MASTER

Agent-based planning at the CTS
analysis and simulation

van der Heijden, B.R.A.

Award date:
2007

Link to publication
Agent-Based Planning
at the CTS
Analysis and Simulation

By

ing. B.R.A van der Heijden

Supervisors:

prof. dr. ir. Han La Poutré (TU/e)
Anke Hutzschenerreuter M.Sc. (TU/e)
dr. Ton Weijters (TU/e)
Jan van Aarle (CZE)
drs. Ilona Blonk-Altena CIRM (CZE)

Eindhoven, April 2007
Abstract

Patient scheduling in hospitals is a complex task. Hospitals have a distributed organization and the treatment process is dynamic. In this thesis we focus on the planning of the Cardio-Thoracic Surgery patient processes from the Catharina Ziekenhuis Eindhoven. In our research, we denote two variants of the patient path, namely regular heart surgery and a short-track version of regular heart surgery. Furthermore, simulation is used to evaluate the patient planning in terms of throughput of patients. We do this from a distributed perspective, using a agent-based approach called ABOCSS. In ABOCSS, each postoperative care level is modeled as autonomous agents with their own goals. This reflects the decentralized structure in hospitals. The goal of the multi-agent system is to schedule patients through the surgery and postoperative care as quickly as possible. To reach this goal, the agents negotiate with each other. Finally, we examine the results of experiments with ABOCSS.
Acknowledgements

There are lots of people that I would like to thank for the support that they have given me in this final project at the TU/e. First of all, I thank my supervisors, Han and Anke, for the insightful conversations during the development of the ideas in this thesis, and for helpful and quick comments on the text. Besides my supervisors, I would like to thank my girlfriend Veerle, without whom the deadline for the final version would have been an impossible task, for the support during my master study!

I thank the Catharina Ziekenhuis Eindhoven for a place to work, and in particular Ilona, Jan, Hans, Frans and Ton for providing all the information that I needed to complete this research.

I would like to thank my friends for the cooperation and companionship in the BIS master.

I thank the parents of Veerle, who always had an encouragement ready.

Als laatste wil ik vooral mijn pa en ma bedankt voor de steun, de mogelijkheden en het vertrouwen dat ze mij al 25 jaar lang geven.
# Table of Contents

1. **Introduction** .......................................................... 1
   - 1.1 Motivation .......................................................... 1
   - 1.2 A short introduction to agent-based architectures .......... 3
     - 1.2.1 Why an agent-based simulation system? ............... 4
   - 1.3 Related work ...................................................... 4
   - 1.4 Overview ........................................................... 5

2. **Process Analysis** .................................................. 7
   - 2.1 Description of the ICU ........................................... 7
   - 2.2 Description of patient categories ............................ 8
   - 2.3 CTS Process description ........................................ 9
     - 2.3.1 ICU patient flow ........................................... 10
     - 2.3.2 Emergency patients ....................................... 10
     - 2.3.3 Characteristics and assumptions .......................... 11
   - 2.4 Daily IC planning ................................................. 11
   - 2.5 Conclusion ......................................................... 12

3. **Data Analysis** ...................................................... 13
   - 3.1 Methods ........................................................... 13
   - 3.2 Results of the data analysis .................................. 15
     - 3.2.1 General Findings .......................................... 15
     - 3.2.2 LoS of patients from different ICU specialisms ...... 16
     - 3.2.3 LoS of CTS patients at the ICU .......................... 16
   - 3.3 Variants of the patient path ................................... 19
     - 3.3.1 Statistical distributions of the LoS ........................ 21
   - 3.4 Mining results ................................................... 23
   - 3.5 Conclusion ........................................................ 23
# TABLE OF CONTENTS

## 4 Simulation system

4.1 Towards an agent-based simulation system

4.1.1 Methods

4.1.2 The name ABOCSS

4.2 Simulation system analysis

4.2.1 Step 1: Use Cases

4.2.2 Step 2: Initial Agent Types Identification

4.2.3 Step 3: Responsibilities Identification

4.2.4 Step 4: Acquaintances Identification

4.2.5 Step 5: Agent Refinement

4.2.6 Summary of the analysis phase

4.3 Simulation model design

4.3.1 Step 1: Agent splitting/merging/renaming

4.3.2 Step 2: Interaction Specification

4.3.3 Step 3: Creating Agent and Related Classes

4.3.4 Summary of the Design phase

4.4 Simulation system implementation

4.4.1 Agent Communication Architecture

4.4.2 Agent Communication Language

4.4.3 Implementation of algorithms

4.4.4 Summary of the Implementation phase

4.5 Conclusion

## 5 Experiments

5.1 Evaluation measures

5.2 Evaluation

5.2.1 Evaluation of ABOCSS with a varied patient planning

5.3 What if-scenarios

5.3.1 What if we changed the planning of resources

5.3.2 What if we changed the PACU opening times

5.3.3 What if the probabilities of patient path change

5.4 Conclusion

## 6 Conclusions and Recommendations

6.1 Conclusions

6.2 Recommendations and further work
# TABLE OF CONTENTS

**Appendices**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A Process Analysis</td>
<td>75</td>
</tr>
<tr>
<td>A.0.1 Legend</td>
<td>80</td>
</tr>
<tr>
<td>A.0.2 Description of the process models</td>
<td>80</td>
</tr>
<tr>
<td>B Data Analysis</td>
<td>83</td>
</tr>
<tr>
<td>B.0.3 The $\alpha$-mining algorithm</td>
<td>90</td>
</tr>
<tr>
<td>C Simulation Design</td>
<td>91</td>
</tr>
<tr>
<td>D User Manual</td>
<td>93</td>
</tr>
<tr>
<td>D.1 Preface</td>
<td>94</td>
</tr>
<tr>
<td>D.2 Prerequisites</td>
<td>94</td>
</tr>
<tr>
<td>D.3 The Menu</td>
<td>94</td>
</tr>
<tr>
<td>D.3.1 Setup</td>
<td>96</td>
</tr>
<tr>
<td>D.3.2 Resources</td>
<td>97</td>
</tr>
<tr>
<td>D.3.3 Simulation</td>
<td>98</td>
</tr>
<tr>
<td>D.3.4 Waiting List</td>
<td>99</td>
</tr>
<tr>
<td>D.3.5 Running simulations</td>
<td>100</td>
</tr>
<tr>
<td>D.3.6 Results</td>
<td>102</td>
</tr>
<tr>
<td>D.3.7 Processing results</td>
<td>102</td>
</tr>
</tbody>
</table>
List of Figures

4.1 Use Case Definition of ABOCSS .................................................. 30
4.2 Initial agent types definition of ABOCSS .................................... 31
4.3 Agent acquaintances definition of ABOCSS ................................. 33
4.4 Agent types definition of ABOCSS ............................................. 35
4.5 Class diagram of ABOCSS .......................................................... 46
4.6 Generalization relation of the supertype agent .............................. 47
4.7 Generalization relation of the supertype bed ................................ 47
4.8 Generalization relation of the supertype patient ............................ 48

5.1 Default settings of ABOCSS ....................................................... 51

A.1 The main process flow of CTS patients ...................................... 76
A.2 The HC sub process ................................................................. 77
A.3 The IC planning process ........................................................... 78
A.4 The IC emergency process ........................................................ 79

B.1 OR duration in minutes per specialism [1] ................................... 87
B.2 Scatterplot of the relation between the LoS at the ICU and the OR duration of 2005 data [1] ................................................................. 88

C.1 The negotiation model of ABOCSS ................................................ 92

D.1 ABOCSS Simulation Runner main screen ..................................... 95
D.2 ABOCSS Simulation Runner Simulation Setup Screen .................. 96
D.3 ABOCSS Simulation Runner Resources Setup Screen .................. 98
D.4 ABOCSS Simulation Runner Simulation Screen ............................ 99
D.5 ABOCSS Simulation Runner Waiting List Setup Screen ................. 100
D.6 ABOCSS Simulation Runner running a simulation ....................... 101
D.7 ABOCSS Simulation Runner results screen ............... 103
List of Tables

2.1 Description of the different care levels that CTS patients can be admitted to. .............................................. 8

3.1 LoS at the ICU in days for the different specialisms ................................................................. 17

3.2 LoS at the ICU in days for the different patient types ................................................................. 18

3.3 Parameters for the Gamma distribution for the HC variant ......................................................... 22

3.4 Parameters for the Gamma distribution for the PACU variant ..................................................... 22

3.5 Parameters for the Lognormal distribution for CTS patients ....................................................... 22

