The time commitment for Expert-Driven Process Mining is largely subjected to the availability of experts in that domain. The collaborative element of this approach that entails expert contribution and manual operation may also delay the modelling timeline. This sometimes increases the period of model development since there would be requirement for extensive discussions, reviews and qualitative evaluations. In many cases, the pace is predetermined by the schedules of experts involved in this process.

Fully Automated Process Mining is likely to be quicker since the algorithms are automated. The use of automated procedures can help speed up the extraction process for process models. Nevertheless, the general time spent will depend on the requirement of

heavy preprocessing of data. Data preprocessing, such as cleaning and formatting it, to be used in automated algorithms may take up a considerable amount of work before the onset. When the data is prepared, automated processes can optimise modelling in the way that it becomes generally more accelerated than with manual approaches.

The time duration of Traditional Discrete Event Simulation can be different from one system to another due to the differences in system complexity and manual efforts. The complexity of the modelling, requirement for parameterisation and level of detail, influence its duration. The manual control offers the advantage of flexibility, but it can also increase modelling time. Depending on their complexity, simpler models could be created in a rather short time span and validated right away, while complicated systems would need an extended period for meticulous modelling and validation.

Resource Needs

The resource requirements for Expert-Driven Process Mining primarily involve both domain experts and simulation experts. Domain experts play a crucial role in collaborative discussions, qualitative assessments, and process modelling, ensuring that the extracted process model aligns with real-world workflows. However, simulation experts are also required to translate these expert-driven insights into structured models suitable for DES, as domain experts alone may not have the technical expertise needed for simulation modelling. Interdisciplinary collaboration is essential to ensure an accurate and implementable process model.

Fully Automated Process Mining primarily relies on computational resources for algorithmic processing, extraction, and analysis of process models. The key requirements include powerful computational infrastructure capable of handling large datasets, big data storage for managing event logs, and high-performance computing for running process discovery algorithms. While automation reduces the need for direct human intervention, human expertise is still required for system setup, validation, and oversight to ensure that the outputs are meaningful.

The requirements for Traditional Discrete Event Simulation (DES) include both human and computational resources. Simulation experts are necessary to handle model formulation, parameterisation, and execution, ensuring that the simulation is methodologically sound. At the same time, domain experts are essential to validate and refine the model to ensure it accurately represents real-world processes. A collaborative effort between domain experts and DES modellers is critical, as neither group alone can develop a realistic and reliable simulation model. Simulations with higher complexity and detailed parameterisation require greater computational power, making computing infrastructure another essential resource for running large-scale simulations.

Software Utilization

The use of software in the Expert-Driven Process Mining may include process-mapping tools that are general-purpose. Usually, these tools are created in the form of collaborative process mapping visualisation and documentation. They are not designed to be used with an automated process mining algorithm, but they offer a way for a domain expert to manually map and analyse processes while including their knowledge of the domain in the modelling.

Fully Automated Process Mining uses special tools of process mining and its own algorithms. Toolkits used in this area are created for the purpose of automatically extracting process models from event logs. They usually use sophisticated algorithms in order to discover processes. The software used in this method is designed to process large amounts of data rapidly and automatically, thus reducing the reliance on manual processes for modelling.

The Traditional Discrete Event Simulation employs simulation software equipped with the capabilities of a manual model build-up and analysis. Such tools give the user an opportunity to define system entities, events and processes in a manual way. Features often found in simulation software for this method are, parameterisation, scenario testing and result analysis. Although they are not specifically meant for process mining, these tools provide a high level of control over the modelling activity enabling users to produce elaborate simulations depending on predefined rules and assumptions.

6.6.2 Comparative Table Illustrating Key Differences

Below is a comparative table highlighting the key differences in steps between the three approaches - Expert-Driven Process Mining, Fully Automated Process Mining, and Traditional Discrete Event Simulation.

Table 6.8:This table provides a comparative overview of the steps involved in the three approaches, highlighting which steps are applicable to each approach. The "√" indicates the inclusion of a step in the respective approach, while "-" signifies that the step is not typically associated with that approach.

Step	Traditional Discrete Event Simulation	Expert-Driven Process Mining	Fully Automated Process Mining
1. Define clear objectives	√	√	√
Data Preparation			
2. Data Protocol	√	√	√
3. Gather Relevant Data	√	√	√
4. Data Profiling	√	√	√
5. Data Cleaning	√	√	√
6. Missing Data Handling	√	√	√
7. Data Extraction	– (Manual)	√	√
8. Data Formatting	– (Manual)	√	√
Conceptual Modeling			
9. Event Log Structure	– Not Applicable	√	√
10. Algorithm Selection	– Not Applicable	√	√(Automated)
11. Process Mapping	– (Manual)	√	√(Automated)
12. Complexity Assessment	– (Manual)	√	√(Automated)
13. Data Abstraction	– (Manual)	√	√(Automated)
14. Process Model for Subgrouping	– (Manual)	√	√(Automated)
15. Validation and Benchmarking	√	√	– (No Expert Validation)
Parameterisation			
16. Parameter Identification	√(Manual)	√	√(Automated)
17. Quantitative Analysis	√(Manual)	√	√(Automated)
18. Validation and Benchmarking	√ (Manual)	√	– (No Expert Validation)
Transforming to DES Model			
19. Model Translation	√(Manual)	√	√(Automated)
20. Activity Logic and Sequencing	√(Manual)	√	√(Automated)
21. Input Parameters	√(Manual)	√	√(Automated)
22. Run Simulation	√(Manual)	√	√(Automated)
Model Validation			
23. Comparative Analysis	√(Manual)	√	√
24. Discrepancy Identification	√(Manual)	√	√
Scenario Analysis			
25. Scenario Generation	√	√	√
26. Outcome Analysis	√	√	√

6.6.3 Discussion

The traditional approach to model development aligns closely with clinical expertise but has some limitations. These models provide in-depth insights into disease mechanisms, aiding in personalized treatment strategies and diagnostic assessments. However, their development is often time-intensive, relying heavily on subjective expert input that might inadvertently introduce biases. Additionally, their interpretability might not always match their complexity, potentially posing challenges in direct clinical translation. The critical challenge lies in balancing the expertise-driven precision with scalability and adaptability to cater to diverse patient populations and evolving healthcare landscapes.

Expert driven process mining approaches, integrating expert-driven insights with automated techniques, present a promising avenue for advancing healthcare simulations. These models aim to combine the strengths of both approaches, offering more accurate predictions while broadening insights. However, the critical challenge lies in effectively integrating diverse data sources and process mining techniques without compromising interpretability or introducing biases. Furthermore, while expert driven process mining approaches offer a potential solution to bridge the gap between precision and broader insights, their validation protocols and interpretive frameworks require attention to prevent misleading conclusions or misinterpretations.

Although, fully automated models provide the benefit of speed and width of analysis; there might be limitation as to how much one can do automation in terms domain specific insights obtained from expert. They yielded results quickly, but their analysis and clinical relevance are not straightforward enough to require final interpretation by themselves. Besides, automated models could also inherit biased in modelling.

The assessment of these modelling approaches underlines the need for caution and in-depth analysis. While traditional methods provide more precision, expert driven process mining aims to balance precision with broader insights, while for the fully automated approaches, it is speed and completeness. Nevertheless, each of these methods has certain challenges and limitations that should be taken into consideration. Given the above perception, it is likely that healthcare community's focus should be on refining hybrid

models to ensure seamless integration, addressing biases in automated algorithms, and establishing rigorous validation protocols to guarantee the accuracy, interpretability, and direct clinical applicability of these models.

CHAPTER 7

ENHANCED FINAL STEP-BY-STEP FRAMEWORK AND BEST PRACTICE CHECKLIST

7.1 Chapter Overview

This chapter introduces an enhanced, step-by-step framework for integrating process mining into DES modelling. The framework is developed based on a thorough review of existing literature and is further refined by the experience gained from real-world case studies conducted in chapters 4 and 5. Additionally, we provide a best practice checklist to guide practitioners interested in using process mining to support DES modelling.

7.2 Final enhanced step-by-step framework.

In this section, we present our enhanced framework for integrating process mining and DES modelling.

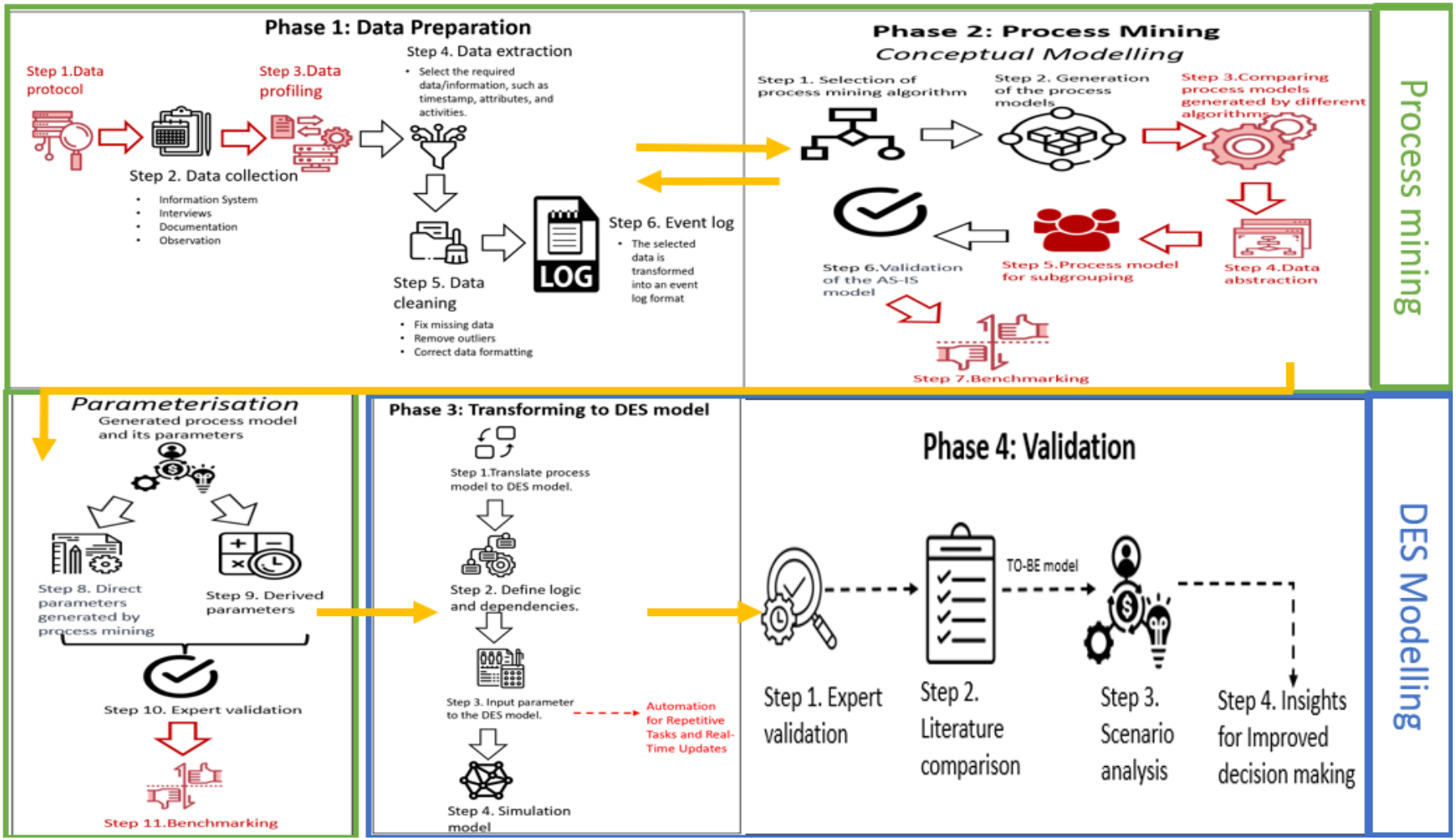


Figure 7.1:Enhanced Final Step-By-Step Framework.

Figure 7.1 illustrates the final steps of the integrated framework for process mining in DES modelling, developed from both existing frameworks (e.g. Rozinat, Tamburis) and insights gained from the case studies. Some of the "new steps" highlighted in red are activities that are commonly undertaken in practice but were made explicit here because the case studies showed their critical role when working with healthcare datasets. For example, data profiling is not unique to this framework, but the challenges observed in the case studies such as missing data, inconsistent event labelling demonstrated the need to formalise it as a distinct step. Other steps represent genuinely new contributions introduced in this research, based on gaps identified through the case studies.

Data Protocol & Data Profiling

A data protocol, particularly during the data collection process, plays a pivotal role in ensuring the accuracy and completeness of data used for process mining. It ensures that data is meticulously analysed and contains precise and comprehensive information. Based on our experience implementing the case study, we found that the kidney case study has better data quality compared to the breast cancer dataset. Drawing on our broader experience with case studies and a review of the literature, we conclude that a well-defined data protocol during data collection is essential for generating a high-quality dataset. We introduce the steps in our framework to guide future research in this area.

Additionally, data profiling helps identify crucial characteristics within the dataset for use in process mining. This step supports a better understanding of the data structure and quality, contributing to the overall reliability of the process.

The advantages of introducing a detailed data protocol during data collection, along with data profiling, are substantial. By thoroughly analysing data through these steps, the framework effectively mitigates issues arising from inconsistent or inaccurate data. Enhancing data quality at an early stage strengthens the entire integration process of process mining into the DES model, making it more robust and trustworthy.

Comparing process models generated by different algorithms

Moreover, we have conducted comparisons using different process mining algorithms in our work. We also introduced the steps for comparing process models resulting from different algorithms as part of our framework. We validated the resulting model with our clinical expert to confirm its correctness, as it was sufficient in producing an appropriate and accurate process model to achieve our objective. We did perform the comparison but decided to proceed with Fuzzy Miner as the most suitable choice for this study. For future research, it would still be beneficial to explore and compare additional algorithms to identify the best-fit process model for different datasets and contexts.

Data abstraction

Additionally, we emphasize the significance steps of data abstraction and expert validation to enhance the credibility of the resulting process model and parameters. Data abstraction step involves simplifying the process model to focus on the essential elements of the process while abstracting away unnecessary details. The goal is to create a conceptual model that retains the core characteristics and behaviour of the process, allowing for a more efficient and manageable representation of the process. This is normally done during the conceptual modelling phase and then adapted in the DES modelling phase, where the process model obtained from process mining is translated into a simulation structure. This involves simplifying the map to the level of detail required, ensuring alignment with the study objectives, and representing it in a way that can be parameterised in the DES. Moreover, including a step to subgroup or classify process models during conceptual modelling is a practical enhancement. Categorization enables focused analysis within distinct subgroups, facilitating targeted improvements or modifications.

Subgrouping process models

Our framework introduces subgroup classification as an essential modelling step. This involves categorizing event logs into subsets based on attributes such as performance metrics or process variations. For example, in our context, we grouped patients by different service providers specifically public and private providers in the Mexican breast cancer dataset.

Benchmarking

We also suggested comparing the process model outputs with existing model and parameters for evaluation. This approach ensures a strong, reliable framework for real-world scenarios, adhering to rigorous methodology and producing valuable insights for informed decision-making. However, it's noteworthy that a more structured and strategic benchmarking procedure is required. Currently, we are engaged in a comparison of the process model and relevant parameters, particularly the time-to-event parameters, with the existing process model. This approach seeks to ensure the robustness of our framework by focusing on specific aspects for comparison.

Additionally, the potential integration of automation for repetitive tasks further augments the efficiency of the DES model. By automating routine and repetitive activities, the model can simulate these tasks efficiently while eliminating unnecessary manual intervention. The increasing demand for testing steps on real-time data is noteworthy, however, we have not accounted for it due to data limitations. This opportunity could serve as a basis for future research endeavours.