4.1 Responsibility table ......................................................................................................................... 31

5.1 Initial run results .............................................................................................................................. 52

5.2 Completion dates of the initial evaluation ...................................................................................... 53

5.3 Simulation runs with an inflow of 1900 patients ........................................................................... 53

5.4 Simulation runs with an inflow of 1600 patients ........................................................................... 53

5.5 Simulation runs with an inflow of 2000 patients ........................................................................... 53

5.6 Completion dates of scenario 1 ........................................................................................................ 55

5.7 Scenario 1a: simulation runs with 2 dedicated IC beds for CTS patients ........................................ 55

5.8 Scenario 1b: simulation runs with 3 dedicated IC beds for CTS patients ........................................ 55

5.9 Scenario 1c: simulation runs with 4 dedicated IC beds for CTS patients ........................................ 55

5.10 Completion dates of scenario 2 ....................................................................................................... 57

5.11 Scenario 2a: simulation runs with a prolonged opening hour of the PACU ................................. 57

5.12 Scenario 2b: simulation runs with a PACU opening time until 4:00 hours ................................. 57

5.13 Scenario 2c: simulation runs with a PACU opening duration of 24 hours .................................. 58

5.14 Completion dates of scenario 3 ....................................................................................................... 60

5.15 Scenario 3a: simulation runs with a 50% PACU patients ............................................................ 60

5.16 Scenario 3b: simulation runs with a 30% PACU patients ............................................................ 60

5.17 Scenario 3c: simulation runs with a change in the path of PACU patients: IC 20%, MC 10% and Surgical Ward 70% .......................................................... 60
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.18</td>
<td>Scenario 3d: simulation runs with a change in the path of HC patients: IC 10%, MC 5% and Surgical Ward 85%</td>
<td>60</td>
</tr>
<tr>
<td>5.19</td>
<td>Scenario 3e: simulation runs with 30% PACU patients and 3 OR’s and IC’s for PACU patients and 5 OR’s and IC’s for HC patients</td>
<td>61</td>
</tr>
<tr>
<td>5.20</td>
<td>Scenario 3f: simulation runs with 75% PACU patients and 5 OR’s and IC’s for PACU patients and 3 OR’s and IC’s for HC patients</td>
<td>61</td>
</tr>
<tr>
<td>B.1</td>
<td>Surgical Interventions Legend with number of patients per category (2005) [1]</td>
<td>84</td>
</tr>
<tr>
<td>B.2</td>
<td>Number of patients per category for the OR (1998 - 2005) [3]</td>
<td>84</td>
</tr>
<tr>
<td>B.3</td>
<td>Number of patients per category for the IC (2002 - 2006) [4]</td>
<td>85</td>
</tr>
<tr>
<td>B.5</td>
<td>LoS in days of PACU patients staying at the IC [4]</td>
<td>86</td>
</tr>
<tr>
<td>B.6</td>
<td>LoS in days of PACU patients staying at the MC [4]</td>
<td>86</td>
</tr>
<tr>
<td>B.7</td>
<td>LoS in days of HC patients staying at the IC [4]</td>
<td>86</td>
</tr>
<tr>
<td>B.8</td>
<td>LoS in days of HC patients staying at the MC [4]</td>
<td>86</td>
</tr>
</tbody>
</table>
### List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ABOCS</strong></td>
<td>Agent Based Or and Care Simulation System</td>
</tr>
<tr>
<td><strong>ACL</strong></td>
<td>Agent Communication Language</td>
</tr>
<tr>
<td><strong>ASCII</strong></td>
<td>American Standard Code for Information Interchange</td>
</tr>
<tr>
<td><strong>CTS/CTC</strong></td>
<td>Cardio-Thoracic Surgery</td>
</tr>
<tr>
<td><strong>CZE</strong></td>
<td>Catharina Ziekenhuis Eindhoven</td>
</tr>
<tr>
<td><strong>CCU</strong></td>
<td>Coronary Care Unit</td>
</tr>
<tr>
<td><strong>EZIS</strong></td>
<td>Electronic Care Information System</td>
</tr>
<tr>
<td><strong>IC</strong></td>
<td>Intensive Care</td>
</tr>
<tr>
<td><strong>ICIS</strong></td>
<td>Intensive Care Information System</td>
</tr>
<tr>
<td><strong>ICU</strong></td>
<td>Intensive Care Unit</td>
</tr>
<tr>
<td><strong>JADE</strong></td>
<td>Java Agent Development Environment</td>
</tr>
<tr>
<td><strong>KQML</strong></td>
<td>Knowledge Query and Manipulation Language</td>
</tr>
<tr>
<td><strong>KQML</strong></td>
<td>Knowledge Query and Manipulation Language</td>
</tr>
<tr>
<td><strong>LoS</strong></td>
<td>Length of Stay</td>
</tr>
<tr>
<td><strong>MC</strong></td>
<td>Medium Care</td>
</tr>
<tr>
<td><strong>MAS</strong></td>
<td>Multi-agent System</td>
</tr>
<tr>
<td><strong>MDM/MDO</strong></td>
<td>Multi Disciplinary Meeting</td>
</tr>
<tr>
<td><strong>MOM</strong></td>
<td>Message Oriented Middle-ware</td>
</tr>
<tr>
<td><strong>OR</strong></td>
<td>Operating Room/Surgery/Operation Theater</td>
</tr>
<tr>
<td><strong>PACU</strong></td>
<td>Post Anaesthesia Care Unit</td>
</tr>
<tr>
<td><strong>PMCC</strong></td>
<td>Pearson product-moment correlation coefficient</td>
</tr>
<tr>
<td><strong>RPC</strong></td>
<td>Remote Procedure Calls</td>
</tr>
<tr>
<td><strong>UC</strong></td>
<td>Use Case</td>
</tr>
<tr>
<td><strong>UML</strong></td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td><strong>WF nets</strong></td>
<td>Workflow nets</td>
</tr>
<tr>
<td><strong>XML</strong></td>
<td>eXtensible Markup Language</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

This master thesis is about the agent-based analysis and simulation at the Cardio-Thoracic Surgery (CTS) department, one of the departments of the Catharina Ziekenhuis Eindhoven (CZE). At this department, heart and lung surgeries like a bypass or valve surgery are performed. This research is focused on two main questions:

• Can an agent-based (decentralized) planning system contribute to the patient planning process at the CTS department of the CZE?

• Can an agent-based simulation system for the planning of the CTS patient processes be built, to evaluate the patient planning in terms of throughput of patients?

1.1 Motivation

The health care system in the Netherlands is changing. Today, hospitals cannot rely on the fact that whenever the budget is short, the gouvernement will settle the difference. Instead hospitals have to negotiate with health care insurances over a fixed price per surgery and the number of surgeries that the surgeons are going to perform. This negotiation is based on the number of patients that the hospital is going to admit per year. The hospital has the need to plan carefully, because the limited availability of beds and the fact that the agreed
number of patients is set high to ensure that sufficient patients are planned. The fixed price per surgery is an average price for which the hospital management thinks that it can deliver the surgery and postoperative care. This is difficult to estimate, because the surgery and postoperative care are never the same for different patients. The focus in this thesis will be on the planning of postoperative care of CTS patients in the CZE from a decentralized planning perspective.

The CZE distinguishes two types of patient categories for CTS patients at the Intensive Care Unit (ICU). The ICU is a department at the CZE where postoperative care is performed. The first category is regular heart surgery patients and the second category is a short-track version of regular heart surgery patients. The condition of the latter patient category is less critical than the condition of regular heart surgery patients. Therefore, the latter patient category is estimated to require less days in postoperative care than regular heart surgery patients.

The planning of surgery and postoperative care is denoted by several problems:

1. Each year, the CZE defines the number of surgeries that each surgical specialism is going to perform based on the negotiation with insurance companies. When the demand for surgery increases, like in 2004 when a hospital in the neighborhood had to close and the surgeries were transferred to the CZE, the actual number of surgeries performed exceeds the agreed number. When the resource capacity remains the same and the number of patients increases, the waiting list will grow and the waiting time for new patients will increase.

2. The second problem occurs if a patient has to stay longer than expected in postoperative care and an adjustment in planning and schedules has to be made. Related to this problem is the uncertainty in the patient path. With uncertainty in the patient

---

1 Resources are beds, surgical theaters, medical specialists, medical staff etcetera.
1.2 A short introduction to agent-based architectures

This thesis is about the planning of CTS patients from a decentralized perspective using software agents. To introduce agent-based architectures, a short description will be provided in this section.

An agent is a software module that is situated in some environment capable of performing autonomous actions in that environment in order to meet its design objective \[19\]. The design objective is commonly called a goal. The goal of the agent differs per type of agent, but the goal of all agents is to try to keep its utility as high as possible. Utility is the degree of satisfaction that is gained by performing actions. Each action corresponds with a raise or reduction of the agent’s utility.

A multi-agent system (MAS) is a system composed of a number of agents, which typically interact with one-another in order to satisfy their goals \[20\]. Coordination is needed either to achieve a common objective or because this helps each of the interacting agents to achieve their own objectives \[5\]. In our research, the agents are used for Cooperative Distributed Problem Solving (CDPS) \[6\]. CPDS concerns problems that are beyond the individual capabilities of one agent or one central system. To solve such a problem, the agents have to
work together by coordinating with each other and use the speciality of each agent to come to a solution.

1.2.1 Why an agent-based simulation system?

The reason that a MAS was used for the implementation of an agent-based simulation system for the planning of the CTS patient processes, is that the CZE is characterized by decentralization. Decisions at the different departments are made autonomously and each department has its own patient schedule. Furthermore, complications and arrivals of emergency patients, which are in urgent need of treatment, result in schedule disturbance. Due to this decentralization, patient scheduling in hospitals requires a distributed approach in order to leave the authority at the responsible hospital units. Hospital scheduling should also be flexible, to be able to react to changes of the planning in an efficient manner [14]. To coordinate the scheduling of postoperative care between the different patient groups under the inherent uncertainty, a MAS is used.

1.3 Related work

This research is about the analysis, design and evaluation of agent-based scheduling in hospitals. Earlier work in this area has been reported in [2], [9], [10], [11], [12], [14] and [15]. Below, a discussion is given about the similarities and differences between the literature and our research.

The paper of Moreno et al. [10] is about a multi-agent system that allows the user to access their medical record, to find out information about medical centers in the city and to make a booking to be visited by a particular kind of doctor. This paper discusses area’s as security and the design of a basic medical ontology and not actual scheduling issues in hospitals. The paper of Nealon and Moreno [11] [12] gives more of an overview on agent-based health care systems. They introduce some main issues related to the deployment of agent-based systems
in health care and argue why agent-based systems are a good choice to tackle problems in the health care domain. They only mention patient scheduling as a field of application within health care. In their research, issues like security, legal issues and social acceptance are discussed. In the paper of Decker and Li [2], hospital patient scheduling is discussed from a distributed problem solving point of view. Agents in the paper of Decker and Li schedule hospital test for patients by bidding on a time interval that the agent needs and a priority of the agent’s task. The agent that has won the time interval keeps its schedule and all the other agents mark the interval as taken, reschedule and bid on another time slot.

The paper of Paulussen et al. [14, 15] does not deviate much from our research. The difference they make is to represent patients as agents that negotiate with resource agents about time slots. The patient agents evaluate their current schedule and calculate demand and supply prices for time slots on the concept of years of well being. This concept describes a cost function in which the disease of the patient is considered as a decrease in quality of life. Both the papers of Decker and Li, and Paulussen focus on scheduling every step and the needed resources in the patient process, while we focus on the planning of the surgery and postoperative care of CTS patients taking other surgical patient groups and emergency patients into account. In the paper of Marinagi et al. [9] the emphasis is on scheduling of patient tests in hospital laboratories. They use an approach in which a test can be seen as a task that can further be decomposed into smaller tasks. This corresponds to the blackboard approach. For the interesting reader, a description of a blackboard approach can be found in [1].

1.4 Overview

Chapter 2 contains the analysis of current patient processes. In this chapter, the first process models that were derived from interviews with medical specialists and hospital planners are presented. The analysis of patient data is described in Chapter 3 and contains a statistical
analysis to provide insight in the patient path and Length of Stay (LoS) of patients at
different wards. The LoS concerns the time that a patient stays at a ward.

The design and implementation of the agent-based simulation system for the planning of
the CTS patient processes is described in Chapter 4. It shows all the steps that were needed
to come to an implementation of the agent-based simulation system. Chapter 5 contains a
description of the scenarios and the outcome of the experiments. Here, also a conclusion is
stated with respect to the scenarios. The main conclusions can be found in Chapter 6.
Chapter 2

Process Analysis

In this chapter we present a model for the CTS patient path based on interviews with medical specialists and hospital planners. In the first part of this chapter, the process analysis of the patient path is discussed. The models can be seen in Appendix A and are founded on a modeling technique from Workflow Management as described in [16].

In the process description we distinguish two patient categories. These two patient categories each follow a variant of the patient path. In the next chapter, we provide a data analysis to give more insight in the patient path. The remainder of this chapter contains a description of the daily ICU planning and a conclusion.

2.1 Description of the ICU

Post operative care in the CZE is provided at the ICU. The ICU is grouped in different care levels. The first one is Intensive Care (IC), the highest level of care. IC is intended for patients that have two or three failing organs. IC can either be for acute patients or for surgical patients in postoperative care. The lowest level of care at the ICU is Medium Care (MC). MC is intended for patients whose condition is too bad for the surgical ward and too good for the IC. Patients with one organ failure can stay at the MC. For heart
Table 2.1: Description of the different care levels that CTS patients can be admitted to.

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Short description</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>The HC ward is intended for regular heart surgery patients. At this ward, a regular heart surgery patient can stay for at most 24 hours.</td>
<td>4 beds</td>
</tr>
<tr>
<td>PACU</td>
<td>The PACU ward is intended for heart surgery patients whose condition is better than regular heart surgery patients. Patients admitted to this ward can stay here for at most 10 hours.</td>
<td>4 beds</td>
</tr>
<tr>
<td>IC</td>
<td>IC is the highest level of care at the ICU. Patients admitted to this ward have multiple organ failures.</td>
<td>11 beds</td>
</tr>
<tr>
<td>MC</td>
<td>MC is intended for patients whose condition is too good for IC and too bad for the surgical ward. Patients that need MC have one organ failure.</td>
<td>4 beds</td>
</tr>
<tr>
<td>Surgical ward</td>
<td>A patient that is prepared for surgery is first admitted to the surgical ward. After surgery and postoperative care, the patient is again returned to the surgical ward to heal in prospect of the patients return home.</td>
<td>35 beds</td>
</tr>
</tbody>
</table>

surgery patients, two levels of postoperative care are available, High Care (HC) and Post Anaesthesia Care Unit (PACU). At the HC, postoperative care for regular heart surgery patients is provided. Post operative care for the short-track version of regular heart surgery patients is provided at the PACU. The difference between these two departments are the opening times. At the HC, the patient can stay and heal for the night, while PACU patients have to leave the PACU before the night. A summary of the care levels that CTS patients can be admitted to, can be referenced in Table 2.1 on page 8.

2.2 Description of patient categories

As stated in the introduction of this thesis, the CTS department uses two categories of surgical patients. One category is regular heart surgery patients and the other category is a short-track version of regular heart surgery patients. In the remainder of this thesis, we will denote a regular heart surgery patient as a HC patient and the short-track version of a
2.3 CTS Process description

When a patient suffers from some kind of heart disease, the patient’s family doctor or cardiologist will refer the patient to a hospital. When the hospital receives the referral and when there is no emergency treatment necessary, the planning team of the hospital adds the patient to the waiting list, otherwise the patient is brought to the hospital with an ambulance and treated immediately. In the waiting list case, the patient visits the hospital several weeks (usually 2 or 3) before the patient is scheduled for surgery. During this visit, the patient’s condition is evaluated according to several medical measures.

The in-patient process starts when a patient is admitted to the hospital. The first department that the patient visits is the surgical ward. This ward consists of 35 beds; all of which are normal care beds. Patients will stay here for at least one day and one night. At this ward, the patient is prepared for surgery and a nurse makes sure that the condition of the patient is not changed with respect to the initial evaluation. When the condition has changed, the patient must undergo several new tests and examinations.

At 8:00 the first four patients for that day are prepared for surgery. For the CTS specialization, four operating theaters are reserved. At best, surgeons start with four PACU patients. The PACU has four beds reserved every morning (actually, these four beds need to be empty at 22:00 the previous evening). Because of this, a PACU patient that is in surgery can be admitted to the PACU without further delay or scheduling. PACU patients
admitted to the PACU are considered to go to the surgical ward within 10 hours of their admission. When a surgeon is finished operating a PACU patient, the surgeon will proceed with a HC patient. After surgery, a HC patient is admitted to the HC. At the HC, four beds are reserved for HC patients.