7.2.1 Framework Steps and Implementation in the Case Study

In this section, we present the steps from the framework that were implemented.

Table 7.1: Framework Steps and Implementation in the Case Study.

Phase	Step	Implemented Section
Phase 1: Data Preparation	Step 1: Data Protocol*	4.5.1.1, 5.5.1
	Step 2: Data Collection	4.5.1.2
	Step 3: Data Profiling*	4.5.1.3
	Step 4: Data Extraction	4.5.1.4
	Step 5: Data Cleaning	4.5.1.5
	Step 6: Generating an Event Log from Process Data	4.5.1.6
Phase 2: Process Mining	Step 1: Select Algorithms for Discovering Process Models	4.5.2.1
	Step 2: Generated Process Model	4.5.2.2

	Step 3: Comparing Process Models Generated by Different Algorithms*	4.5.2.3
	Step 4: Data Abstraction*	4.5.2.4
	Step 5: Subgroup process model*	4.5.2.5
	Step 6: Parameterisation	4.5.2.6
	Step 7: Validation of AS-IS Model	4.5.2.7
	Step 8: Benchmarking*	6.4.3
Phase 3: Converting Process Mining to DES Modelling	Step 1: Translate Process Model to DES Model	4.5.3.1
	Step 2: Define Logic and Dependencies	4.5.3.2
	Step 3: Input Parameters to the DES Model	4.5.3.3
	Step 4: Simulation Model	4.5.3.4
Phase 4: Validation	Expert Face-to-Face Validation	4.5.4.1
	Literature Validation	4.5.4.2

*New steps tested in the Mexican Breast Cancer Dataset.

7.3 User Guidance to Apply the Enhanced Framework

This guidance outlines how to apply the Enhanced Final Step-by-Step Framework (Figure 7.1) in practice. Designed to support process mining integration into discrete event simulation (DES), the framework offers a structured methodology particularly relevant for healthcare process analysis and improvement. It guides users through four interconnected phases, from initial data preparation to simulation-based validation.

Note: The new step shown in brackets refers to an enhancement that was tested in our case study and subsequently incorporated into the framework

7.3.1 Phase 1: Data Preparation

This phase ensures that data collected for process mining is complete, high-quality, and ready for conversion into event logs. It consists of six sequential steps, two of which, Defining a Data Protocol and Conducting Data Profiling.

7.3.1.1 Step 1: Define Data Protocol (New Step)

Create a clear data protocol document that defines data sources, access methods, formats, and responsible personnel.

Example: Before extracting data on breast cancer diagnosis from Hospital X, prepare a protocol specifying:

Data source: Electronic Health Record (EHR) system.
Scope: Cases diagnosed between 2020–2022.
Fields: Patient ID, activity name, timestamp, department.
Responsible officer: IT Dept Head.

Case study reference: In the Mexican breast cancer study (Chapter 4.5.1.1, pp. 112–113), the data protocol included approval from the University of Sheffield and defined access to six key structured data files from INCan and HGMEL, covering cases between 2016 and 2017. These files included patient demographics, service consultations, diagnostic and treatment activities. The protocol clearly outlined anonymisation, timestamp formatting, and responsibilities for data cleaning and quality checks.

7.3.1.2 Step 2: Data Collection

Collect data through system exports, interviews, observation logs, or clinical notes.

Example: Export visit logs from the oncology department's EHR system and cross-verify them with specialty service schedules and consultation records to construct reliable patient pathways.

Case study reference: In the implementation (Chapter 4.5.1.1, pp. 112–116), data were obtained from structured sources, including spreadsheets and .dta files, which covered demographics, consultations, diagnostics, and treatments. These structured files were complemented by interviews conducted in the original Mexican study, which helped address critical gaps particularly related to care-seeking behaviour and private facility usage.

7.3.1.3 Step 3: Conduct Data Profiling (New Step)

Use SQL queries or profiling tools (e.g., R, Python Pandas) to examine missing values, event frequencies, and value distributions.

Example: Check if all patient entries have timestamps for biopsy, diagnosis, and treatment. Identify if timestamps are recorded in inconsistent formats (e.g., "2021-04-01" vs "01/04/2021").

Case study reference: Chapter 4.5.1.2 describes the profiling process in detail, using R to identify missing fields such as treatment dates (missing in 562 of 700+ cases) and duplicated or incorrectly formatted dates (e.g., invalid "6201") (p. 121–122).

7.3.1.4 Step 4: Data Cleaning

Fix inconsistent date formats. Impute or remove records with missing critical fields. Remove duplicate or irrelevant activities.

Example: Remove cases with missing diagnosis date or where timestamps are outside the expected study period.

Case study reference: Chapter 4.5.1.4 describes how inconsistencies were resolved with the help of data owners. Irregular timestamps and ambiguous activity names were cleaned

or grouped into valid categories, such as merging “auxiliary diagnostic tests” under “diagnostic imaging” (p. 126).

7.3.1.5 Step 5: Data Extraction

Data extraction involves selecting only the relevant variables needed to construct a process mining event log. This typically includes identifiers, activity labels, timestamps, and optional contextual information such as department or provider.

Example: Extract the following structure:

Patient_ID	Activity	Timestamp
P1001	*Biopsy*	*2021-05-03 09:00*

Case study reference: Chapter 4.5.1.3 lists 15 identified event types selected for mining (e.g., first consultation, imaging, biopsy, diagnosis confirmation). These were extracted and renamed into simplified, consistent labels suitable for event log creation (p. 123–124).

7.3.1.6 Step 6: Create Event Log

Use tools like Disco or ProM to transform your cleaned data into a standardised event log format.

Example: Load the event log into Disco where:

Case ID = Patient_ID
Activity = Activity
Timestamp = Timestamp

Case study reference: As described in Chapter 4, Section 4.5.1.5 (p. 127), date-based logs were used to support the modelling of diagnostic and treatment delays. Each log captured key activity names and their corresponding timestamps per patient. The data were tested in Disco to visualise the process flow and to verify the completeness and consistency of the event sequences used for process mining.

7.3.2 Phase 2: Process Mining (Conceptual Modelling)

7.3.2.1 Step 1: Select Algorithm to Generate Process Model

Choose an algorithm based on process complexity, noise, and clarity required.

Example: Use Fuzzy Miner to simplify complex diagnostic pathways and highlight high-level flows through abstraction and significance-based filtering. Use Inductive Miner to ensure a sound, structured model if the event log quality is high and process variation is limited.

Case study reference: In Chapter 4, Section 4.5.2.1 (pp. 129–140), four process discovery algorithms Alpha Miner, Heuristics Miner, Fuzzy Miner, and Inductive Miner were applied to the breast cancer diagnostic event log for comparison. Through visualisation and structural analysis of the resulting process models (Figures 4.4–4.7), Fuzzy Miner was selected for the final implementation due to its ability to abstract and simplify complex patient journeys while preserving key transitions. The model generated by Fuzzy Miner clearly highlighted dominant paths and allowed for tunable abstraction, making it well-suited to capture fragmented and variable real-world pathways, particularly in healthcare scenarios where standardised sequences are uncommon and optional activities may be skipped.

7.3.2.2 Step 2: Generate Initial Process Model

Visualise the discovered model from Disco or ProM.

Example: The model shows:

Registration → Biopsy → Diagnosis → Treatment Planning → First Chemotherapy

Case study reference: In Chapter 4.5.2.2, the Fuzzy Miner output revealed over 150 unique pathway variants, capturing real-world variability in sequencing and delays (p. 129).

7.3.2.3 Step 3: Compare Models from Different Algorithms (New Step)

Run the same log through multiple algorithms (e.g., Inductive, Fuzzy) and compare clarity, completeness, and usability.

Example: Use Fuzzy Miner to group related or minor activities into clusters and highlight dominant flows for easier interpretation. Use Inductive Miner if a clear, fully structured model is needed, especially when the event log is highly regular and consistent. Choose the model that best supports stakeholder understanding and reflects real-world variation.

Case study reference: In Chapter 4, Section 4.5.2.3 (pp. 133–135), process models generated by Fuzzy Miner and Inductive Miner were compared. The Fuzzy Miner output allowed for flexible abstraction, helping to visualise key flows while collapsing less critical details. In contrast, Inductive Miner was found unsuitable for this healthcare data due to its strict structural requirements. It failed to capture optional and alternative diagnostic sequences, such as patients who proceeded directly to biopsy without a mammogram or underwent repeated consultations. As a result, it oversimplified the diagnostic journey, making it less effective in representing unstructured or variable care pathways, which are common in real clinical settings.

7.3.2.4 Step 4: Perform Data Abstraction (New Step)

This step simplifies the process model by applying threshold-based abstraction to reduce visual complexity while maintaining both frequent and clinically important transitions. It supports clearer interpretation without overlooking critical but less common pathways. Use the abstraction feature in tools like Disco to adjust significance and correlation thresholds. This controls which activities and paths are shown in the process model, allowing users to explore and filter the process at varying levels of detail

Example: Apply an 80% threshold in Disco to reduce clutter for simulation modelling, but also review models at 10% or 50% to identify low-frequency yet clinically significant pathways that should be retained.

Case study reference: In Chapter 4, Section 4.5.2.4 (p. 144), abstraction was performed using Fuzzy Miner in Disco with threshold settings of 10%, 50%, and 80%. These

threshold levels were used iteratively to examine different levels of process detail. While higher thresholds simplified the model for simulation readiness, lower thresholds were reviewed to ensure that rare but clinically meaningful pathways, such as skipped imaging or repeated consultations, were considered and validated with expert input before final abstraction.

7.3.2.5 Step 5: Subgroup Classification

Split logs based on variables (e.g., Public vs Private hospitals) to identify variation.

Example: Model A (Private): Biopsy within 3 days

Model B (Public): Biopsy after >14 days

Case study reference: Differences between public and private pathway performance were explored using subgrouping in Chapter 4.3 and demonstrated during abstraction and model comparison in 4.5.2.4 (p. 117–118, p. 144).

7.3.2.6 Step 6: Validate AS-IS Model with Experts

Discuss model steps with clinicians or process expert for confirmation.

Example: Clinicians confirm that “CT Scan” is typically conducted as part of the staging process before the “Treatment Plan” is finalised this correction is applied in the process model to reflect actual clinical practice.

Case study reference: In Chapter 4, Sections 4.5.1.4 and 4.5.2.4 (pp. 125–126, 144), expert oncologists validated the discovered process model by reviewing event sequencing and activity labels. They identified that some events, such as “Treatment” were appearing too early in the model before necessary diagnostic confirmations like biopsy results. Based on their feedback, the model was corrected to align with standard diagnostic and treatment protocols.

7.3.2.7 Step 7: Benchmarking (New Step)

Compare the generated process model with best practices or existing model.

Example: Compare the average time from diagnosis to treatment against existing model benchmark (⩽30 days).

Case study reference: Chapter 4.5.2.4 shows how the discovered process model for kidney care was benchmarked against existing clinical pathways from national renal care models reported in the literature (p. 145).

7.3.3 Phase 2: Process Mining (Parameterisation)

This phase focuses on identifying the quantitative parameters required to configure a discrete event simulation (DES) model. These parameters include timing information, activity frequencies. Both direct and derived parameters are extracted using process mining tools, and then validated with domain experts. A new step—benchmarking—has been added in this research to support external performance comparison.

7.3.3.1 Step 1: Identify Process Parameters

Use Disco to extract key parameters such as activity durations, transition frequencies, and routing probabilities for simulation modelling.

Example: The time interval between “First Consultation” and “Biopsy” was calculated and used as an input parameter in the DES model to reflect real-world delays in the diagnostic process.

Case study reference: In Chapter 4.5.3.1 (p. 146), activity durations were computed by measuring the time between recorded timestamps for each patient event. For example, the time interval between “First Consultation” and “Biopsy” was calculated and used as a parameter in the DES model.

7.3.3.2 Step 2: Use Direct Parameters

Direct parameters can be read directly from the process mining output.

Example: Average waiting time from biopsy to diagnosis = 5.2 days

Case study reference: Chapter 4.5.3.1 notes that the time between diagnosis and treatment averaged 29 days across all cases, which was used directly in the simulation inputs (p. 146).

7.3.3.3 Step 3: Derive Complex Parameters

Calculate advanced parameters such as routing probabilities from the discovered process model.

Example: 46% of patients followed a short, direct path to diagnosis, while others went through repeated testing and consultations, indicating different routing behaviours.

Case study reference: In Chapter 4, Section 4.5.3.2 (p. 147), routing probabilities were derived from the Fuzzy Miner model output. tiple testing stages, showing variations in routing probabilities (p. 147).

7.3.3.4 Step 4: Expert Validation

Review parameters with clinical professionals to check accuracy.

Case study reference: In Chapter 4.5.3.3, simulation input values (e.g., average time, and transition probabilities) were validated through expert feedback from Mexican oncologists to ensure plausibility in a simulated care environment (p. 148).

7.3.3.5 Step 5: Benchmarking (New Step)

Compare time and other parameters with best practices or existing model.

Example: Compare your average time from diagnosis to treatment against existing model benchmark (⩽30 days).

Case study reference: Chapter 5, Section 5.5.2.4 shows how timelines derived from process mining—such as the median time from catheter insertion to first PD dialysis—

were compared against benchmarks from an existing conceptual model developed by collaborators for health economics simulation. This comparison helped validate the derived model structure and assess consistency with clinical expectations (pp. 205–206)

7.3.4 Phase 3: Transforming to DES Model

This phase involves transforming the conceptual process model and parameter data into a working DES model. The model replicates the process logic, patient flows, and timing dynamics of the real system.

7.3.4.1 Step 1: Translate to DES Model

Build the model in SIMUL8, Arena, or AnyLogic using discovered process logic.

Example: The activity "Consultation" was modelled as a work entry point with a defined duration, followed by branching pathways leading to either "Diagnosis" or a follow-up, depending on routing probabilities derived from process mining.

Case study reference: In Chapter 4.5.4.1, SIMUL8 was used to create a simplified simulation model of the abstracted pathways, where activities such as "Consultation," "Diagnosis," and "Treatment Start" were translated into simulation blocks (p. 150).

7.3.4.2 Step 2: Define Logic and Dependencies

Define branching logic and flow conditions.

Example: "Consultation → Biopsy (80%)" vs. "Consultation → Imaging (20%)"

Case study reference: As described in Chapter 4.5.4.2, probabilities for transitions were estimated from pathway frequency analysis in Fuzzy Miner. For example, patients following private pathways were more likely to bypass public referral delays (p. 151).

7.3.4.3 Step 3: Input Parameters

Enter average times, probabilities into the simulation tool.

Case study reference: In Chapter 4.5.4.3, time delays extracted in Phase 4 (e.g., 5–12 days for imaging) were directly entered into SIMUL8's activity blocks. These time values

were derived from observed data and shaped into distributions for simulation. (p. 151–152).

7.3.4.4 Step 4: Develop Simulation Model

Ensure that the pathways and timing are accurately represented in the simulation model.

Case study reference: Chapter 4.5.4.4 describes how the simulation model was designed. The framework also includes provisions for regular model refinement and monitoring (p. 153).

7.3.5 Phase 4: Validation

This final phase involves verifying that the simulation results are accurate, useful, and capable of supporting decisions. Results are compared to historical data, used to test hypothetical changes, and interpreted through stakeholder engagement. A feedback loop is established to ensure continuous model refinement.

7.3.5.1 Step 1: Run Simulation

Run the developed model.

Case study reference: In Chapter 4.5.5.1, the model was executed using patient flow data that reflected the public-private proportions observed in the actual cohort (p. 154).

7.3.5.2 Step 2: Analyse Results

Check outputs such as: Average delay

Case study reference: In Chapter 4.5.5.2, the output revealed that treatment delays could be reduced by 15–20% through minor workflow adjustments, confirming the model's sensitivity to changes in diagnostic timing (p. 155).

7.3.5.3 Step 3: Perform Scenario Analysis

Run “What-if” scenarios.

Case study reference: Chapter 4.5.5.3 explores scenarios designed to reduce diagnostic time. Each scenario was evaluated to assess its potential impact on pathway efficiency, specifically focusing on diagnostic delays and earlier initiation of treatment (p. 156).

7.3.5.4 Step 4: Decision Making

Present results to stakeholders to support strategic decision-making.

Case study reference: In Chapter 4.5.5.4, stakeholders were presented with performance summaries from SIMUL8, which facilitated discussions on key bottlenecks (p. 157).

7.4 Best practice checklist for integrating process mining in DES modelling.

In this section, we presented the checklist for integrating process mining in DES modelling. The checklist serves as a practical guide, outlining key steps and best practices to combine these methodologies.

Section/item	Item number	Recommendation	Questions
Objectives	1	Define clear objectives for integrating process mining into DES modelling.	- Have you defined clear objectives and used metrics like delay, cycle time, or waiting time to measure success?
Data Preparation	2	Data Protocol: Define the protocol for data collection and processing.	-Has the adequacy of the protocol been verified through pilot testing or expert review?
	3	Gather Relevant Data: Identify sources and collect event log data from various sources.	- Have you identified all relevant data sources and collected data accordingly? - Have you verified data completeness and relevance through cross-validation or expert consultation?
	4	Data Profiling: Analyse collected data to understand its characteristics and structure.	- Have you analysed the collected data using profiling tools like dplyr in R-language or Python's pandas profiling? - Have you assessed profiling adequacy by identifying missing or inconsistent values?
	5	Data Cleaning: Remove outliers, correct inconsistencies, and format data.	- Have you addressed data inconsistencies using techniques like IQR filtering? - Have you confirmed the quality of the cleaned data by revalidating against benchmarks?

	6	Missing Data Handling: Replace missing values and consult domain experts.	- Have you used appropriate imputation methods such as replacement with real values or imputation by mean/median/mode. - Have domain experts been consulted to ensure the accuracy of imputed values?
	7	Data Extraction: Select relevant variables and extract data for further analysis.	-Have you selected the relevant variables to achieve your objectives? and have you validated with an expert?
	8	Data Formatting: Format data for compatibility with analysis tools and check timestamp granularity.	- Have you formatted the data for compatibility with analysis tools like Disco or Simul8 (e.g .csv) - Have you checked timestamp granularity t to ensure consistency?
Conceptual Modelling	9	Event Log Structure: Create a structured event log with IDs, activity labels, and timestamps.	- Have you established a structured event log that includes unique IDs, activity labels, and timestamps?
	10	Algorithm Selection: Choose appropriate process mining algorithms.	-Have appropriate process mining algorithms (e.g., Alpha Miner, Heuristic Miner, Fuzzy Miner) been selected to analyse the event log?
	11	Process Mapping: Automatically generate and compare process maps using various algorithms.	- Have automated processes been used to generate process maps using tools like ProM or Disco?

			- Have outcomes been compared across different algorithms to assess their accuracy in capturing process variations?
	12	Complexity Assessment: Identify critical steps and paths within the process.	- Have critical steps and paths been identified to assess process complexity using techniques like frequency analysis? - Have critical steps been validated through stakeholder reviews or expert consultation?
	13	Data Abstraction: Simplify data to extract key insights and reduce complexity.	- Has data been simplified using techniques like thresholding or filtering? - Has the adequacy of data abstraction been verified by expert consultation?
	14	Process Model Subgrouping: Subgroup the process model when required.	- Has the process model been subgrouped for more detailed analysis using clustering techniques? - Has subgroup adequacy been validated by evaluating subgroup insights or patterns?
	15	Validation and Benchmarking: Evaluate and validate results against benchmarks.	- Have results been validated by comparing them to benchmarks? - Have results been verified using statistical tests?

Parameterisation	16	Parameter Identification: Identify activity-related parameters.	- Have activity-related parameters been identified and been confirmed by ensuring alignment with objectives and expected behaviour?
	17	Quantitative Analysis: Derive quantitative insights and calculate required parameters.	- Has quantitative analysis been conducted to derive necessary insights calculation or statistics?
	18	Validation and Benchmarking: Evaluate and validate results against benchmarks.	-Have analysis results been validated and compared to benchmarks or expert review?
Transforming to DES Model	19	Model Translation: Translate process model to DES model.	- Has the process model been translated into a DES model for simulation using software like SIMUL8? - Has the translation been verified by comparing simulation outputs with process insights or historical data?
	20	Activity Logic and Sequencing: Define logic and dependencies.	-Have logic and dependencies for activities been clearly defined and validated by running test simulations or conducting expert reviews?
	21	Input Parameters: Define simulation input parameters, automate repetitive tasks, and enable real-time updates.	- Have all input parameters been identified and defined for the DES model?

	22	Run Simulation: Implement and validate simulation behaviour.	- Has simulation behaviour been implemented using discrete event-driven models? - Has simulation behaviour been validated by comparing outcomes to historical data or benchmarks?
Model Validation	23	Comparative Analysis: Compare DES model with process mining results and expert insights.	- Has a comprehensive comparison between the DES model and process mining results been conducted using statistical tests or expert reviews? - Have domain expert insights been incorporated to validate the model's alignment with real-world scenarios?
	24	Discrepancy Identification: Address inconsistencies and fine-tune the model.	- Have discrepancies between DES and process mining results been identified using by error tracking?
Scenario Analysis	25	Scenario Generation: Create scenarios.	- Have scenarios been generated to test the DES model under various conditions? -Has scenario adequacy been confirmed by testing under multiple conditions or obtaining expert feedback?
	26	Simulation Execution: Run simulations for scenarios.	- Have simulations been executed successfully without issues?
	27	Outcome Analysis: Analyse simulation results.	- Have outcomes been analysed to ensure alignment with expectations?

CHAPTER 8

REFLECTIONS, FUTURE WORK, AND CONCLUSION

8.1 Key contributions

In this section, the key contributions arising from the PhD research are presented and discussed.

8.1.1 Developed and enhanced a step-by-step framework for integrating process mining into DES modelling.

The primary contribution of this PhD thesis is development of a structured framework for integrating process mining and DES modelling. This framework encompasses four key phases: data preparation, conceptual modelling, parameterisation, DES model development, and model validation.

The framework emphasizes data quality and adaptability. It includes steps for data protocol, data profiling, data abstraction and validation. It acknowledges the importance of algorithm selection and introduces subgrouping for complex processes.

The framework also addresses practical challenges with activity synchronization, dependencies, and automation. Its flexibility allows tailoring to specific contexts, enhancing its relevance across industries and cases. While certain aspects like algorithm comparison could be explored further, the strengths of the framework lie in its responsiveness to multiple case studies and its potential to provide generalized guidance for broader applications.

8.1.2 Implemented the framework in multiple clinical case studies.

The thesis makes practical contributions by implementing the developed framework in real-world case studies. Through these case studies, the effectiveness and applicability of the framework are evaluated, providing insights into its strengths,

limitations, and areas for improvement. By applying the framework to multiple clinical processes, such as breast cancer pathways and peritoneal dialysis, the thesis demonstrates the adaptability of the framework in different domains. The findings from the case studies contribute to refining and enhancing the framework, ensuring its practicality and relevance in real-world scenarios. This empirical validation strengthens the confidence in the framework and its potential for successful integration of process mining and DES modelling.

8.1.3 Comparison of process mining with existing traditional modelling approach.

We conducted a comparison of process mining results applied in the context of peritoneal dialysis against an traditional modelling approach. The objective of this comparative analysis was to assess the comprehensiveness and accuracy of the process mining approach. By comparing the outcomes generated by process mining with those derived from the traditional model, we were able to validate the process mining results in terms of the comprehensiveness and accuracy of process maps and parameters. This validation significantly contributes to a broader understanding of how process mining can be effectively utilized to enhance the efficiency and effectiveness of clinical processes.

8.1.4 Developed a best practice checklist for integrating process mining into DES modelling.

After a comprehensive review of literature, development of the framework and implementation of the case study, this PhD thesis ends with a best practice checklist to integrate process mining with Discrete Event Simulation (DES) modelling. This checklist is a practical guide for researchers who would like to use the process mining and DES modelling in an effective manner. It provides a methodology to drive users through data preparation, conceptual modelling, parameterisation, DES model construction, validation, and scenario modelling. This contribution has a strong practical value, as it allows using process mining and DES modelling for process optimisation and data-driven decision-making.

8.2 Reflections

This chapter reflects upon the objectives accomplished within the context of the PhD study. It brings attention to the limitations encountered and offers suggestions for future work, ultimately reaching the study's conclusion.

8.2.1 Reflection on Objective 1: Literature review on integration of process mining in DES in past studies.

In this section, we discuss the integration of process mining into discrete event simulation (DES) based on a thorough literature review, focusing on the achievement of Objective 1. The primary objective was to gain a deeper understanding of the current state of knowledge in the field and identify valuable insights, trends, and gaps related to the integration of these two methodologies.

The general steps for integrating process mining into discrete event simulation (DES) modelling have been widely discussed in the literature (W. M. P. van der Aalst, 2018; A. Rozinat et al., 2009; Mans et al., 2013). These steps involve data preparation, conceptual modelling, parameterisation, transformation into a DES model, model validation, and scenario analysis. The consensus on these steps demonstrates a collective understanding of the core processes involved in achieving integration.

However, the literature and this reflection both underscore the challenges related to data availability and quality (Mesabbah et al., 2019; Măruşter & Van Beest, 2009). Inconsistencies, missing data, or limited coverage in event log data can compromise the accuracy and reliability of the integrated approach. Addressing these data-related challenges requires careful consideration and the development of appropriate, data cleaning and preprocessing techniques, a sentiment echoed by the literature.

Validation of the conceptual model and its parameters heavily relies on expert input. Experts provide domain knowledge and real-world insights, ensuring the model's accuracy and applicability. Collaboration between experts and technology is considered crucial for a robust validation process.

Effective data transformation is emphasized to bridge the gap between process mining data and DES models. Maintaining high standards of data consistency and

coherence throughout this transformation is essential, and steps such as "Activity synchronization and dependencies" and "Automation for repetitive tasks" align with best practices. Automation is a key aspect to streamline the transformation process.

This reflection demonstrates an awareness of the literature and suggests enhancements to the integration process based on the identified limitations. Introduction of new steps like data protocol, data profiling, and a comparison of process models resulting from different algorithms showcases a proactive approach to addressing limitations and advancing the integration process. The assessment of their practicality through case studies aligns with established research methodologies, combining theoretical insights with practical validation.

8.2.2 Reflection on Objective 2: Developed a Step-by-Step Framework.

The provided enhanced step-by-step framework for integrating process mining and Discrete Event Simulation (DES) stands as a structured and systematic approach, offering a comprehensive solution to the intricacies involved in data-driven modelling. The framework unfolds in five essential phases: data preparation, conceptual modelling, parameterisation, DES model development, and model validation and scenario modelling. These phases align with the general steps observed in integrating process mining and DES, as highlighted by many authors (W. M. P. van der Aalst, 2018; A. Rozinat et al., 2009).

One of the notable strengths of this framework lies in its adaptability, particularly as influenced by insights from literature and practical case studies, notably in the clinical domain involving breast cancer and kidney-peritoneal dialysis datasets.

The emphasis on addressing data quality issues within the framework is a significant contribution, aligning with best practices and findings in the literature. The inclusion of data protocol and data profiling as critical steps underlines the importance of leveraging high-quality datasets for meaningful insights. Data profiling, specifically, is crucial in comprehending data reliability and usability through analysis and summarization of key characteristics and quality indicators. Additionally, acknowledging

the potential role of data visualisation in mitigating data quality concerns aligns with recognised practices, even though it may be of lower priority.

The integration of data abstraction into the conceptual modelling phase reflects the framework's commitment to incorporating best practices. Recognizing the significance of algorithm selection in this phase aligns with existing literature that underscores the impact of algorithms on model accuracy and efficiency.

The introduction of a step for subgrouping or classifying process models within conceptual modelling is a pragmatic addition, aligning with the practice of breaking down complex processes into manageable subgroups to enhance understanding. Moreover, the inclusion of steps for expert validation and benchmarking ensures the robustness and reliability of the models before transitioning into DES modelling.

Considering activity synchronization, dependencies, and automation for repetitive tasks during the transformation from conceptual models to DES modelling reflects a nuanced understanding of practical challenges in this domain.

A key strength of the framework lies in its acknowledgment that the steps must be adaptable to suit the unique characteristics of each case study. This adaptability ensures the framework's relevance and broad applicability across diverse contexts and aligns with the notion of seeking common patterns and principles that can be generalized into a broader integration framework.

8.2.3 Reflections on Objective 3: Perform case studies using the step-by-step framework.

In this section, we engage in a meaningful discussion centred on the execution of case studies using the step-by-step framework outlined in this study (Objective 3). The main objective was to assess how effective and practical this framework was in real-life settings, by analysing two case studies in a healthcare setting.

In the case studies, process mining was used in the breast cancer and kidney peritoneal dialysis pathways to derive useful information that would help in the

optimisation of patient care. The case study of the breast cancer pathway, in particular, brought out differences between the durations of different stages of the pathway.

These case studies revealed some important limitations and challenges. However, one major issue was data availability and quality, which is a common problem for healthcare analytics. Data preparation and standardization required extensive efforts that were also outlined in the literature as an issue of healthcare data integration and preparation.

In terms of the kidney peritoneal dialysis case study, it is evident that there are differences in the time to start dialysis and infection rates depending on the method of catheter insertion. This analysis coheres with the overall tendency of using data analytics to improve medical practices and results. The results emphasized the advantages of the medical approach, which included a shortened time to the start of dialysis and a lower rate of infections. Such insights heavily influence clinical decision-making and patient outcomes, serving to help advance healthcare delivery.

The case studies provided a critical appraisal of the performance of the developed framework in analysing healthcare pathways. Although they presented valuable insights about health sector, they also highlighted issues of data quality and availability. Overcoming these obstacles and improving the framework through a case study experience can result in a sustainable and generalized approach, which is consistent with the dynamic world of health analytics.

8.2.4 Reflections On objective 4: The best practice checklist

The best practice checklist for process mining and discrete event simulation (DES) integration provides a systematic approach for smooth integration. This discussion is intended to enhance the checklist by harmonizing it with other methodologies, highlighting its strengths, and defining the areas for further improvement.

In the first phase, Data Preparation, the focus on a precise Data Protocol aligns with the existing standards for reliable data capturing and processing. Data collection from various sources supports the requirement for broad datasets in process mining. The

checklist is recognizing the importance of Data Profiling in understanding the data characteristics. Nevertheless, it can be improved by adding methods for handling incorrect data, which is a well-known issue of event log data.

In Conceptual Modelling, the addition of the steps of Data Abstraction follows the best practices aimed at simplifying complex models for analysis. Nevertheless, there is a possibility to analyse the recent developments such as cognitive process mining and further improve the depth and understanding of the modelled process. In addition, the checklist could also benefit from incorporating methods for dynamic process discovery, allowing for real-time updates to the model.

In Parameterisation, the checklist correctly identifies the need for quantitative analysis. Machine learning approaches for automated parameter identification could further increase the accuracy as well as the quality of parameter. The transition to a DES Model in Phase 3 is consistent with traditional simulation modelling methodologies. The validation step is important, and the checklist could be improved by adding advanced modelling validation to increase model accuracy.

In Phase 4, Model Validation, although comparative analysis is important, incorporating methods such as cross-validation would lead to a more detailed evaluation of the model's performance. Moreover, using statistical methods for model validation would fit with contemporary validation techniques.