2.3.1 ICU patient flow

A patient is admitted to the ICU when this patient’s condition is getting worse. In the process diagrams in the appendices (denoted in Figure A.1) this is shown by the dollar ($) path. First, the patient’s condition is evaluated. Then a decision is made whether the patient should get IC or MC. When a patient is scheduled in the IC subprocess, the decision has to be made whether a patient should be admitted to an IC bed or get surgery redone.

2.3.2 Emergency patients

At all times in the process, the hospital is open for emergency patients. Emergency patients are difficult to plan, because they arrive unplanned. The process of admitting an emergency patient (denoted in Figure A.4) starts when the hospital gets a request from an ambulance transporting an emergency patient. The ICU planner first checks whether there is a bed available. If a bed is available, the emergency patient can be admitted directly. If there is no bed available, the ICU planner meets with the medical staff to check if there is a patient ready to be transferred to another ward. If there is a patient that can be transferred, a bed becomes available and the emergency patient can be admitted. If there is no patient ready, the emergency patient will either be admitted to the Coronary Care Unit (CCU) in the hospital or cannot be admitted at all. The CCU is a hospital department which is specialized in the care of patients with various cardiac conditions (for example heart attacks) that require continuous monitoring and treatment. When the hospital has no bed available, the ambulance will try another hospital in the neighbourhood.
2.3.3 Characteristics and assumptions

In the process description stated above, several assumptions are made with respect to the planning of patients. These assumptions have to be taken into account during the implementation and simulation of the process.

- When a problem situation is encountered at the ICU, it is up to the creativity of the ICU planner to come up with a solution. When, for example, all beds are occupied at the IC and a patient needs to be admitted there, the ICU planner can admit the patient to the CCU.

- Patients can be admitted to a higher care ward when there is no bed available at the intended ward (i.e. a patient scheduled for MC can be admitted to the HC ward or the IC ward) but never the other way around (i.e. a patient for the IC ward can never be admitted to the MC ward).

2.4 Daily IC planning

At 07:30, the ICU planner, ICU doctors and ICU nurses meet for a Multi-Disciplinary Meeting (MDM). Here, all patients staying at the ICU are evaluated. The patients’ progress is marked and some patients are considered for admission to other care levels. This is the first step in the ICU process (Figure A.3 on page 78). After this meeting, the ICU planner can make a list of all patients with their destination. This destination can either be: 1) stay at the current ward, 2) move to a lower care ward or 3) move to a higher care ward. With this information and the planning of the Operating Room (OR), the ICU planner composes an ICU planning. At the time of composition, there are already four operating theaters in use with the preparation of PACU patients.¹

Until 13:00, the time that an afternoon MDM is scheduled, more or less beds at the ICU

¹And of course with other patients who perhaps need IC, but these types of patients are not considered in the initial model; in later models, other patient flows will be described.
can be required, compared to the planning. In most of the cases this is due to the fact that a bed is required for an emergency patient. Sometimes a surgery is canceled and a planned bed becomes available. In the afternoon MDM, patients’ conditions are evaluated. From a planning perspective, this meeting is less interesting, because no more patients are discharged. Between 16:00 and 16:30 the ICU doctors and ICU planners visit all four PACU patients. One of the following decisions can be made according to the patient’s condition: 1) the patient is transferred to the surgical ward, 2) the patient needs more time to heal and can stay at the PACU until 21.30 or 3) the patient needs to stay for the night and is transferred to the IC or MC or, when there is a bed available at the HC (i.e. there are only three beds occupied), the patient can stay in this bed.

2.5 Conclusion

In this chapter, a process model was discussed to describe the general CTS patient path. Two categories of patients follow this path, namely regular heart surgery patients and a short-track version of the regular heart surgery patients. The regular heart surgery patients are denoted by HC and the short-track version of the regular heart surgery patients by PACU. In the next chapter, a data analysis providing insight in the different aspects of the patient paths is described.
Chapter 3

Data Analysis

In this chapter we present the data analysis. The data analysis has been conducted for two purposes: 1) to obtain insight into the patient path and 2) to find the underlying probability distributions of the Length of Stay (LoS), in such a way that the probability distributions can be implemented in the simulation system for the planning of the CTS patient processes.

This chapter is organized as follows: First the methods are provided that were used to collect and analyze the data, followed by the results of the analysis. Finally a conclusion on the results is presented.

3.1 Methods

In the first stage of the data analysis, several interviews were conducted to get an overview of the information that was stored in the different information systems of the CZE (like patient records, durations of surgeries etcetera). In the second stage, data from the Electronic Care Information System (EZIS), the Intensive Care Information System (ICIS), the Cardio2000 system and the admission office was collected and analyzed. EZIS is a hospital-wide information system, ICIS is only used at the ICU, Cario2000 is used by the CTS specialism
to collect surgical data and the admission office admits patients for all specialisms.

The following techniques were used to analyze this data:

**Descriptive Statistics** Descriptive statistics were used to give a quick overview of collected data. Measures like average and standard deviation are included in descriptive statistics. Using these measures, differences in the LoS of different patient categories were identified that would need further investigation with the methods described below.

**KS-test** When the descriptive statistics suggested that two samples were statistically different, non-parametrical statistical tests where used to test whether the difference was significant. Non-parametrical test do not rely on the estimation of distribution parameters (such as the mean or the standard deviation) describing the distribution of the sample [8].

The specific non-parametrical test that was used in our research is called the Kolmogorov-Smirnov test (KS-test). KS-tests can be used for one sample or two samples: whether a sample differs significantly from a hypothesized distribution or whether two samples differ significant respectively. KS-tests are used in this chapter and in Chapter 5 to analyze the results of the experiments.

**Correlation** To test whether two samples are correlated, the Pearson product-moment correlation coefficient (PMCC) was used. PMCC is a measure to see if two samples increase or decrease together [8].

**Process Mining** To obtain insight in the clinical path that CTS patients follow, process mining [17] was used. Process mining is a technique used to retrieve a process model from process logs. A process log contains information about the order in which tasks were executed. Each individual execution of the process that is logged, is called a trace. A trace contains the time of the beginning of each task that is executed in the process and the time of the end of the task. Using a number of traces, process mining can be used to create a
3.2 Results of the data analysis

In this paragraph we present the results of the data analysis. First, the results of the analysis on the LoS of patients from different specialisms at the ICU are presented. Next, the results of the analysis on the LoS of CTS patients at the ICU are presented. Finally, the mining results are presented.

3.2.1 General Findings

Table 3.1 on page 84 shows the abbreviation, the Dutch meaning and the English meaning of each surgical specialism. Furthermore, this table shows the number of patients that were
admitted in 2005 and the percentage against all patients admitted at the CZE (denoted as the patient population). In this table we can see that CTS represents 8.23% of the patient population and thus forms the fourth biggest specialism. In Table B.3 on page 85 the number of patients admitted to the ICU can be seen. 57.96% of the patients admitted to the ICU are CTS patients. CTS is the biggest specialism at the ICU. Next are the general surgery patients with 16.98%. Table B.4 on page 85 shows the number of patients and percentages of the variants of the patient path. This table depicts that about 60% of the patients is HC and about 40% PACU.

3.2.2 LoS of patients from different ICU specialisms

The LoS in days of patients of different specialisms that are admitted to the ICU are pre-

presented in Table 3.1. In this table we can see that CTS patients have an average LoS of 1.55 days (±4.12 stdev). The neurosurgical patients\(^2\) are the patients with the longest LoS in the surgical patient category. From the acute patient group, respiratory patients have the longest LoS, namely 1.99 days (±7.07). In the surgical patient group, trauma patients have the longest LoS, namely 4.11 days (±6.35).

3.2.3 LoS of CTS patients at the ICU

In Table 3.2 the LoS in days for four CTS urgency categories is presented: elective and urgent, in house, referred and emergency. An elective patient has to wait for surgery on the waiting list. An urgent patient requires surgery within one week. An emergency patient requires immediate treatment. Elective and urgent CTS patients stays for 1.29 days (±3.46) at the ICU. Emergency patients are due to stay longer at the ICU, namely 5.11 days (±8.45). An in house patient has to wait for surgery at the waiting list, like an elective patient. The last category is the referred patient category. Patients that belong to this category are

\(^{2}\)A remark must be made that the CZE does not actually facilitate neurosurgical interventions, this is done by hiring a neurosurgical specialist from another hospital.
### Table 3.1: LoS at the ICU in days for the different specialisms

<table>
<thead>
<tr>
<th>Surgical/Acute</th>
<th>Category</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surgical</td>
<td>thoracic surgery</td>
<td>1.55</td>
<td>4.12</td>
</tr>
<tr>
<td></td>
<td>respiratory surgery</td>
<td>1.99</td>
<td>7.07</td>
</tr>
<tr>
<td></td>
<td>general surgery</td>
<td>2.26</td>
<td>5.15</td>
</tr>
<tr>
<td></td>
<td>orthopaedics</td>
<td>.76</td>
<td>2.37</td>
</tr>
<tr>
<td></td>
<td>traumatology</td>
<td>4.11</td>
<td>6.35</td>
</tr>
<tr>
<td></td>
<td>urological surgery</td>
<td>1.63</td>
<td>2.56</td>
</tr>
<tr>
<td></td>
<td>gynaecology</td>
<td>.37</td>
<td>1.35</td>
</tr>
<tr>
<td></td>
<td>neurosurgery</td>
<td>12.00</td>
<td>16.97</td>
</tr>
<tr>
<td></td>
<td>other surgery</td>
<td>1.14</td>
<td>3.50</td>
</tr>
<tr>
<td>Acute</td>
<td>cardiology</td>
<td>4.31</td>
<td>8.27</td>
</tr>
<tr>
<td></td>
<td>respiratory</td>
<td>6.52</td>
<td>10.24</td>
</tr>
<tr>
<td></td>
<td>digestive</td>
<td>1.97</td>
<td>4.16</td>
</tr>
<tr>
<td></td>
<td>renal</td>
<td>1.35</td>
<td>2.62</td>
</tr>
<tr>
<td></td>
<td>CNS</td>
<td>3.58</td>
<td>7.39</td>
</tr>
<tr>
<td></td>
<td>endocrine</td>
<td>1.93</td>
<td>3.56</td>
</tr>
<tr>
<td></td>
<td>haematological</td>
<td>1.13</td>
<td>1.13</td>
</tr>
<tr>
<td></td>
<td>skin / muscle / bone</td>
<td>1.33</td>
<td>2.09</td>
</tr>
<tr>
<td></td>
<td>intoxication</td>
<td>.84</td>
<td>1.66</td>
</tr>
<tr>
<td></td>
<td>other acute</td>
<td>2.19</td>
<td>4.39</td>
</tr>
</tbody>
</table>
Table 3.2: LoS at the ICU in days for the different patient types

<table>
<thead>
<tr>
<th>Patient type</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elective and urgent CTS patients</td>
<td>1.29</td>
<td>3.46</td>
</tr>
<tr>
<td>Emergency CTS patients</td>
<td>5.11</td>
<td>8.45</td>
</tr>
<tr>
<td>In_house CTS patients</td>
<td>1.47</td>
<td>3.91</td>
</tr>
<tr>
<td>Referred CTS patients</td>
<td>5.60</td>
<td>8.89</td>
</tr>
</tbody>
</table>

transferred from another hospital, because they need emergency treatment. Therefore, this category is as urgent as the emergency patient category.

Table 3.2 shows that the average LoS for emergency patients is longer compared to planned patients. The two samples differ statistically significant at a confidence level of 95%, which is founded by the two-sample KS-test. The confidence level of 95% will be used throughout this research.

The same conclusion holds for the in_house patient group compared to the referred patient group. This can be explained by the fact that the referred patient group can be considered as urgent; CTS patients that are referred from another hospital are always in need of urgent treatment.

To test whether the LoS at the ICU depends on the duration of the surgery, the correlation between the corresponding samples was determined. The following conclusion can be drawn:

Using Figure B.2 on page 88 and the correlation coefficient of 0.000, the conclusion can be drawn that there is no linear relationship between the samples. We can therefore conclude that the LoS at the ICU does not depend on the duration of the surgery.
3.3 Variants of the patient path

As already mentioned before, two variants of the patient path are considered in this thesis. These variants of the patient path are denoted by HC and PACU. In the remainder of this paragraph, a summary of both variants is given, together with a decision tree per variant.

**HC** A HC patient is admitted to the surgical ward and prepared for surgery. After surgery, the patient will receive post-operative care at the HC. Within 24 hours of the patient’s stay at the HC and when the patient’s condition is stable, the patient is transferred to the surgical ward. When the patient has to stay longer than 24 hours at the HC, the patient is transferred to either the IC or the MC. The LoS of the different care levels for this variant of the patient path can be seen in Table B.7 and Table B.8 on page 86 and in Tree 3.3.2.

**PACU** When surgery is completed on a PACU patient, the patient is admitted to the PACU. PACU patients can stay at the PACU for at most 10 hours. When a patient has to stay longer than 10 hours, the patient is admitted to the IC, the MC or the patient can stay at the HC when a bed is available. When the patient’s condition is stable, the patient is transferred to the surgical ward.

In Tree 3.3.1, a taxonomy is presented of the different patient classifications in the CZE. Tree 3.3.2 and 3.3.3 represent the two variants of the patient path.
Tree 3.3.1: Taxonomy of the different patient classifications in the CZE.

Tree 3.3.2: Decision tree of the HC variant of the patient path.
### 3.3. Variants of the patient path

#### Tree 3.3.3: Decision tree of the PACU variant of the patient path.

---

#### 3.3.1 Statistical distributions of the LoS

We have seen two patient classifications in this and the previous chapter, namely a classification based on the variants of the patient path and a classification based on the urgency of patients. Furthermore, we have concluded that both classifications contain patient groups that differ significantly from each other in LoS. Nevertheless, we only use the classification based on the variants of the patient path when determining the LoS, because in the simulation system for the planning of the CTS patient processes a design choice is made in which we do not distinguish whether a patient is a planned patient or an emergency patient. This is due to the fact that we wanted to evaluate the patient processes of the variants of the patient path in terms of throughput of patients. To find the underlying statistical distribution of the LoS at the ICU for both variants of the patient path, several distributions\(^3\) were tested and ranked according to their goodness-of-fit. Goodness-of-fit describes how well a

---

\(^3\)The distributions were Poisson, Beta, Gamma, Erlang, Lognormal, Normal, Exponential, Weibull, Triangular and Uniform
Table 3.3: Parameters for the Gamma distribution for the HC variant

<table>
<thead>
<tr>
<th>Ward</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>2.88 hours</td>
<td>2.18 hours</td>
</tr>
<tr>
<td>IC</td>
<td>73.2 hours</td>
<td>0.701 hours</td>
</tr>
<tr>
<td>MC</td>
<td>20.6 hours</td>
<td>1.07 hours</td>
</tr>
</tbody>
</table>

Table 3.4: Parameters for the Gamma distribution for the PACU variant

<table>
<thead>
<tr>
<th>Ward</th>
<th>$\alpha$</th>
<th>$\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>PACU</td>
<td>9.05 hours</td>
<td>1.1 hours</td>
</tr>
<tr>
<td>IC</td>
<td>70.0 hours</td>
<td>0.839 hours</td>
</tr>
<tr>
<td>MC</td>
<td>35.7 hours</td>
<td>0.99 hours</td>
</tr>
</tbody>
</table>

statistical model fits a set of observations. The distribution that ranked with the overall best goodness-of-fit was the Gamma distribution. The Gamma distribution is parameterized by two parameters. The parameters for each individual postoperative care level in the patient path are presented in Table 3.3 and Table 3.4. These parameters denote the LoS in hours. When a patient is admitted to the surgical ward, we do not distinguish between the LoS of the HC and PACU patients. The choice is based on expert opinion and the suitability of available data. Therefore, when determining the distribution of the surgical ward, a divergence is made from the Gamma distribution. The Lognormal distribution had a better goodness-of-fit than the Gamma distribution. The Lognormal distribution takes two parameters, the average and the standard deviation. The parameters for CTS patients staying at the surgical ward are presented in Table 3.5.

Table 3.5: Parameters for the Lognormal distribution for CTS patients

<table>
<thead>
<tr>
<th>Ward</th>
<th>mean</th>
<th>stdev</th>
</tr>
</thead>
<tbody>
<tr>
<td>Surgical ward</td>
<td>120 hours</td>
<td>22.08 hours</td>
</tr>
</tbody>
</table>
3.4 Mining results

The Heuristic Mining Algorithm rediscovered a process model that corresponded very well to the process models that were created based on interviews, which were described in Paragraph 2.4. The process model is presented in Figure B.3 on page 89. In this process model, the HC and PACU variants of the patient path can be seen, corresponding to the lower and upper flow, respectively. The schedule transition refers to admission of a patient to a bed. The level of care is denoted by an @-sign. The transition marked as complete refers to the event that the patient is discharged.