In Scenario Analysis, the checklist correctly points to using the validated DES model for simulating different scenarios. Techniques such as scenario discovery might be used to provide more detailed knowledge of complex systems' behaviour and improve scenario analysis.

The focus of the checklist on data quality and the significance of careful data preparation is aligned with established best practices. Moreover, offering advice on the algorithm choice depending on the use cases, as recommended, would improve the checklist's applicability.

8.3 Strengths

The main strength of this PhD study lies in developing a structured and methodical approach to integrating process mining into Discrete Event Simulation (DES) modelling. The development of a step-by-step framework for this integration provides an enhanced roadmap, facilitating integration of process mining techniques into DES. This systematic framework enhances the efficiency of the modelling process by guiding practitioners through each essential stage.

In addition to the theoretical framework, the study demonstrates its practical application through the implementation of the developed approach in multiple clinical case studies. This practical integration showcases the real-world adaptability and effectiveness of the proposed framework. By implementing the framework in diverse clinical contexts, the study validates its applicability and potential benefits. This hands-on validation underscores the viability of integrating process mining into DES modelling, particularly within complex clinical processes, setting a strong foundation for future research and practical implementations.

Moreover, comparative analysis of the process mining results with existing models ensures the accuracy and reliability of the integrated approach. By evaluating the outcomes against established models, the study substantiates the improvements and advancements achieved through the incorporation of process mining techniques into DES modelling. This comparative validation enriches the credibility and contributes to the body of knowledge in the field of process mining and DES modelling.

8.4 Limitations

One limitation of this PhD study is the absence of resource/capacity modelling within the integrated framework. While the study focuses on integrating process mining into Discrete Event Simulation (DES) modelling and successfully develops a comprehensive framework, incorporating resource/capacity considerations into the framework is beyond the scope of the PhD and as such, does not delve into resource/capacity modelling. This omission restricts the study from providing a complete view of the simulation process, potentially overlooking crucial factors that influence system performance and resource allocation.

Another limitation lies in the scope of process mining within the study. The study primarily focuses on process discovery, a vital aspect of process mining. However, it does not extensively explore other facets of process mining, such as conformance checking or performance analysis. While the study's emphasis on process discovery aligns with its objectives, a more comprehensive exploration of diverse process mining techniques could enrich the findings and broaden the study's applicability.

Furthermore, the evaluation of the step-by-step framework developed in this study face limitations due to the restricted availability of experts with experience in both the process mining and DES modelling domains. Reviewing the framework requires expertise from both fields, making it challenging to gather a sufficiently diverse and expert panel for thorough evaluation. This limitation could potentially impact the comprehensiveness and depth of the framework evaluation, highlighting the need for a broader pool of experts for a more comprehensive assessment.

These limitations, while important to acknowledge, also present opportunities for future research and refinement of the integrated approach, ultimately contributing to a more robust and all-encompassing integration of process mining into DES modelling.

8.5 Future work

In this Ph.D. research, our primary focus has been on time to event DES modelling. An exciting avenue for future work involves extending our framework to include resource modelling. This addition aims to enhance the realism of our simulations and address resource allocation challenges. By integrating resource modelling, we can optimise resource allocation, improve process performance, and contribute to more robust process optimisation methodologies.

Another promising area for future research is the segmentation of patient data based on characteristics such as gender, age, or other relevant factors within the healthcare system. This aims to identify distinct patterns or subgroups based on patient characteristics.

To further validate our proposed framework, future research could involve testing it in various industries and domains to assess its generalizability and robustness. By

applying the framework in different settings, researchers can ensure its applicability and effectiveness in diverse contexts, supporting decision-making across various industries.

Future research could also explore the integration of machine learning and artificial intelligence techniques with process mining. This integration would enable more precise predictions and allow the system to learn from historical data, improving its adaptability and responsiveness. This approach holds the potential to create more sophisticated and intelligent modelling systems.

Additionally, to comprehensively capture complex system behaviour, future research could investigate the integration of process mining with other simulation techniques, such as system dynamics or agent-based modelling. Combining these methodologies would result in hybrid models capable of capturing intricate interactions and relationships within a system. This interdisciplinary approach can lead to a deeper understanding of complex systems and enable more accurate simulation outcomes. Such hybrid models could offer valuable insights into system dynamics that are not achievable through single-method simulations.

8.6 Conclusion

In conclusion, this Ph.D. thesis has developed an integrated framework for combining process mining with discrete event simulation (DES) modelling. Key findings highlight the critical role of data quality and effective data cleaning techniques. The parameterisation phase should consider practical data challenges and address potential biases, while transitioning from a conceptual model to a DES model requires acknowledgment of the iterative nature of the process. Model validation is a critical, necessitating strategies for handling conflicting results. While the provided checklist offers structure, acknowledging real-world complexities and ensuring iterative approach is essential for robust results. This research advances the field, laying the groundwork for future enhancements in integrating process mining with DES modelling.

REFERENCES

Aalst, W. Van Der. (2016). *Process Mining: Data Science in Action* (Heidelberg (ed.); Vol. 2). Springer.

Abohamad Waleed, Ramy Ahmed, A., & Amr. (2017). *A Hybrid Process-Mining Approach For Simulation Modelling*. 1527–1538.

Ajah, I. A., & Nweke, H. F. (2019). *Big Data and Business Analytics : Trends , Platforms , Success Factors and Applications*.

Aloui, Thomas Franck Paolo Bercelli Saber, V. A. (2019). A generic framework to analyse and improve patient pathways within a healthcare network using process mining and discrete-event simulation. *Proceedings of the 2020 Winter Simulation Conference*, *ii*, 9–25. https://doi.org/10.1109/WSC48552.2020.9384021

Antonio, P., Safanelli, J., Liberato, R. B., Moro, C. H. C., Eduardo, P., Moro, C., Freitas, E. De, Loures, R., Alves, E., & Santos, P. (2022). *A hybrid model to support decision making in the stroke clinical pathway*. *120*(June). https://doi.org/10.1016/j.simpat.2022.102602

Antunes, B. B. P., Manresa, A., Bastos, L. S. L., Marchesi, J. F., & Hamacher, S. (2019). A Solution Framework Based on Process Mining, Optimisation, and Discrete-Event Simulation to Improve Queue Performance in an Emergency Department. *Lecture Notes in Business Information Processing*, *362 LNBIP*, 583–594. https://doi.org/10.1007/978-3-030-37453-2_47

Brailsford, S., & Hilton, N. (2001). A comparison of discrete event simulation and system dynamics for modelling health care systems. *Proceedings from ORAHS 2000*, 18–39. http://eprints.soton.ac.uk/35689/1/glasgow_paper.pdf

Braun, M. M., & Khayat, M. (2021). Kidney Disease: Chronic Kidney Disease. *FP Essentials*, *509*(Suppl 1), 20–25.

Bright, K., Barghash, M., Donach, M., Gutiérrez de la Barrera, M., Schneider, R. J., & Formenti, S. C. (2011). The role of health system factors in delaying final diagnosis and treatment of breast cancer in Mexico City, Mexico. The Breast,

20(Suppl 2), S54–S59.

Buijs, J. C. A. M., van Dongen, B. F., & van der Aalst, W. M. P. (2012). On the Role of Fitness , Precision , Generalization. *OTM Confederated International Conferences On the Move to Meaningful Internet Systems*, 305–322.

Buss, A., & Al Rowaei, A. (2010). A comparison of the accuracy of discrete event and discrete time. *Proceedings - Winter Simulation Conference*, 1468–1477. https://doi.org/10.1109/WSC.2010.5679045

Chen, J., Chiang, R. H., & Storey, V. C. (2012). *Business intelligence and analytics: From big data to big impact*. *36*(4), 1–24. papers3://publication/uuid/42576320-BCA9-4A5E-AA23-0C74A0CC26D3

Cho, M., Song, M., Yoo, S., & Reijers, H. A. (2019). An Evidence-Based Decision Support Framework for Clinician Medical Scheduling. *IEEE Access*, *7*, 15239–15249. https://doi.org/10.1109/ACCESS.2019.2894116

Collins, A. J., Cotter, T. S., & Lynch, C. J. (2020). *EDUCATION IN ANALYTICS NEEDED FOR THE MODELLING & SIMULATION PROCESS*. 3236–3247.

Comas, M. (2005). *Implementation of a simulation model to estimate the demand and waiting time for elective surgery: cataract surgery*. *February*.

De Medeiros, A. K. A., Weijters, A. J. M. M., & Van Der Aalst, W. M. P. (2007). Genetic process mining: An experimental evaluation. *Data Mining and Knowledge Discovery*, *14*(2), 245–304. https://doi.org/10.1007/s10618-006-0061-7

De Weerdt, J. (2012). *Business Process Discovery: New Techniques and Applications*. *403*, 229. http://www.kuleuven.be/doctoraatsverdediging/fiches/3H10/3H100298.htm

Elgendy, N., & Elragal, A. (2016). Big Data Analytics in Support of the Decision Making Process. *Procedia - Procedia Computer Science*, *100*, 1071–1084. https://doi.org/10.1016/j.procs.2016.09.251

Fernandez-Llatas, C., Munoz-Gama, J., Martin, N., Johnson, O., Sepulveda, M., & Helm, E. (2021). *Process Mining in Healthcare*. 41–52.

https://doi.org/10.1007/978-3-030-53993-1_4

Fontanili, F., Lamine, E., & Lamothe, J. (2022). *A discrete event simulation-based methodology for building a digital twin of patient pathways in the hospital for near real- time monitoring and predictive simulation [version 1 ; peer review : 1 approved with reservations] Abdallah Karakra*. 1–18.

Fotheringham, J., Solis-Trapala, I., Briggs, V., Lambie, M., McCullough, K., Dunn, L., Rawdin, A., Hill, H., Wailloo, A., Davies, S., & Wilkie, M. (2023). Catheter event rates in medical compared to surgical peritoneal dialysis catheter insertion. *Kidney International Reports*, 1–11. https://doi.org/10.1016/j.ekir.2023.09.015

Friederich, J., Francis, D. P., Lazarova-Molnar, S., & Mohamed, N. (2022). A framework for data-driven digital twins for smart manufacturing. *Computers in Industry*, *136*, 103586. https://doi.org/10.1016/j.compind.2021.103586

Friederich, J., Lugaresi, G., Lazarova-Molnar, S., & Matta, A. (2022). Process Mining for Dynamic Modelling of Smart Manufacturing Systems: Data Requirements. *Procedia CIRP*, *107*, 546–551. https://doi.org/10.1016/j.procir.2022.05.023

Gatta, R., Vallati, M., Fernandez-Llatas, C., Martinez-Millana, A., Orini, S., Sacchi, L., Lenkowicz, J., Marcos, M., Munoz-Gama, J., Cuendet, M. A., de Bari, B., Marco-Ruiz, L., Stefanini, A., Valero-Ramon, Z., Michielin, O., Lapinskas, T., Montvila, A., Martin, N., Tavazzi, E., & Castellano, M. (2020). What role can process mining play in recurrent clinical guidelines issues? A position paper. *International Journal of Environmental Research and Public Health*, *17*(18), 1–19. https://doi.org/10.3390/ijerph17186616

Greasley, A. (2018). Using analytics with discrete-event simulation. *Simulation Series*, *50*(10), 145–156. https://doi.org/10.22360/summersim.2018.scsc.022

Günal, M. M., & Pidd, M. (2010). Discrete event simulation for performance modelling in health care: A review of the literature. *Journal of Simulation*, *4*(1), 42–51. https://doi.org/10.1057/jos.2009.25

Gunn, B. M. (2017). *Inference of Resource-based Simulation Models from Process Event-log Data by*. http://dro.deakin.edu.au/eserv/DU:30110902/gunn-

inferenceofresource-2018.pdf

Günther, C. W., & Van Der Aalst, W. M. P. (2007). Fuzzy mining - Adaptive process simplification based on multi-perspective metrics. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *4714 LNCS*, 328–343. https://doi.org/10.1007/978-3-540-75183-0_24

Gupta, E. (2014). Process mining algorithms. *International Journal of Advance Research In Science And Engineering*, *3*(11), 401–412.

Halawa, F., Chalil Madathil, S., & Khasawneh, M. T. (2021). Integrated framework of process mining and simulation–optimisation for pod structured clinical layout design. *Expert Systems with Applications*, *185*(July), 115696. https://doi.org/10.1016/j.eswa.2021.115696

Hamrock, E., Medicine, J. H., Parks, J., Medicine, J. H., Scheulen, J. J., & Levin, S. (2013). *Discrete Event Simulation for Healthcare Organisations : A Tool for Decision Making*. *April 2016*. https://doi.org/10.1097/00115514-201303000-00007

Hanf, V., & Kreienberg, R. (2020). *WHO report on cancer: setting priorities, investing wisely and providing care for all*. https://doi.org/10.1007/978-3-662-11496-4_24

Hitt, L., & Wu, L. (n.d.). *Data Analytics Skills and the Corporate Value of Social Media*.

Ito, S., Vymětal, D., & Šperka, R. (2020). Process mining approach to formal business process modelling and verification: a case study. *Journal of Modelling in Management*. https://doi.org/10.1108/JM2-03-2020-0077

Jadric, M. (2019). *Framework for discrete-event simulation modelling supported by LMS data and process mining*.

Jadrić, M., Pašalić, I. N., & Ćukušić, M. (2020). Process mining contributions to discreteevent simulation modelling. *Business Systems Research*, *11*(2), 51–72. https://doi.org/10.2478/bsrj-2020-0015

Jans, M., Soffer, P., & Jouck, T. (2019). Building a valuable event log for process

mining: an experimental exploration of a guided process. *Enterprise Information Systems*, *13*(5), 601–630. https://doi.org/10.1080/17517575.2019.1587788

Jerry Banks, John S.Carson, Barry L.Nelson, D. M. N. (2010). *Discrete Event System Simulation (Fifth Edition) .pdf*.

Johnson, O. A., Ba Dhafari, T., Kurniati, A., Fox, F., & Rojas, E. (2019a). The ClearPath Method for Care Pathway Process Mining and Simulation. *Lecture Notes in Business Information Processing*, *342*, 239–250. https://doi.org/10.1007/978-3-030-11641-5_19

Johnson, O. A., Ba Dhafari, T., Kurniati, A., Fox, F., & Rojas, E. (2019b). The ClearPath Method for Care Pathway Process Mining and Simulation. In *Lecture Notes in Business Information Processing* (Vol. 342). Springer International Publishing. https://doi.org/10.1007/978-3-030-11641-5_19

Karnon, J., Stahl, J., Brennan, A., Caro, J. J., Mar, J., & Möller, J. (2012). Modelling using discrete event simulation: A report of the ISPOR-SMDM modelling good research practices task force-4. *Medical Decision Making*, *32*(5), 701–711. https://doi.org/10.1177/0272989X12455462

Khodyrev, I., & Popova, S. (2014). Discrete modelling and simulation of business processes using event logs. *Procedia Computer Science*, *29*, 322–331. https://doi.org/10.1016/j.procs.2014.05.029

Kovalchuk, S. V., Funkner, A. A., Metsker, O. G., & Yakovlev, A. N. (2018). Simulation of patient flow in multiple healthcare units using process and data mining techniques for model identification. *Journal of Biomedical Informatics*, *82*(May), 128–142. https://doi.org/10.1016/j.jbi.2018.05.004

Kunc, M., & Brien, F. A. O. (2019). The role of business analytics in supporting strategy processes : Opportunities and limitations. *Journal of the Operational Research Society*, *5682*, 1–12. https://doi.org/10.1080/01605682.2018.1475104

Lakshmanan, G. T., Khalaf, R., & Watson, I. B. M. T. J. (2012). Leveraging Process Mining Techniques. *IT Professional*, *15*(5), 22–30.

Lamine, E., Fontanili, F., Di Mascolo, M., & Pingaud, H. (2015). Improving the Management of an Emergency Call Service by Combining Process Mining and Discrete Event Simulation Approaches. *IFIP Advances in Information and Communication Technology*, *463*, 243–254. https://doi.org/10.1007/978-3-319-24141-8

Law, A. M. (2015). *Simulation Modelling and Analysis*.

Lim, H. S., Sun, W., Parivar, K., & Wang, D. (2019). Predicting Overall Survival and Progression-Free Survival Using Tumor Dynamics in Advanced Breast Cancer Patients. *AAPS Journal*, *21*(2), 1–12. https://doi.org/10.1208/s12248-018-0290-x

Lord, J., Willis, S., Eatock, J., Tappenden, P., Trapero-Bertran, M., Miners, A., Crossan, C., Westby, M., Anagnostou, A., Taylor, S., Mavranezouli, I., Wonderling, D., Alderson, P., & Ruiz, F. (2013). Economic modelling of diagnostic and treatment pathways in National Institute for Health and Care Excellence clinical guidelines: The Modelling Algorithm Pathways in Guidelines (MAPGuide) project. *Health Technology Assessment*, *17*(58), 1–150. https://doi.org/10.3310/hta17580

Lugaresi, G., & Matta, A. (2021a). Automated manufacturing system discovery and digital twin generation. *Journal of Manufacturing Systems*, *59*(December 2020), 51–66. https://doi.org/10.1016/j.jmsy.2021.01.005

Lugaresi, G., & Matta, A. (2021b). Discovery and digital model generation for manufacturing systems with assembly operations. *IEEE International Conference on Automation Science and Engineering*, *2021-Augus*(August), 752–757. https://doi.org/10.1109/CASE49439.2021.9551479

Lugaresi, G., Zanotti, M., Tarasconi, Di., & Matta, A. (2019). Manufacturing systems mining: Generation of real-time discrete event simulation models. *Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics*, *2019-Octob*(3), 415–420. https://doi.org/10.1109/SMC.2019.8914025

Mans, R., Reijers, H., Wismeijer, D., & Van Genuchten, M. (2013). A process-oriented methodology for evaluating the impact of IT: A proposal and an application in

healthcare. *Information Systems*, *38*(8), 1097–1115. https://doi.org/10.1016/j.is.2013.06.005

Marin-Castro, H. M., & Tello-Leal, E. (2021). Event log preprocessing for process mining: A review. *Applied Sciences (Switzerland)*, *11*(22), 1–29. https://doi.org/10.3390/app112210556

Martin, N., Depaire, B., & Caris, A. (2015). The use of process mining in business process simulation model construction structuring the field. *Business and Information Systems Engineering*, *58*(1), 73–87. https://doi.org/10.1007/s12599-015-0410-4

Măruşter, L., & Van Beest, N. R. T. P. (2009). Redesigning business processes: A methodology based on simulation and process mining techniques. *Knowledge and Information Systems*, *21*(3), 267–297. https://doi.org/10.1007/s10115-009-0224-0

Masakane, I., Nakai, S., Ogata, S., Kimata, N., Hanafusa, N., Hamano, T., Wakai, K., Wada, A., & Nitta, K. (2015). An Overview of Regular Dialysis Treatment in Japan (As of 31 December 2013). *Therapeutic Apheresis and Dialysis*, *19*(6), 540–574. https://doi.org/10.1111/1744-9987.12378

Mayr, M., Luftensteiner, S., & Chasparis, G. C. (2022). Abstracting Process Mining Event Logs From Process-State Data To Abstracting Process Mining Event Logs From Process-State Data To Monitor Control-Flow Of Industrial Manufacturing Processes. *Procedia Computer Science*, *200*, 1442–1450. https://doi.org/10.1016/j.procs.2022.01.345

Mesabbah, M., Abo-Hamad, W., & McKeever, S. (2019). A Hybrid Process Mining Framework for Automated Simulation Modelling for Healthcare. *Proceedings - Winter Simulation Conference*, *2019-Decem*, 1094–1102. https://doi.org/10.1109/WSC40007.2019.9004800

Mesabbah, M., & McKeever, S. (2019). Presenting a hybrid processing mining framework for automated simulation model generation. *Proceedings - Winter Simulation Conference*, *2018-Decem*, 1370–1381. https://doi.org/10.1109/WSC.2018.8632467

Mohammadi, I., Wu, H., Turkcan, A., Toscos, T., & Doebbeling, B. N. (2018). *Data Analytics and Modelling for Appointment No-show in Community Health Centres*. https://doi.org/10.1177/2150132718811692

Murillas, G. L. De. (2019). *Process Mining on Databases* (Issue 2019).

Nakatumba, J., Westergaard, M., & Van Der Aalst, W. M. P. (2012). Generating event logs with workload-dependent speeds from simulation models. *Lecture Notes in Business Information Processing*, *112 LNBIP*(January 2014), 383–397. https://doi.org/10.1007/978-3-642-31069-0_31

Nikkhah Ghamsari, B. (2017). Modelling and Improving Patient flow at an Emergency Department in a Local Hospital Using Discrete Event Simulation. *ProQuest Dissertations and Theses*, *December*, 147. https://search.proquest.com/docview/2025949855?accountid=188395

Novák, P., & Vyskoˇcil, J. (2022). Digitalized Automation Engineering of Industry 4.0 Production Systems and Their Tight Cooperation with Digital Twins. *MDPI*. https://doi.org/https://doi.org/10.3390/pr10020404

Ohlhorst, F. (2012). *Big Data Analytics: Turning Big Data into Big Money.* (Issue October).

Overbeck, L., Brützel, O., Teufel, M., Stricker, N., Kuhnle, A., Lanza, G., & Siadat, A. (2021). Continuous adaption through real data twins analysis simulation models into digital digital twins a the functional A new methodology to analyse and physical architecture *, for exi. *Procedia CIRP*, *104*, 98–103. https://doi.org/10.1016/j.procir.2021.11.017

Pegoraro, F., Caroline, A., Marques, C., Lima, G. M., Consolação, R., Rank, I. C., Gonçalves, N., De Bessa, F., Lopes, R., Poletto, K., Tatielle, S., Gomes, M., Cibene, J., Borges, M., Falcão De Sousa, S., Oliveira Coelho, R., & Laus, F. W. (2022). Framework for Emergency Department Management: A Proposal Based on Process Mining and Simulation. *American Academic Scientific Research Journal for Engineering*, *85*(1), 223–234. http://asrjetsjournal.org/

Petitdemange, E., Fontanili, F., Lamine, E., Lauras, M., & Okongwu, U. (2020). A

Tool-Based Framework to Assess and Challenge the Responsiveness of Emergency Call Centres. *IEEE Transactions on Engineering Management*, *67*(3), 568–581. https://doi.org/10.1109/TEM.2019.2954013

Phan, R., Augusto, V., Martin, D., & Sarazin, M. (2019). Clinical Pathway Analysis Using Process Mining and Discrete-Event Simulation: An Application to Incisional Hernia. *Proceedings - Winter Simulation Conference*, *2019-Decem*, 1172–1183. https://doi.org/10.1109/WSC40007.2019.9004944

Pourbafrani, M., Jiao, S., & Aalst, W. M. P. Van Der. (2021). SIMPT : Process Improvement Using Interactive Simulation of Time-aware Process Trees. *International Conference on Research Challenges in Information Science*, 588–594.

Pourbafrani, M., & van der Aalst, W. M. P. (2022). *Interactive Process Improvement using Simulation of Enriched Process Trees*. 1–15. http://arxiv.org/abs/2201.07755

Prackwieser, C., & Buchmann, R. (2013). *Towards a Generic Hybrid Simulation Algorithm Based on a Semantic Mapping and Rule Evaluation Approach*. 147–148.

Rashid, K. M., & Louis, J. (2022). Integrating Process Mining with Discrete-Event Simulation for Dynamic Productivity Estimation in Heavy Civil Construction Operations. *Algorithms*, *15*(5), 173. https://doi.org/10.3390/a15050173

Robinson, S. (2003). *Simulation: The Practice of Model*.

Rockney, R. E., & Kleeman, C. R. (n.d.). *PERITONEAL DIALYSIS TECHNIQUE AND APPLICATIONS*.

Rodič, B. (2017). *Industry 4.0 and the New Simulation Modelling Paradigm*. *50*(3), 193–207. https://doi.org/10.1515/orga-2017-0017

Rozinat, a, W.M.P. van der Aalst(Promotor), & (Co-promotor), a. J. M. M. W. (2010). Process mining : conformance and extension. In *TU Eindhoven, Diss*.

Rozinat, A., Mans, R. S., Song, M., & van der Aalst, W. M. P. (2009). Discovering simulation models. *Information Systems*, *34*(3), 305–327.

https://doi.org/10.1016/j.is.2008.09.002

Saravanan, M. S., & Rama Sree, R. J. (2011). A Role of Heuristics Miner Algorithm in the Business Process System. *International Journal of Computer Technology and Applications*, *2*(2), 340–344.

Selig, H. (2017). *Continuous Event Log Extraction for Process Mining*. 1–60. http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1119380&dswid=-9352#sthash.9YFCQO9U.dpbs

Tamburis, O. (2019). Bridging the gap between process mining and des modelling in the healthcare domain. *2019 7th E-Health and Bioengineering Conference, EHB 2019*, 23–26. https://doi.org/10.1109/EHB47216.2019.8969912

Tamburis, O., & Esposito, C. (2020a). Process mining as support to simulation modelling: A hospital-based case study. *Simulation Modelling Practice and Theory*, *104*(July), 102149. https://doi.org/10.1016/j.simpat.2020.102149

Tamburis, O., & Esposito, C. (2020b). Process mining as support to simulation modelling: A hospital-based case study. *Simulation Modelling Practice and Theory*, *104*(June), 102149. https://doi.org/10.1016/j.simpat.2020.102149

Tangkawarow, I. R. H. T., Sarno, R., & Fauzan, A. C. (2018). Evaluation the Performance of Tax Determination Using Discrete Event Simulation. *2018 2nd International Conference on Informatics and Computational Sciences, ICICoS 2018*, 23–28. https://doi.org/10.1109/ICICOS.2018.8621819

Teitelbaum, I., & Burkart, J. (2003). Peritoneal Dialysis. *American Journal of Kidney Diseases*, *42*(5), 1082–1096. https://doi.org/10.1016/j.ajkd.2003.08.036

Tullavardhana, T., Akranurakkul, P., Ungkitphaiboon, W., & Songtish, D. (2016). Surgical versus percutaneous techniques for peritoneal dialysis catheter placement: A meta-analysis of the outcomes. *Annals of Medicine and Surgery*, *10*, 11–18. https://doi.org/10.1016/j.amsu.2016.07.007

Turner, C. J., Tiwari, A., Olaiya, R., & Xu, Y. (2012). Process mining: From theory to practice. *Business Process Management Journal*, *18*(3), 493–512.

https://doi.org/10.1108/14637151211232669

Unger-Saldaña, K. (2014). Challenges to the early diagnosis and treatment of breast cancer in developing countries. *World Journal of Clinical Oncology*, *5*(3), 465–477. https://doi.org/10.5306/wjco.v5.i3.465

Unger-Saldaña, K., Cedano Guadiamos, M., Burga Vega, A. M., Anderson, B. O., & Romanoff, A. (2020). Delays to diagnosis and barriers to care for breast cancer in Mexico and Peru: a cross sectional study. *The Lancet Global Health*, *8*, S16. https://doi.org/10.1016/s2214-109x(20)30157-1

Unger-Saldaña, K., Fitch-Picos, K., & Villarreal-Garza, C. (2019). Breast cancer diagnostic delays among young Mexican women are associated with a lack of suspicion by health care providers at first presentation. *Journal of Global Oncology*, *2019*(5), 1–12. https://doi.org/10.1200/JGO.19.00093

Unger-Saldaña, K., Miranda, A., Zarco-Espinosa, G., Mainero-Ratchelous, F., Bargallõ-Rocha, E., & Miguel Lázaro-Leõn, J. (2015). Health system delay and its effect on clinical stage of breast cancer: Multicentre study. *Cancer*, *121*(13), 2198–2206. https://doi.org/10.1002/cncr.29331

Unger-Saldaña, K., Ventosa-Santaulària, D., Miranda, A., & Verduzco-Bustos, G. (2018). Barriers and Explanatory Mechanisms of Delays in the Patient and Diagnosis Intervals of Care for Breast Cancer in Mexico. *The Oncologist*, *23*(4), 440–453. https://doi.org/10.1634/theoncologist.2017-0431

Utama, N. I., Sutrisnowati, R. A., Kamal, I. M., Bae, H., & Park, Y. J. (2020). Mining shift work operation from event logs. *Applied Sciences (Switzerland)*, *10*(20), 1–18. https://doi.org/10.3390/app10207202

van der Aalst, W. (2012). *Process mining: discovering and improving Spaghetti and Lasagna processes*. *c*, 1–7. https://doi.org/10.1109/cidm.2011.6129461

van der Aalst, W. M. P. (2011). Process Mining. *Process Mining*. https://doi.org/10.1007/978-3-642-19345-3

van der Aalst, W. M. P. (2018). Process mining and simulation: A match made in

heaven! *Simulation Series*, *50*(10), 39–50. https://doi.org/10.22360/summersim.2018.scsc.005

van der Aalst, W. M. P. (2021). Concurrency and Objects Matter! Disentangling the Fabric of Real Operational Processes to Create Digital Twins. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12819 LNCS*(August 2021), 3–17. https://doi.org/10.1007/978-3-030-85315-0_1

Van Der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, *16*(9), 1128–1142. https://doi.org/10.1109/TKDE.2004.47

van Hulzen, G., Martin, N., & Depaire, B. (2021). The Need for Interactive Data-Driven Process Simulation in Healthcare: A Case Study. *Lecture Notes in Business Information Processing*, *406 LNBIP*(Cm), 317–329. https://doi.org/10.1007/978-3-030-72693-5_24

van Hulzen, G., Martin, N., Depaire, B., & Souverijns, G. (2022). Supporting capacity management decisions in healthcare using data-driven process simulation. *Journal of Biomedical Informatics*, *129*(March), 104060. https://doi.org/10.1016/j.jbi.2022.104060

Weijters, A. J. M. M., van der Aalst, W. M. P., & Alves de Medeiros, A. K. (2006). Process mining with the Heuristics Miner algorithm (BETA Working Paper Series, WP 166). Eindhoven University of Technology.