3.5 Conclusion

In this chapter the data analysis is presented. This data analysis is focussed on data from three hospital information systems: EZIS, ICIS and Cardio2000, and data from the admission office. The most important result in this chapter is the analysis of the LoS and the evaluation of distribution for the LoS for the patient path. In the next chapter, a description of the agent-based analysis and design steps of the simulation system for the planning of patient processes for CTS patients is given. In the simulation system, the results of this chapter and the previous chapter are used.
Chapter 4

Description of the agent-based simulation system

In this chapter we present the analysis, design and implementation of the simulation system for the planning of the CTS patient processes. First, a description of general agent concepts like the goal and utility, with respect to our simulation system is provided. Next, the results of the system analysis phase are shown, based on a methodology presented by Nikraz [13], consisting of five incremental steps. The design phase is based on work from Deloach [3, 4] and Zambloni [22]. In the design phase, the results of the system analysis phase are translated into a system design. The fourth paragraph contains a discussion about the implementation phase of the simulation model. A conclusion is presented in the last paragraph.

4.1 Towards an agent-based simulation system

As already described in the introduction of this thesis, an agent is a software module situated in some environment capable of performing actions in that environment in order to meet its design objective [19]. The environment in which the agents in our system are operating,
is called the world. The world is an object from which the agents can request information, such as the current date or the number of patients on the waiting list.

The agent can perform actions like sending and receiving messages. These messages contain proposals of patients and answers to received proposals.

The last aspect of the agent definition is the design objective or the reason that the agent exists. In our system, the design objective of each individual agent is to admit a proposed patient when a bed is available, and to discharge a patient when the LoS for that patient is fulfilled.

The objective of the MAS is to schedule patients through the surgery and postoperative care as quickly as possible, where each agent is responsible for one care level.

### 4.1.1 Methods

The methodology that was used to design the simulation system for the planning of the CTS patient processes is described in Nikraz [13]. It is a methodology proposed for the development of MAS using the Java Agent DEvelopment (JADE) platform. JADE is a platform that uses middle-ware for the implementation of MAS. Middle-ware is computer software that connects software components or applications. JADE complies with the Foundation of Intelligent Physical Agents (FIPA)\(^1\) specifications. JADE focuses on the key issues in the analysis and design of MAS.Due to performance considerations concerning the extensive hospital simulations, we have chosen not to use JADE as an implementation platform. In the design phase, we have used a mixture of methodologies. Besides the methodology presented by Nikraz [13], also the methodologies proposed by Deloach [3] [4] and Zambloni [22] were used.

\(^1\)The FIPA promotes agent-based technology and the interoperability of its standards with other technologies.
4.2 Simulation system analysis

4.1.2 The name ABOCSS

The simulation system for the planning of the CTS patient processes is given the name ABOCSS. ABOCSS can be explained in two ways: 1) the Agent-Based OR and Care Scheduling System and 2) the Agent-Based OR and Care Simulation System. Both names describe what the application is intended for: simulating the planning of the CTS patient processes.

4.2 Simulation system analysis

In this paragraph, the analysis phase of ABOCSS is described. As mentioned in the introduction of this chapter, this phase consists of five steps: the definition of the UML Use Cases, the initial agent types definition, the responsibilities definition, the acquaintances definition and the agent refinement. In the remainder of this paragraph, each step is described.

The assumptions about the analysis are presented below.

Assumptions:

- When an operating theater is required, a fully equipped operating theater is meant (i.e. with all necessary technical equipment and staff for performing the surgery).

- The waiting list is considered given to the agents. Therefore, an agent has no control over who enters the waiting list and the position that a patient shifts on the waiting list.

- The surgery time slots for HC and PACU patients are given to the agents.

- A bed at the surgical wards can usually be made available.

- We assume that patients following the PACU variant of the patient path are not routed to the HC variant of the path and vice versa.
### 4.2.1 Step 1: Use Cases

The first step in the methodology used is the definition of UML Use Cases. UML Use Cases describe the functionalities of a system. The UML Use Cases can be seen in Figure 4.1 on page 30.

<table>
<thead>
<tr>
<th>identification</th>
<th>name</th>
<th>description</th>
<th>preconditions</th>
<th>postconditions</th>
<th>exceptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>UC1</td>
<td>Remove patient from system</td>
<td>The patient is released from the hospital either because the patient is healed or because the patient died in the process.</td>
<td>The patient was no longer on the waiting list and was in or after surgery.</td>
<td>The patient is removed from the system and the (possibly) scheduled bed is released.</td>
<td>-</td>
</tr>
<tr>
<td>UC2</td>
<td>Do surgery planning</td>
<td>A planning is set up for the surgery, taking the available OR capacity, the waiting list and the availability of postoperative care into account.</td>
<td>-</td>
<td>A planning is set up for the OR.</td>
<td>-</td>
</tr>
<tr>
<td>UC3</td>
<td>Schedule surgical emergency patient</td>
<td>A patient in need of urgent surgery is admitted to the system.</td>
<td>There is a patient in urgent need of surgical treatment.</td>
<td>The patient is scheduled for surgery as quickly as possible and schedules of the IC and OR are updated.</td>
<td>-</td>
</tr>
</tbody>
</table>
4.2. Simulation system analysis

<table>
<thead>
<tr>
<th>identification</th>
<th>UC4</th>
</tr>
</thead>
<tbody>
<tr>
<td>name:</td>
<td>Admit acute patient</td>
</tr>
<tr>
<td>description:</td>
<td>A patient in need of urgent IC treatment is admitted to the system.</td>
</tr>
<tr>
<td>preconditions:</td>
<td>There is a patient in urgent need of IC treatment.</td>
</tr>
<tr>
<td>postconditions:</td>
<td>The patient is admitted to the ICU as quickly as possible and schedules of the ICU are updated.</td>
</tr>
<tr>
<td>exceptions:</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>identification</th>
<th>UC5</th>
</tr>
</thead>
<tbody>
<tr>
<td>name:</td>
<td>Transfer patient</td>
</tr>
<tr>
<td>description:</td>
<td>A patient’s condition implies that the patient should either go to a lower care ward or a higher care ward.</td>
</tr>
<tr>
<td>preconditions:</td>
<td>-</td>
</tr>
<tr>
<td>postconditions:</td>
<td>The care planning is re-created and communicated to the wards.</td>
</tr>
<tr>
<td>exceptions:</td>
<td>No bed is available.</td>
</tr>
</tbody>
</table>

### 4.2.2 Step 2: Initial Agent Types Identification

In this step, we define the initial agent types that were identified from the UML Use Cases. The following agent types can be distinguished in our simulation system:

1. waiting list agent;

2. operating theater agent;

3. postoperative care agent.

The waiting list agent schedules patients from the waiting list into surgery and postoperative care. The operating theater agent and the postoperative care agent are responsible for the scheduling of OR’s and beds. These agent types are presented in Figure 4.2 on page 31.
Figure 4.1: Use Case Definition of ABOCSS.
Figure 4.2: Initial agent types definition of ABOCSS.

<table>
<thead>
<tr>
<th>Agent type</th>
<th>Responsibilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Waiting list agent</td>
<td>The purpose of the waiting list agent is to schedule patients from the waiting list into surgery.</td>
</tr>
<tr>
<td>Operating theater agent</td>
<td>The operating theater agent is responsible for OR loading.</td>
</tr>
<tr>
<td>Post operative care agents</td>
<td>The post-operative care agent is responsible for bed allocation of postoperative care beds.</td>
</tr>
</tbody>
</table>

Table 4.1: Responsibility table

This figure shows which actor can communicate with which agent. Communication is shown by an arrow. As can be seen in this figure, all actors can communicate to all agents.

4.2.3 Step 3: Responsibilities Identification

In this step, the main responsibilities per agent type are summarized in the responsibility table (Table 4.2.3). In the responsibility table, the responsibilities of the different agents, derived in 4.2.2, are presented.
4.2.4 Step 4: Acquaintances Identification

In this step, the acquaintances of the agent types from Section 4.2.2 are described. The interactions between the agent types are specified together with the interactions between agents and human actors. The acquaintances are presented in Figure 4.3 at page 33. In this figure we can see that the actors communicate to the resource and scheduling agents via so called transducer agents. We can define a transducer agent as an interface between a human user and the agents [13]. Furthermore, we can see that the resource and scheduling agents are able to communicate to each other and to the transducer agents via an Agent Communication Language (ACL). Communication between agents is based on ACL.