Werner-Stark, A., Gerzson, M., & Hangos, K. M. (2011). Discrete event model structure identification using process mining. *Proceedings of the IASTED International Conference on Modelling, Identification and Control, April*, 228–233. https://doi.org/10.2316/P.2011.718-069

Wright, M. J., Bel'eed, K., Johnson, B. F., Eadington, D. W., Sellars, L., & Farr, M. J. (1999). Randomized prospective comparison of laparoscopic and open peritoneal dialysis catheter insertion. *Peritoneal Dialysis International*, *19*(4), 372–375. https://doi.org/10.1177/089686089901900415

Zhou, Z., Wang, Y., & Li, L. (2014). Process mining based modelling and analysis of workflows in clinical care - A case study in a chicago outpatient clinic. *Proceedings of the 11th IEEE International Conference on Networking, Sensing and Control, ICNSC 2014*, 590–595. https://doi.org/10.1109/ICNSC.2014.6819692

APPENDIX 1: SEARCH STRING

Database	Search String
PubMed	(("process* mining" [Title/Abstract] OR "workflow* mining" [Title/Abstract] OR " mining workflow*" [Title/Abstract] OR "process* discovery"[Title/Abstract] OR "event-log*"[Title/Abstract] OR "event log*"[Title/Abstract] OR "log-file"[Title/Abstract] OR "log file"[Title/Abstract] OR "conformance checking" [Title/Abstract]) AND ("discrete-event*"[Title/Abstract] OR " discrete event*"[Title/Abstract] OR "simulat*" OR "computer simulat*"))
Medline	TI= ((“process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*")) OR AB = (("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*")) OR AK=(("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*"))
Web of Science	TI= (("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*")) OR AB = (("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance

	checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*")) OR AK=(("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*"))
Scopus	TITLE-ABS-KEY (("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*" OR "computer simulat*"))
Google Scholar	(("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*"))
MATHSCINET	"Anywhere= ((("process* mining" OR "workflow* mining" OR " mining workflow*" OR "process* discovery" OR "event-log*" OR "event log*" OR "log-file" OR "log file" OR "conformance checking") AND ("discrete-event*" OR " discrete event*" OR "simulat*")))"
ACM Digital Library	[[All: "process* mining"] OR [All: "workflow* mining"] OR [All: " mining workflow*"] OR [All: "process* discovery"] OR [All: "event-log*"] OR [All: "event log*"] OR [All: "log-file"] OR [All: "log file"] OR [All: "conformance checking"]] AND [[All: "discrete-event*"] OR [All: " discrete event*"] OR [All: "simulat*"]]
IEEE Xplore	("All Metadata":("process* mining" OR "All Metadata":"workflow* mining" OR "All Metadata": " mining workflow*" OR "All Metadata": "process* discovery" OR "All Metadata": "event-log*" OR "All Metadata": "event log*" OR "All Metadata": "log-file" OR "All Metadata": "log file" OR "All Metadata": "conformance checking")) AND ("All Metadata":("discrete-event*" OR "All Metadata":"discrete event*" OR "All Metadata":"simulat*"))

JSTOR	Search Strategy 1: (("process* mining" OR "workflow* mining" OR "process* discovery" OR "conformance checking") AND ("discrete-event*" OR "discrete event*" OR "simulat*")) Search strategy 2: (("mining workflow*" OR "event-log*" OR "event log*" OR "log-file" OR "log file") AND ("discrete-event*" OR "discrete event*" OR "simulat*"))

APPENDIX 2: LIST OF VARIABLES IN THE MEXICAN BREAST CANCER CASE STUDY (REDACTED).

This table has been removed from the publicly available version of this thesis to maintain data confidentiality.

APPENDIX 3: SOURCE CODE FOR PROCESS MINING, INCLUDING DATA PREPARATION, CONCEPTUAL MODELLING, AND PARAMETERISATION FOR THE MEXICAN BREAST CANCER DATASET.

Data Preparation

```
library(haven)
library(readr)
library(dplyr)
library(tidyverse)

dtafile <- read_stata("dtafile.dta", encoding = "latin1")
#dtaupdate <- read_csv("UpdateFirst.csv")
dtaupdate <- read_csv("UpdateSecond.csv")

convert_date = function(date) {
  as.Date(date, format="%d/%m/%Y")
}

dtaupdate = dtaupdate %>%
  mutate_at(vars(-folio), 'convert_date')


# We will take rows from original file, which needs to be updated
newrows = dtafile %>%
  filter(folio %in% dtaupdate$folio) %>%
  arrange(folio)

# And we want to be sure, that they have same order as file with updates
dtaupdate = dtaupdate %>%
  arrange(folio)

# Now we iterate each column of df with updates and replace columns of selected
from original df rows
for (i in 2:ncol(dtaupdate)) {
  col = colnames(dtaupdate)[i]
  newrows[, col] = dtaupdate[, col]
}

# Then we delete tht rows from original df and add updated rows instead
dta_updated = dtafile %>%
  filter(!(folio %in% dtaupdate$folio)) %>%
```

```
 rbind(newrows)

 for (i in 1:nrow(dtafile)) {
  dta_updated[i, 'earliest_date'] =
min(unlist(c(dta_updated[i,"busq6_fecha_consulta1"],
                                dta_updated[i,"ss2_11_fecha_primera_consulta"],
                                dta_updated[i,"ss3_11_fecha_primera_consulta"])),
                           na.rm=T)
  dta_updated[i, 'latest_date'] =
max(unlist(c(dta_updated[i,"ss1_15_fecha_ultima_consulta"],
                               dta_updated[i,"ss2_13_fecha_ultima_consulta"],
                               dta_updated[i,"ss3_13_fecha_ultima_consulta"])),
                          na.rm=T)

  dta_updated[i, 'p3_4_fecha_inicio_tx_1_1_institu_']
=min(unlist(c(dta_updated[i,"p3_4_fecha_inicio_tx_1_1_institu"],
                                        dta_updated[i,"ss_tot_pre_36_cirugia_fecha"])),
                                   na.rm=T)
 }
dta_updated = dta_updated %>%
 mutate(earliest_date=as.Date(earliest_date,origin="1970-01-01"),
      latest_date=as.Date(latest_date,origin="1970-01-01"),
      p3_4_fecha_inicio_tx_1_1_institu =
as.Date(p3_4_fecha_inicio_tx_1_1_institu_,origin="1970-01-01"))

activities= c("Symptom discovery","Start consultation","End
consultation","Consultation in cancer", "Mammogram", "Ultrasound", "Biopsy",
"Biopsy report", "Treatment")
start_columns =
c("perc1_fecha_identif","earliest_date","latest_date","ss_tot_pre_40_consulta1onco_f
ech","ss_tot_pre_4_fecha_estudio1","ss_tot_pre_14_fecha_estudio2","ss_tot_pre_26_b
iopsia_fecha","p3_1_rhp_fecha_definitivo", "p3_4_fecha_inicio_tx_1_1_institu")
#age_columns =
c("sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad",
"sd1_edad","sd1_edad","sd1_edad")
#income_columns =
c("sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13
_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingres
o_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_perso
nal")

df = data.frame()
```

```
for (i in 1:length(activities)) {
  new_df = data.frame(ID=dtafile$folio, Activity="")
  for (id in dtafile$folio) {
    new_df[new_df$ID == id, "Activity"] = activities[i]
    new_df[new_df$ID == id, "Date"] = dta_updated[dta_updated$folio == id,
start_columns[i]]
    # new_df[new_df$ID == id, "Age"] = dtafile[dtafile$folio == id, age_columns[i]]
    #  new_df[new_df$ID == id, "Income"] = dtafile[dtafile$folio == id,
income_columns[i]]
  }
```

Conceptual modelling and parameterisation

```
library(haven)
library(readr)
library(dplyr)
library(tidyverse)

dtafile <- read_stata("dtafile.dta", encoding = "latin1")

for (i in 1:nrow(dtafile)) {
  dtafile[i, 'earliest_date'] = min(unlist(c(dtafile[i,"busq6_fecha_consulta1"],
                                  dtafile[i,"ss2_11_fecha_primera_consulta"],
                                  dtafile[i,"ss3_11_fecha_primera_consulta"])),
                           na.rm=T)
  dtafile[i, 'latest_date'] = max(unlist(c(dtafile[i,"ss1_15_fecha_ultima_consulta"],
                                  dtafile[i,"ss2_13_fecha_ultima_consulta"],
                                  dtafile[i,"ss3_13_fecha_ultima_consulta"])),
                                na.rm=T)

  dtafile[i, 'p3_4_fecha_inicio_tx_1_1_institu_']
=min(unlist(c(dtafile[i,"p3_4_fecha_inicio_tx_1_1_institu"],
                                  dtafile[i,"ss_tot_pre_36_cirugia_fecha"])),
                           na.rm=T)
}
dtafile = dtafile %>%
  mutate(earliest_date=as.Date(earliest_date,origin="1970-01-01"),
        latest_date=as.Date(latest_date,origin="1970-01-01"),
        p3_4_fecha_inicio_tx_1_1_institu =
as.Date(p3_4_fecha_inicio_tx_1_1_institu_,origin="1970-01-01"))
```

```
activities= c("Symptom discovery","Start consultation","End
consultation","Consultation in cancer", "Mammogram", "Ultrasound", "Biopsy",
"Biopsy report", "Treatment")
start_columns =
c("perc1_fecha_identif","earliest_date","latest_date","ss_tot_pre_40_consulta1onco_f
ech","ss_tot_pre_4_fecha_estudio1","ss_tot_pre_14_fecha_estudio2","ss_tot_pre_26_b
iopsia_fecha","p3_1_rhp_fecha_definitivo", "p3_4_fecha_inicio_tx_1_1_institu")
#age_columns =
c("sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad","sd1_edad",
"sd1_edad","sd1_edad","sd1_edad")
#income_columns =
c("sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13
_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingres
o_personal","sd13_ingreso_personal","sd13_ingreso_personal","sd13_ingreso_perso
nal")

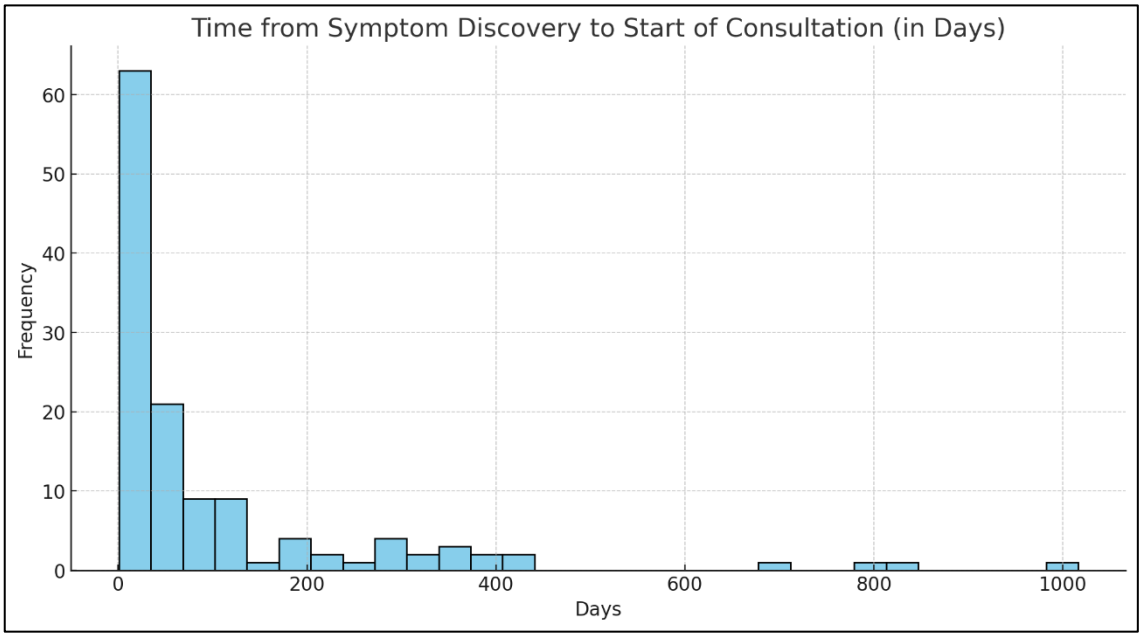
df = data.frame()

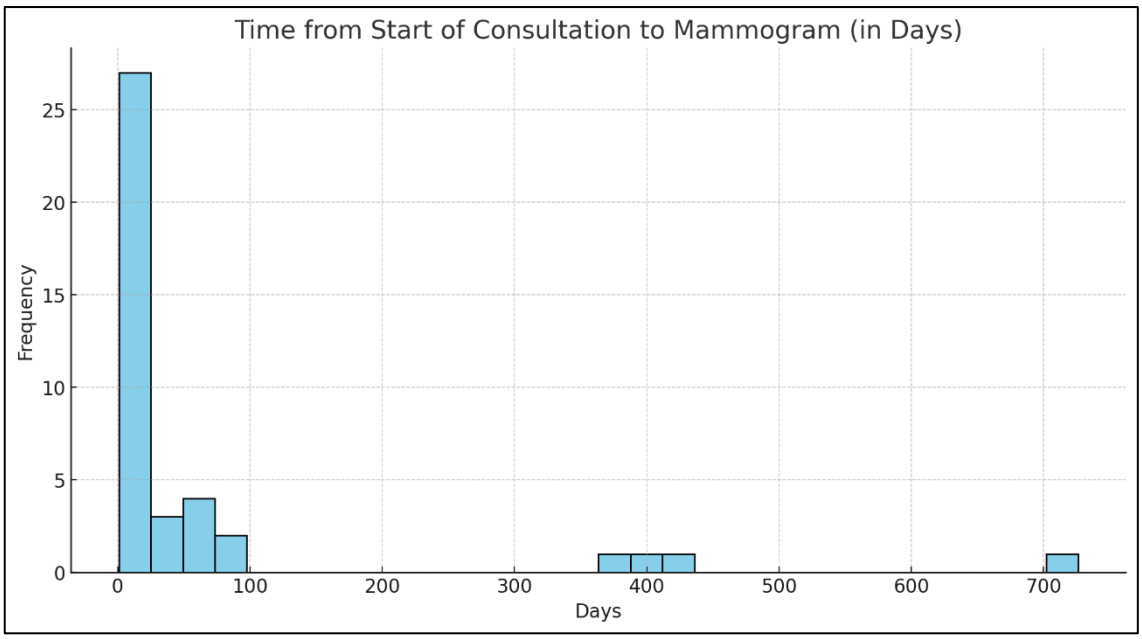
for (i in 1:length(activities)) {
  new_df = data.frame(ID=dtafile$folio, Activity="")
  for (id in dtafile$folio) {
    new_df[new_df$ID == id, "Activity"] = activities[i]
    new_df[new_df$ID == id, "Date"] = dtafile[dtafile$folio == id, start_columns[i]]
   # new_df[new_df$ID == id, "Age"] = dtafile[dtafile$folio == id, age_columns[i]]
  #  new_df[new_df$ID == id, "Income"] = dtafile[dtafile$folio == id,
income_columns[i]]
  }

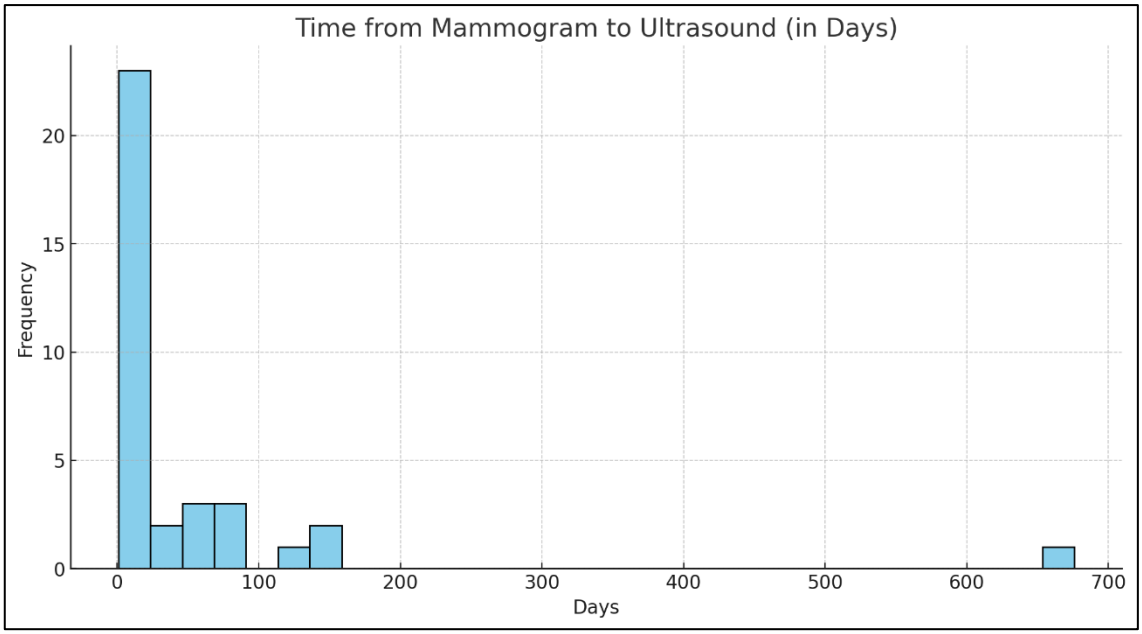
  df = rbind(df, new_df)
}
df = df %>% arrange(ID)

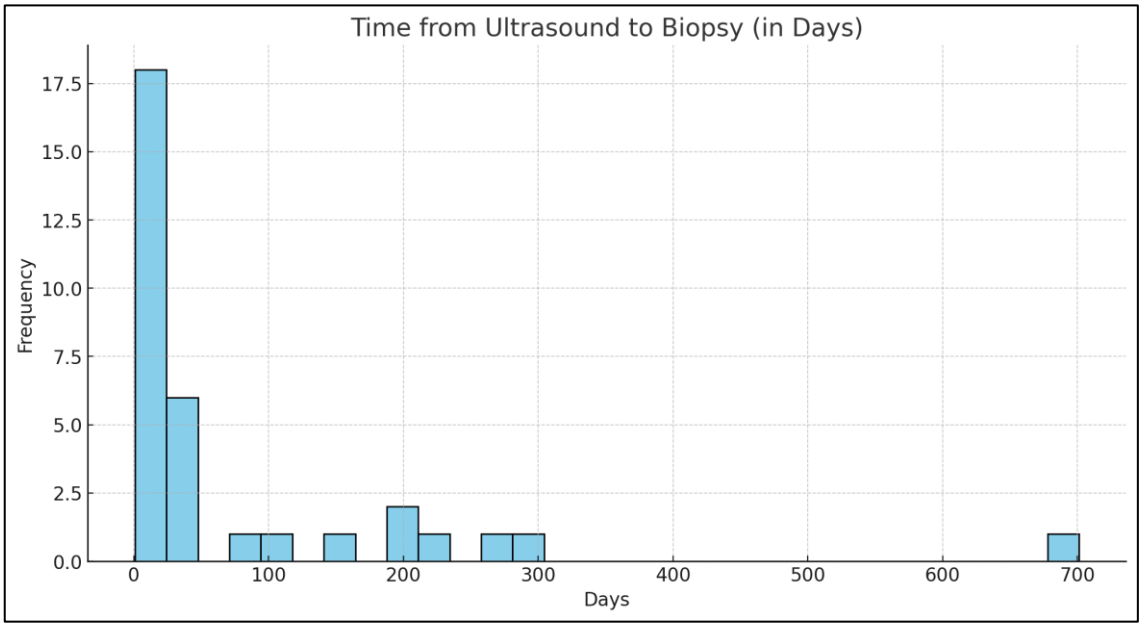
dfBC = df[(df$ID %in% unlist(dtafile[dtafile$Tipo_CA == 1, "folio"])),]
dfST_1 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 1, "folio"])),]
dfST_2 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 2, "folio"])),]
dfST_3 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 3, "folio"])),]
dfST_4 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 4, "folio"])),]
dfST_5 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 5, "folio"])),]
 dfST_6 = dfBC[(dfBC$ID %in% unlist(dtafile[dtafile$SS1_1_tipos2 == 6, "folio"])),]
```

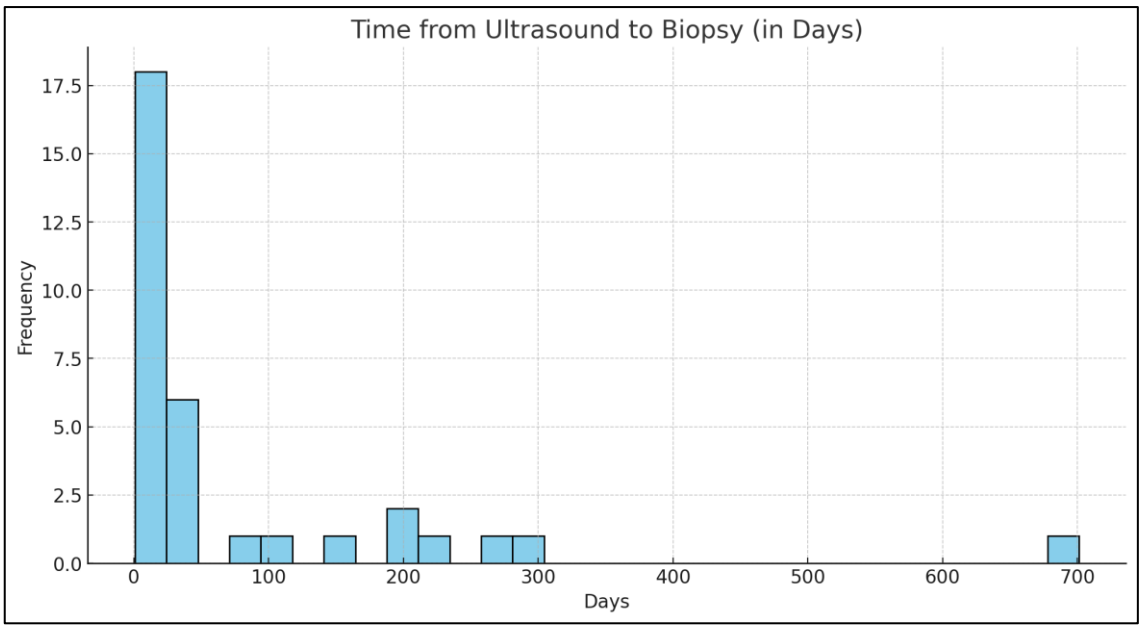
APPENDIX 4: EXAMPLE OF EMPIRICAL DISTRIBUTIONS IN THE MEXICAN BREAST CANCER CASE STUDY.

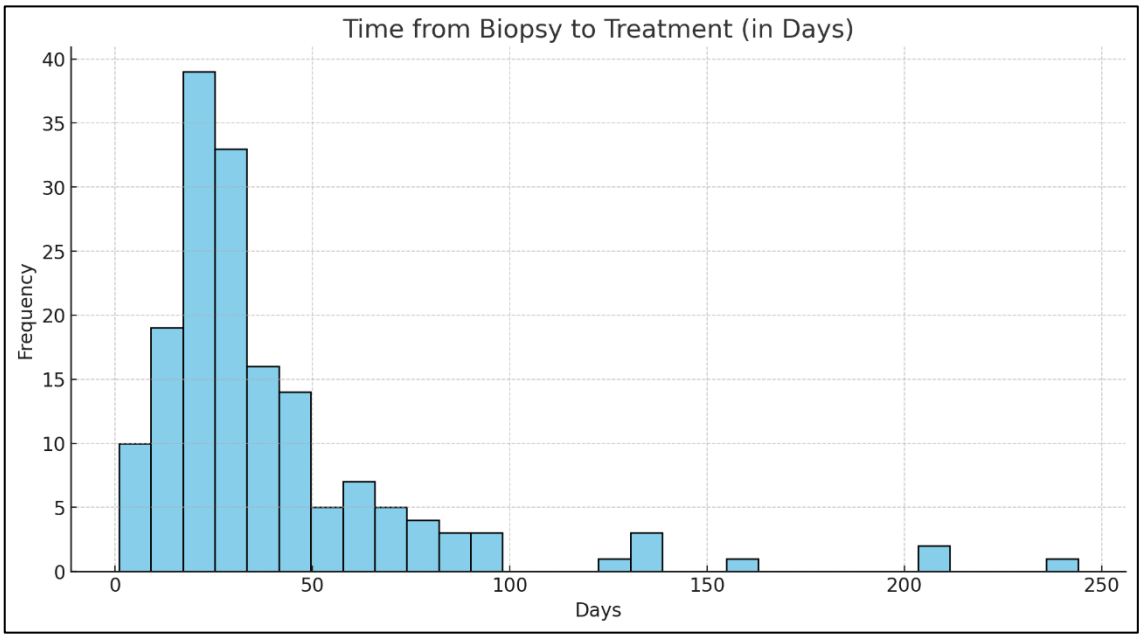





Time from Mammogram to Ultrasound (in Days)
Frequency
Days
0
5
10
15
20
0
100
200
300
400
500
600
700


Time from Ultrasound to Biopsy (in Days)
Frequency
Days
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
0
100
200
300
400
500
600
700


Time from Ultrasound to Biopsy (in Days)
Frequency
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0
100
200
300
400
500
600
700
Days


Time from Biopsy to Treatment (in Days)
Frequency
40
35
30
25
20
15
10
5
0
0
50
100
150
200
250
Days

APPENDIX 5: SOURCE CODE FOR PROCESS MINING, INCLUDING DATA PREPARATION, CONCEPTUAL MODELLING, AND PARAMETERISATION FOR THE UK KIDNEY DATASET.

Data Cleaning

```
      # converting these columns to the dd/mm/yyyy format
      CatheterInsertionDate = to_date(CatheterInsertionDate),
      FirstPDTrainingDate = to_date(FirstPDTrainingDate),
      FirstPDDialysisDate = to_date(FirstPDDialysisDate),
      LastPDDialysisDate = to_date(LastPDDialysisDate),
      DeathDate = to_date(DeathDate),
      ModalityStartDate = to_date(ModalityStartDate),
      ModalityStopDate = to_date(ModalityStopDate),
      qPresentDt_per = to_date(qPresentDt_per),
      qPresentDt_exit = to_date(qPresentDt_exit),
      CEtheter = to_date(CE),
      ExitSite = to_date(Exit),
      CE_Hosp = to_date(CE_Hosp),
      Hospalisation = to_date(Hosp),
      CE_Hosp = to_date(CE_Hosp),
      Peritonitis = to_date(Perit),
      G1_FirstDateTraining = str_replace(G1_FirstDateTraining, ' 00:00:00', ''),
      G1_FirstDateTraining = to_date(G1_FirstDateTraining, orders="%d/%m/%y")

 ) %>%
 mutate(FirstPDTrainingDate = pmin(G1_FirstDateTraining, FirstPDTrainingDate,
na.rm=T),
      LastPDDialysisDate = if_else(LastPDDialysisDate > DeathDate &
!is.na(DeathDate), DeathDate, LastPDDialysisDate),
      ModalityStopDate = if_else(ModalityStopDate < ModalityStartDate &
!is.na(ModalityStartDate), ModalityStartDate, ModalityStopDate))

# Replacing missing FirstPD
d$FirstPD_changed = F
FirstPDDialysisDates = d %>%
 filter(is.na(FirstPDDialysisDate)) %>%
 group_by(ID) %>%
 mutate(FirstPDDialysisDate_replace = min(ModalityStartDate, na.rm=T)) %>%
 select(ID, FirstPDDialysisDate_replace)
d[is.na(d$FirstPDDialysisDate), 'FirstPD_changed'] = T
```

```
d[is.na(d$FirstPDDialysisDate), 'FirstPDDialysisDate'] =
FirstPDDialysisDates$FirstPDDialysisDate_replace
#d[d$FirstPD_changed & d$FirstPDDialysisDate == '2000-01-01', 'HDStart'] =
d[d$FirstPD_changed & d$FirstPDDialysisDate == '2000-01-01',
'CatheterInsertionDate']
#d[d$FirstPD_changed & d$FirstPDDialysisDate == '2000-01-01',
'FirstPDDialysisDate'] = NA
problem_ID = d[d$FirstPD_changed & d$FirstPDDialysisDate == '2000-01-01', 'ID']

# Replacing missing LastPD
LastPDDialysisDates = d %>%
 filter(Modality == 1 & is.na(LastPDDialysisDate)) %>%
 group_by(ID) %>%
 mutate(ModalityStopDate_replace = max(ModalityStopDate, na.rm=T)) %>%
 select(ID, ModalityStopDate, ModalityStopDate_replace)
d[d$Modality == 1 & is.na(d$LastPDDialysisDate), 'LastPDDialysisDate'] =
LastPDDialysisDates$ModalityStopDate_replace
replace_all_modalities = d %>%
 filter(Modality == 1) %>%
 group_by(ID) %>%
 summarize(LastPDDialysisDate = max(LastPDDialysisDate)) %>%
 select(ID, LastPDDialysisDate) %>%
 rename(LastPD_replace = LastPDDialysisDate)
d = left_join(d, replace_all_modalities, by="ID")
d[is.na(d$LastPDDialysisDate), 'LastPDDialysisDate'] =
d[is.na(d$LastPDDialysisDate), 'LastPD_replace']
d$LastPD_replace = NULL
d[d$LastPDDialysisDate<0 & !is.na(d$LastPDDialysisDate), 'LastPDDialysisDate'] =
NA
```