4.2.5 Step 5: Agent Refinement

In this step, the agents defined in Section 4.2.2 are refined based on a number of considerations: support, discovery of agents and management and monitoring.

Since our agent-based system consists only of the fixed number of agents described in the previous steps, there is no need to describe the discovery of agents. Management and Monitoring relates to the requirement that the system should keep track of existing agents, or the starting and stopping of agents on demand. This is not of our interest in ABOCSS. The first aspect, support, is relevant because it elaborates on the kind of support information that an agent needs to accomplish its responsibilities. Next, a description of the support information that each agent type needs, will be given.

**Waiting list agent:** The waiting list agent needs to know whether the patient is a PACU or HC patient. This way the agent can provide the operating theater agent with a patient proposal for the scheduling of OR’s.

**Operating theater agent:** The operating theater agent needs information about the current OR’s and the time slots available.

**Post operative care agent:** The postoperative care agent needs to know to which variant
Figure 4.3: Agent acquaintances definition of ABOCSS.
of the patient path a patient belongs and with this, the mean LoS and standard deviation of the mean LoS for the postoperative care level the agent is responsible for. The postoperative care agent needs to know how much beds are available.

4.2.6 Summary of the analysis phase

In the analysis phase five steps have been completed to analyze our simulation system for the planning of the CTS patient processes. In each of these steps, the identified agent types are improved. Finally, the following agent types are defined: a waiting list agent, an operating theater agent and a postoperative care agent. In the next paragraph, the design phase is described. In the design phase, the MAS is described that contains the agent types from the analysis phase.

4.3 Simulation model design

This paragraph contains a description of three steps used to create a design for ABOCSS. Input to the design is the analysis, described in Paragraph 4.2. The outcome of the design is a MAS and a set of class diagrams. A class diagram describes the structure of a system by showing the system's classes and the relationships between the classes, called links. Furthermore, generalization relations are specified, indicating that if, for example, two classes are related, then the more general type is denoted by the supertype and the more specific one is denoted by the subtype.

4.3.1 Step 1: Agent splitting/merging/renaming

This section is devoted to the splitting, merging and renaming of the three agent types. The postoperative care agent is split into the PACU, HC, IC, MC and surgical ward agent, because this corresponds to the different care levels currently used at the ICU. An explanation about the different care levels used at the ICU can be found in Paragraph 2.1. Now each
postoperative care agent can represent a specific postoperative care level and schedule the patients according to the availability of beds.

In the initial agent definition (Figure 4.2 on page 31), the operating theater agent and the waiting list agent are different agent types. Since we consider only the CTS waiting list and the OR’s for this specialism, the waiting list agents and the operating theater agent are merged into the OR Scheduling agent. This agent will have the task to schedule patients for surgery and transfer the patients to postoperative care.

4.3.2 Step 2: Interaction Specification

Unlike the method presented by Nikraz [13], UML sequence diagrams are used for specifying interactions in this research. A UML sequence diagram shows the messages parsed between different objects. The usage of UML sequence diagrams is advocated in DeLoach [3 4]. All Use Cases have the interaction mechanism in common, which is described in Figure C.1 on page 92 in the appendices.
The agent interaction specification describes the way agents communicate with each other. The interaction process can be described as follows: Agent A sends a patient proposal to agent B. This patient proposal consists of a number of patients that agent A wants to transfer to agent B. Agent B receives the patient proposal and calculates whether agent B has beds available for admitting the proposed patients. If agent B has beds available, agent B will accept the proposal. If agent B hasn’t got enough beds available, agent B will send a proposal to agent C. This proposal contains patients currently admitted to beds managed by agent B that can be transferred to agent C (i.e. the care level that agent C is responsible for, is a higher care level or the condition of the patients is good enough in such a way that the patient can be transferred to agent C). When agent C accepts this proposal, agent B has beds available and hence can admit the patients proposed by agent A.

If agent B receives a negative response from agent C in such a way that agent C cannot admit any patients from agent B, and agent B has beds available, but not enough to admit all patients proposed by agent A, agent B can formulate a counter proposal. This counter proposal holds the patients that agent B is willing to admit. Agent A can either accept (and send the patients) or decline. If agent A receives a decline, agent A can propose the patients to an agent that is responsible for the same or a higher care level as agent B.

In the current version of ABOCSS, the agents admit proposed patients when there are enough beds available and reject a proposal when the requested beds are occupied. Sending counter proposals is not implemented in the current version. Furthermore, when a proposal of an agent is rejected the agent would, in the scenario described above, propose its patients to another (from the same or a higher level of care). This is not implemented in the current version of ABOCSS.
4.3.3 Step 3: Creating Agent and Related Classes

The methodology presented by Nikraz [13] contains no phase to transform the created models into Java Code. Therefore UML Class Diagrams were used. The usage of UML class diagrams is advocated in Deloach [3, 4]. The class diagram of ABOCSS can be found in Figure 4.5 on page 46. The generalization relation of the agent and resource classes can be seen in Figure 4.6 and 4.7. A description of each class is given below.

**Agent** The agent class is a supertype that represents an agent. The agent class holds variables and functions that are common to all agents. The agent generalization relation can be seen in Figure 4.6 on page 47.

**State** The state class represents the state that the agent is currently in and by means of this state, the agent knows which actions the agent must perform. The default is *inactive* but the state can change over time by receiving and sending proposals.

**Language** The language class is an implementation of the ACL.

**Proposal** A proposal is an object that contains a number of patients, a sender, a receiver.

An example of a proposal can be seen in Paragraph 4.4.2.

**Status** The status class represents the progress of a proposal, for example *created* or *rejected*.

**ProposalMemory** The proposal memory is used to store received and sent proposals. This is necessary, because an agent wants to keep track of proposals and counter-proposals.

**Bed** The bed class is a supertype that represents a bed at a care level. Bed contains a boolean to indicate whether the bed is occupied and a patient reference to get the patient occupying the bed. For each care level (HC, PACU, MC, IC and surgical ward) from Paragraph 2.1 on page 7, a subtype of bed is created. The generalization relation of bed can be seen in Figure 4.7 at page 47.
Patient  The patient class represents a patient. The patient class is a supertype and the
generalization relation can be found in Figure 4.8 on page 48. The generalization
relation consists of this supertype patient where all attributes that are common to
surgical and acute patients can be set (like the ID of the patient and the date of
arrival). Subtypes of patients are further classified according to the specialism that is
going to treat them as described in Paragraph 3.2.1 on page 15.

CareProfile  This class represents the variant of the patient path. This can either be HC
or PACU as described in Paragraph 2.2 on page 8.

World  The world represents an interface between an agent and the clock and the waiting
list. The agents retrieve patient information and date and time information through
the world.

Clock  The clock represents the date and time in ABOCSS.

Waitinglist  The waiting list class represents a list of patients that are waiting for surgery.

        An agent can request patients from the waiting list through the world. The agent can
        specify the variant of the patient path when requesting a patient.

4.3.4 Summary of the Design phase

To summarize the results of the design phase: The waiting list agent is merged with the
operating theater agent into an OR Scheduling agent. The postoperative care agent is split
into five different agents, the PACU agent, the HC agent, the IC agent, the MC agent
and the surgical ward agent. Further, we have identified a set of classes and generalization
relations for the implementation of ABOCSS.
4.4 Simulation system implementation

The design of the previous paragraph showed a number of classes in the UML Class Diagrams connected to each other by links and generalization relations. From this design, it is possible to make an implementation. However, there are some architectural choices to be made regarding the Agent Communication Architecture (ACA) and the ACL. These choices, together with a description of the agent algorithms, are presented in this paragraph.

4.4.1 Agent Communication Architecture

The ACA describes the way in which agents communicate with each other. For the implementation of the ACA there are several integration patterns available [21]:

File transfer The agents use a shared file to communicate to each other. This means that one agents writes a message in a file and the other agents can retrieve this message by reading it from the file.

Shared database Communication via a shared database is similar as the communication via file transfer, but instead of a file, the agents perform write and read actions on a shared database.

Raw data transfer The agents communicate via a network transfer protocol or shared memory. The more common name for this pattern is communication via sockets.