Data Profiling and Data Preparation

```
library(tidyverse)
library(readxl)
library(lubridate)

census = read_excel('census.xlsx')
mod = read_excel('modalitynew.xlsx') %>%
  mutate(ModalityStartDate = as.Date(ModalityStartDate, '%m/%d/%y'),
         ModalityStopDate = as.Date(ModalityStopDate, '%m/%d/%y'))

census[census == '.'] = NA # replacing all values with only dot to NA
census = census %>% mutate(
  CatheterInsertionDate = as.Date(CatheterInsertionDate, '%m/%d/%y'),
  FirstPDTrainingDate = as.Date(FirstPDTrainingDate, '%m/%d/%y'),
  FirstPDDialysisDate = as.Date(FirstPDDialysisDate, '%m/%d/%y'),
  LastPDDialysisDate = as.Date(LastPDDialysisDate, '%m/%d/%y'),
  DeathDate = as.Date(DeathDate, '%m/%d/%y'),
)

duplicates = census[duplicated(census),]
nrow(duplicates)
write_csv(duplicates, 'duplicates.csv')

length(unique(census$PatientID))
colSums(is.na(census))

length(unique(census$FacilityID))
length(unique(census$StudyID))
length(unique(census$CatheterStatus))
length(unique(census$Diabetes))
length(unique(census$Gender))
length(unique(census$ReasonStarted))
length(unique(census$ReasonStopped))
length(unique(census$PatientAge))


# For checking frequency table for some column. specify the name of column in
both lines group_by
census %>%
  group_by(Diabetes, PatientID) %>%  # (first argument in the brackets)
  count() %>%
```

```
 group_by(Diabetes) %>%
 count() %>%
 arrange(desc(n))

df_step4 = data.frame(column = character(), N = numeric())
for(col in c('PatientAge', 'CatheterInsertionDate', 'FirstPDTrainingDate',
         'FirstPDDialysisDate', 'LastPDDialysisDate', 'DeathDate')) {
 vector = census[, col][[1]]
 new_row = data.frame(column = col,
                 N = length(unique(vector)))
 df_step4 = rbind(df_step4, new_row)
}

df_step4
summary(census[, c('PatientAge', 'CatheterInsertionDate', 'FirstPDTrainingDate',
              'FirstPDDialysisDate', 'LastPDDialysisDate', 'DeathDate')])
sd(census$PatientAge)
sd(census$CatheterInsertionDate)
sd(census$FirstPDTrainingDate)
sd(census$FirstPDDialysisDate)
sd(census$LastPDDialysisDate)
sd(census$DeathDate)

census %>%
 select('CatheterInsertionDate', 'FirstPDTrainingDate',
      'FirstPDDialysisDate', 'LastPDDialysisDate', 'DeathDate') %>%
 pivot_longer(cols = everything(),names_to = 'Column')  %>%
 ggplot() +
  geom_point(aes(Column, value, colour = Column), fill = 'orange', size=3) +
 labs(y = 'Year', x = 'Activity') +
 coord_flip() +
 theme(legend.position='none')

census %>%
 select('CatheterInsertionDate', 'FirstPDTrainingDate',
      'FirstPDDialysisDate', 'LastPDDialysisDate', 'DeathDate') %>%
 pivot_longer(cols = everything(),names_to = 'Column')  %>%
 ggplot() +
 geom_boxplot(aes(Column, value), fill = 'orange') +
 labs(y = 'Year', x = 'Activity') +
 coord_flip() +
 theme(legend.position='none')
```

```
census = census %>%
 mutate(range_1 = FirstPDTrainingDate - CatheterInsertionDate,
       range_2 = FirstPDDialysisDate - FirstPDTrainingDate,
       range_3 = LastPDDialysisDate - FirstPDDialysisDate,
       range_4 = DeathDate - LastPDDialysisDate)

census %>%
 select(PatientID,range_1, range_2,range_3,range_4) %>%
 write_csv('4_durations.csv')

mean(census$range_1, na.rm=T)
median(census$range_1, na.rm=T)
sd(census$range_1, na.rm=T)
mean(census$range_2, na.rm=T)
median(census$range_2, na.rm=T)
sd(census$range_2, na.rm=T)
mean(census$range_3, na.rm=T)
median(census$range_3, na.rm=T)
sd(census$range_3, na.rm=T)
mean(census$range_4, na.rm=T)
median(census$range_4, na.rm=T)
sd(census$range_4, na.rm=T)

negatives = census %>%
 filter(range_1 < 0 | range_2 < 0 | range_3 < 0 | range_4 < 0) %>%
 select(PatientID,range_1, range_2,range_3,range_4)

# export  negative table to csv
write.csv(negatives, 'negatives_of_duration_2.csv')

negatives %>% filter(range_1 < 0) %>% nrow() #number of negatives in range 1
negatives %>% filter(range_2 < 0) %>% nrow()
negatives %>% filter(range_3 < 0) %>% nrow()
negatives %>% filter(range_4 < 0) %>% nrow()

# Mod
mod[mod == '.'] = NA
length(unique(mod$ModalityStopDate                    ))
colSums(is.na(mod))

mod %>%
 group_by(PatientID, Modality) %>%
 count() %>%
```

```
 group_by(PatientID) %>%
 count(name='number_of_modalities') %>%
 write_csv('mod_number_of_modalities_for_each_ID.csv')

mod %>%
 group_by(PatientID) %>%
 count(name = 'times of modality')

mod %>%
 filter(Modality %in% c(1,3)) %>%
 group_by(PatientID) %>%
 count(name = 'times of modality') %>%
 write_csv('times of modality for_each_ID.csv')

mod %>%
 filter(ModalityStartDate > ModalityStopDate) %>%
 nrow()# number of occurencies

mod %>%
 filter(ModalityStartDate > ModalityStopDate) %>%
 write_csv('occurencies when start later than stop1.csv')

mod = mod %>%
 mutate(Duration = ModalityStopDate - ModalityStartDate)

mod %>%
 select(PatientID, Modality, Duration) %>%
 write_csv('modality durations.csv')

mean(mod$Duration, na.rm=T)
median(mod$Duration, na.rm=T)
sd(mod$Duration, na.rm=T)

mod %>%
 filter(Duration < 1) %>%
 nrow()

matches = mod %>%
 filter(Modality == 1) %>%
 group_by(PatientID) %>%
 summarise(earliest = min(ModalityStartDate)) %>%
 left_join(census, by='PatientID') %>%
 select(PatientID, earliest, FirstPDDialysisDate) %>%
```

```
  mutate(match = earliest == FirstPDDialysisDate)  %>%
  arrange(match)
write_csv(matches, 'matches.csv')

matches %>%
  count(match)

matches_late = mod %>%
  filter(Modality == 1) %>%
  group_by(PatientID) %>%
  summarise(latest = max(ModalityStopDate)) %>%
  left_join(census, by='PatientID') %>%
  select(PatientID, latest, LastPDDialysisDate) %>%
  mutate(match = latest == LastPDDialysisDate)  %>%
  arrange(match)
write_csv(matches_late, 'matches_latest1.csv')

matches_late %>%
  count(match)

mod = mod %>%
  arrange(PatientID) %>%
  mutate(ModStartStop_match = NA)

prevID = 0
for(i in 1:nrow(mod)) {
  currentID = mod$PatientID[i]
  if(currentID == prevID) {
    mod$ModStartStop_match[i] = mod[i-1, 'ModalityStopDate'] == mod[i,
'ModalityStartDate']
  }
  prevID = currentID
}

mod %>%
  count(ModStartStop_match)

mod %>%
  select(PatientID, Modality, ModalityStartDate, ModalityStopDate,
ModStartStop_match) %>%
  write_csv('Start and Stop dates do not match.csv')

mod %>%
```

```
  group_by(PatientID) %>%
  filter(ModalityStartDate == min(ModalityStartDate))%>%
  ungroup() %>%
  count(Modality, name='starts_with_it')

mod_0101_start = read_excel('modalitynew.xlsx') %>%
  filter(ModalityStartDate == '01/01/00')
mod_0101_stop = read_excel('modalitynew.xlsx') %>%
  filter(ModalityStopDate == '01/01/00')

nrow(mod_0101_start)
nrow(mod_0101_stop)
write_csv(mod_0101_start, 'startdates 01-01-00.csv')

mod %>%
  count(Modality)

length(unique(mod$PatientID[mod$Modality == 1]))
length(unique(mod$PatientID[mod$Modality == 2]))
length(unique(mod$PatientID[mod$Modality == 3]))
length(unique(mod$PatientID[mod$Modality == 4]))
length(unique(mod$PatientID[mod$Modality == 5]))
```

Conceptual modelling and parameterisation

```
library(tidyverse)
library(readxl)
library(lubridate)
library(magrittr)

to_date = function(x, orders = "%m/%d/%y"){    # this function will convert all
columns to date format
  x = parse_date_time(x, orders=orders)
}

# reading data
per = read_excel('peritonitis.xlsx')[, c('mergeID','qPresentDt')] %>% rename(ID =
mergeID, qPresentDt_per = qPresentDt)
exit = read_excel('exitSite.xlsx')[, c('mergeID','qPresentDt')] %>% rename(ID =
mergeID, qPresentDt_exit = qPresentDt)
hosp = read_excel('hospitalisation.xlsx')[, c('mergeID','hDiagProcCode',
'hAdmitDate')] %>% rename(ID = mergeID)
cns = read_excel('censusv1.xlsx')[, c('ID',
'CatheterInsertionDate','FirstPDTrainingDate','LastPDDialysisDate','DeathDate','Rea
sonStopped', 'FirstPDDialysisDate')]
mdl = read_excel('modalityv1.xlsx')[,
c('ID','ModalityStartDate','ModalityStopDate','Modality')]
ct = read_csv('CT.csv')[, c('ID','AR','AS','AT','AU','AV','AW')] %>%
  group_by(ID) %>%  # Merging rows with same ID in one row (if some column has
1 in at least one row, then final row will have 1 in the same columns)
  summarise(AR = max(AR),
            AS = max(AS),
            AT = max(AT),
            AU = max(AU),
            AW = max(AW),
            AV = max(AV)
  )
mq = read_csv('MQ_T.txt') %>%
  mutate(ID = P_StudyID) %>% filter(ID!= 1)
infections_raw = read_csv('infections.csv')
infections_raw$StateStartDate = dmy(infections_raw$StateStartDate)
infections = infections_raw %>%
  filter(!is.na(EventType)) %>%
  rename(ID = P_StudyID) %>%
  mutate(combined_event = case_when(EventType %in% c('CE') ~ 'CEtheter',
```

```
                        EventType %in% c('Exit') ~ 'ExitSite',
                        EventType %in% c('Hosp') ~ 'Hospalisation',
                        EventType %in% c('Perit') ~ 'Peritonitis')) %>%
  group_by(ID,combined_event) %>%
  summarise(Earliest_date = min(StateStartDate)) %>%pivot_wider(names_from =
"combined_event",values_from = "Earliest_date")


# we need hospital dataframe to have only one row for each ID, here we will find
all IDs we want to add

# joining data into 1 dataframe
d = left_join(cns, mdl, by='ID') %>%
  left_join(ct, by='ID') %>%
  left_join(mq, by='ID') %>%
  left_join(per, by='ID') %>%
  left_join(exit, by='ID') %>%
  left_join(hosp, by='ID') %>%
  left_join(infections, by='ID') %>%


d = d %>%
  mutate(TRDate = ReasonStopped == 23)
new_rows = d %>%
  filter(TRDate) %>%
  mutate(ModalityStartDate=LastPDDialysisDate,
         Modality = 'transplant')
new_rows = new_rows[!duplicated(new_rows$ID),]# if there are few modalities,
we still want to add only one row for transplant
new_rows$ModalityStartDate = new_rows$LastPDDialysisDate
new_rows$ModalityStopDate = NA
d = rbind(d, new_rows) %>%
  arrange(ID, Modality)

d = d %>%
  group_by(ID) %>%
  mutate(switch = case_when(LastPDDialysisDate >= FirstPDDialysisDate  &
                      !(FirstPDDialysisDate < ModalityStartDate &
                          ModalityStartDate < LastPDDialysisDate) ~ 1,
                    is.na(ModalityStartDate) | is.na(LastPDDialysisDate) ~ 1,
                    T ~ 2
  ),
  switch = max(switch),
```

```
  peritonitis = min(qPresentDt_per),
  exit_site = min(qPresentDt_exit),
  )

activities = c("CatheterInsertionDate","FirstPDTrainingDate",
"FirstPDDialysisDate",
            "LastPDDialysisDate", "DeathDate","Perit", "Exit", "Hosp", "CE") #
activities2 = c("PDStart", "HomeHDStart", "CentreHDStart", "HybridStart",
"NoDialysisStart", "TransplantStart",
             'PDStop','HHDStop',"CHDStop", "HybridStop", "NoDialysisStop")

df = data.frame() # it will be resulting dataframe
previous_ID = 0
ModalityStartDates = .POSIXct(character(0))
ModalityStopDates = .POSIXct(character(0))
modalities = c()
for(i in 1:nrow(d)){

  if(d$ID[i] != previous_ID & i>1) {

    if(sum(ModalityStartDates > FirstPD, na.rm=T) > 0) {
      ModalityStartDates_ = ModalityStartDates[ModalityStartDates > FirstPD] #Find
all dates with modality start date after FirstPD
      ModalityStopDates_ = ModalityStopDates[ModalityStartDates > FirstPD] #Find
Stopdates for the same rows
      modalities = modalities[ModalityStartDates > FirstPD] # Find modalities for the
same rows
      modality = modalities[which.min(ModalityStartDates_)] # Find modality of the
earliest of found StartDates
      ModalityStartDate = ModalityStartDates_[which.min(ModalityStartDates_)] #
Take the earliest of found Startdates
      ModalityStopDate = ModalityStopDates_[which.min(ModalityStopDates_)]
      if(length(ModalityStopDate) == 0) ModalityStopDate = NA #If we keep it as
empty list, StartDate will added twice instead

      if(modality == 1) activity = c('PD_Start', 'PDStop')  # We are adding two rows,
for start and stop dates
      if(modality == 2) activity = c('HomeHD_Start', 'HomeHDStop')
      if(modality == 3) activity = c('CentreHD_Start', 'CentreHDStop')
      if(modality == 4) activity = c('HybridPD+HD_Start', 'HybridStop')
      if(modality == 5) activity = c('NoDialysis_Start', 'NoDialysisStop')
```

```
    if(modality == 'transplant') activity = c('Transplant_Start', 'Transplate stop
date')


    new_df2 = data.frame(ID = rep(d$ID[i-1],2),
                         Activity=activity,
                         Date=c(ModalityStartDate, ModalityStopDate),
                         Group=rep(group,2))

    if(modality == 'transplant') new_df2 = new_df2[1,]
    df = rbind(df, new_df2)
   }


   ModalityStartDates = .POSIXct(character(0))
   ModalityStopDates = .POSIXct(character(0))
   modalities = c()
 }

 FirstPD = d$FirstPDDialysisDate[i]
 ModalityStartDates = c(ModalityStartDates, d$ModalityStartDate[i])
 ModalityStopDates = c(ModalityStopDates, d$ModalityStopDate[i])
 modalities = c(modalities, d$Modality[i])

 TRDate = ifelse (d$ReasonStopped[i] == 23, d$LastPDDialysisDate[i], NA)
 #ActivityPD = ifelse (d$Modality[i] %in% c(1), d$ModalityStartDate[i], NA)
 #ActivityHD = ifelse (d$Modality[i] %in% c(3), d$ModalityStartDate[i], NA)
 #ActivityPDSTop = ifelse (d$D[i] %in% c(1), d$F[i], NA)
 # ActivityHDSTop = ifelse (d$D[i] %in% c(3), d$F[i], NA)

 group=NA
 group_M = F
 group_S = F
 if (d$AT[i] == 1 | d$AU[i] == 1| d$AV[i] == 1 | d$AW[i] == 1) group_M = T
 if (d$AR[i] == 1 | d$AS[i] == 1) group_S = T
 if(group_S) group = 'S'
 if(group_M) group = 'M'
 if(group_S & group_M) group = 'SM'


 new_df = data.frame(ID = rep(d$ID[i],length(activities)),
                     Activity=activities,
                     Date=c(d$CatheterInsertionDate[i], d$FirstPDTrainingDate[i],
```

```
                    d$FirstPDDialysisDate[i],
                    d$LastPDDialysisDate[i],
                    d$DeathDate[i],
                    d$Perit_exit[i],

                    d$CE_Hosp[i]),
                Group=rep(group,length(activities)))

 if(d$ID[i] == previous_ID) {
  new_df = new_df %>%
   filter(!(Activity %in% activities))
 }
 df = rbind(df, new_df)
 previous_ID = d$ID[i]
}
new_rows <- data.frame()
unique_ids <- unique(df$ID)
for (id in unique_ids) {
 if (!any(df$ID == id & df$Activity == "DeathDate")) {
  new_row <- data.frame(
   ID = id,
   Activity = "EndOfStudy",
   Date = as.Date("2018-09-30"),
   Group = unique(df$Group[df$ID == id])
  )
  new_rows <- rbind(new_rows, new_row)
 }
}
df <- rbind(df, new_rows)
infections = infections %>% pivot_longer(cols = -ID,names_to =
"Perit_exit",values_to = "CE_Hosp")
colnames(infections) = c("ID","Activity","Date")

infections$Group = df$Group[match(infections$ID,df$ID)]
mr = df[df$ID %in% infections$ID,]
nr = infections[infections$ID %in% mr$ID,]
dfFinal = rbind(df,nr)
dfFinal = na.omit(dfFinal)

dfFinal$Group
=ifelse(is.na(dfFinal$Group),df$Group[match(dfFinal$ID,df$ID)],dfFinal$Group)
write.csv(dfFinal,"filelatest2_8_23.csv")
```

APPENDIX 6: TRAINING SUMMARY

List of participated modules

No.	Modules
1	FCM6100 Research Ethics and Integrity
2	HAR672 Advanced Simulation Methods
3	HAR6260 Economic Evaluation
4	Coursera: Process Mining: Data science in Action
5	Future Learn: Process Mining in Healthcare
6	Coursera: R Programming
7	ELTC6051 Academic Writing for PhD Candidates
8	ELTC6050 Thesis Writing-Principles and Practice
9	ELTC6060 Speaking Skills for Research Purposes
10	HAR6169 Study Design and Systematic Review Methods

List of Participated Conferences and Seminars

No.	Events
1	North East London NHS Foundation Trust's Research and Development Annual Conference 2020
2	Seminar series- Methods in Health Economics
3	ScHARR Conferences 2021 (Presented)
4	ScHARR Conferences 2022
5	ScHARR Conferences 2023
6	EURO Working Group on Operational Research Applied to Health Services (ORAHS 2021)
7	EURO Working Group on Operational Research Applied to Health Services (ORAHS 2022) in Bergamo, Italy (Presented)
8	The 9th KU Healthcare Operations Workshop 2023 at Koc University
9	The 9th International Symposium and 31st National Conference on Operations Research (HELORS 2023) at the University of West Attica (Presented)