RPC RPC is a protocol that allows a computer program running on one computer to call a function in another component, to be executed without the programmer explicitly implementing the details for this interaction.

Messaging The so-called Message Oriented Middle-ware is also referred to as the bus pattern. This concept relies on asynchronous message-passing. The idea is that objects send messages via a shared bus (or a queue). This bus will take care of the delivery of the message.
Below, a discussion is provided in which we will substantiate our selection of the ACA. The communication architecture should be decentralized, as the agent paradigm suggests. The file transfer pattern and the shared database pattern are centralized architectures. Using a shared file or database, agents communicate to each other via write and read actions on either the file or the database. What we want is decentralized communication architecture and coordination via communication.

Messaging is normally the best option for a MAS, but then we are talking about a MAS where agents can register real time, never to know how many agents and what kind of agents are present in the network. The messaging pattern is used in the implementation of JADE. For ABOCSS, messaging is too expensive to implement, because we already know which agent types (and how many of each type) are present, and the location (IP-address and port) of each agent will also be available (or can be set via a configuration file).

The remaining patterns are RPC and Raw Data Transfer. If we should use RPC, then the agents communicate by calling functions of each other. Because one agent should not have knowledge of the implementation of another agent, we eliminate the RPC pattern and hence stick to the Raw Data Transfer communication pattern (i.e. communication via sockets). Raw Data Transfer resembles the decentralized agent communication best. With Raw Data Transfer, each agent has a communication channel that can be used to send or receive messages.

4.4.2 Agent Communication Language

For the implementation of the ACL, we have three possibilities:

- XML based conversations;
- ASCII based conversations;
- KQML based conversations [7].
The advantage of XML based communication is that XML messages are structured and self-describing, and that many applications can be adapted to process XML messages. The disadvantage of XML is that the implementation of the communication is not standardized. ASCII messages have the same disadvantages as XML based conversations, plus, ASCII messages have no defined structured context. KQML on the other hand is already a structured implementation for message parsing between agents. The syntax of KQML is based on a balanced parenthesis list. The initial element of the list is a performative and the remaining elements are the performative’s arguments as keyword/value pairs\textsuperscript{[7]}. KQML has the possibility, since it is the standard in agent communication, to be implemented with the JADE platform (in possible future versions of the ABOCSS). Based on this discussion, KQML is chosen as the ACL.

Below, an example of KQML code can be seen in which the OR_SCHEDULING agent tells the PACU agent that he wants to propose four patients from the PACU variant of the patient path:

$$
\begin{align*}
\text{(tell} & \text{:sender OR_SCHEDULING} \\
& \text{:receiver PACU} \\
& \text{:content (propose} \\
& \text{(id 1175707391424} \\
& \text{(patient :p_id "0" :category "PACU" :weight "0")} \\
& \text{(patient :p_id "1" :category "PACU" :weight "0")} \\
& \text{(patient :p_id "2" :category "PACU" :weight "0")} \\
& \text{(patient :p_id "3" :category "PACU" :weight "0")}\text{)} \\
\text{)}
\end{align*}
$$
4.4.3 Implementation of algorithms

In Algorithm 1 on page 43, the implementation of the OR Scheduling agent working algorithm can be seen. The working algorithm is the engine of the agent. This algorithm is a loop in which the agent performs tasks. The OR Scheduling agent schedules patients for admission to the surgical ward once a day. When the agent identifies the current time as a HC time slot, the agent will schedule a surgery for a HC patient. Otherwise, when the agent identifies the current time as a PACU time slot, a surgery for a PACU patient is scheduled. When a surgery is completed, the agent will transfer the patient to the next ward in the patient path. The last action that the agent performs, is to check its proposal memory for bounced proposals (that could not be sent due to network errors). The OR Scheduling agent will resend the bounced proposals. This loop continues for a specified number of iterations or a number of patients to be admitted.

The second algorithm (Algorithm 2 on page 44) is the working algorithm of the resource agent (MC agent, IC agent, PACU agent, HC agent and the surgical ward agent). This algorithm works like the OR Scheduling agent’s working algorithm, with one difference: resource agents work on triggers, possibly from other agents and triggers from patients currently admitted to the care level that the agent is responsible for, that are ready to be transferred to another care level or ward. When a resource agent receives a proposals, the agent performs an action needed for that particular proposal, like admit the patients from the proposal or reject a proposal when insufficient beds are available.

The algorithm that all agents have in common is the decideToAdmit algorithm which is described by Algorithm 3 on page 44. This algorithm is used to make a decision whether to admit patients from a proposal. The proposal is accepter when the number of available beds is at least as big as the number of proposed patients. Otherwise, the proposal is rejected.
Algorithm 1 OR Scheduling agent working algorithm

1: start communication listener  
2: for all resources do  
3: initialize resource  
4: end for  
5: currentDay := World.getCurrentDay()  
6: run := true  
7: while run do  
8: if currentDay \neq World.getCurrentDate() then  
9: schedule patients for surgical ward  
10: currentDay := World.getCurrentDate()  
11: end if  
12: if PACUdateAndTime then  
13: request PACU patients  
14: else \{HCdateAndTime\}  
15: request HC patients  
16: end if  
17: for all patientToBeTransferred do  
18: transfer patient to next care level in patient path  
19: end for  
20: for all proposalsWithStatusBounced do  
21: resend proposals  
22: end for  
23: if oneYearHasPassed or nrOfAdmittedPatient > userDefinedLimit then  
24: run := false  
25: end if  
26: end while  
27: print result
Algorithm 2 Resource Agent working algorithm
1: start communication listener
2: for all resources do
3: initialize resource
4: end for
5: run := true
6: while run do
7: for all patientToBeTransferred do
8: transfer to next ward in patient path
9: end for
10: for all proposalsWithStatusBounced do
11: resend proposals
12: end for
13: if oneYearHasPassed or nrOfAdmittedPatient > userDefinedLimit then
14: run := false
15: end if
16: end while
17: print result

Algorithm 3 Decide to admit algorithm
1: currentDay := World.getCurrentDay()
2: if currentDay.day ≥ openingDay and currentDay.day ≤ closingDay and
   currentDay.hours ≥ openingTime and currentDay.hours ≤ closingTime then
3: if numberOfFreeBeds ≥ nrOfProposedPatients then
4: admit patients
5: else
6: reject patients
7: end if
8: else
9: reject patients
10: end if
4.4.4 Summary of the Implementation phase

In Paragraph 4.4 some implementation related issues were discussed. The first issue was the ACA, in which the used protocol to facilitate the communication between agents was selected. We have selected Raw Data Transfer from the available options, because this resembles the way that agents communicate best. Next, the structure of the ACL was discussed. From the three proposed options, KQML was the best suitable for agent communication. The final issue discussed in this paragraph was the definition of the algorithms that the agents use as main working loop.

4.5 Conclusion

In this chapter, a description of the simulation model was presented. First, a step-by-step system analysis and design was given using methodologies from Nikraz [13], Deloach [3, 4] and Zambloni [22]. The most important results were the selection of agent types and the formulation of the UML Class Diagrams. Implementation related issues like the ACA and the ACL are discussed in Paragraph 4.4.

In the next chapter, the results of the experiments with ABOCSS are presented and analyzed.
Figure 4.5: Class diagram of ABOCSS.
4.5. Conclusion

Figure 4.6: Generalization relation of the supertype agent.

Figure 4.7: Generalization relation of the supertype bed.
Figure 4.8: Generalization relation of the supertype patient.
Chapter 5

Experiments

In this chapter we present the results of initial experiments with ABOCSS in the form of what if scenarios. These results are initial in the sense that further research is needed with ABOCSS to improve the quality of the results. In this chapter, we show the possible scenarios that require further study.

A simulation run is the executions of a number of simulation in sequence. Each simulation run is characterized by a set of parameters like the number of patients and the opening time of the PACU. These simulation runs are called what if scenarios. Each scenario resembles a situation that could happen in the CZE. The outcome of the scenarios is explained in terms of what this means for the CZE. Finally a conclusion is drawn from the simulation results.

5.1 Evaluation measures

The measures used in the evaluation of the experiments are the individual admitted, rejected and bounced number of patients for each agent and a comparison with the initial evaluation and with the other scenarios. Furthermore, the completion date of the scenarios is compared with the completion date of the other scenarios to see if there is an improvement or deterioration. Each scenario is started on the 1st of January and the simulation is
run until the planned number of patients is admitted. Finally, the throughput of patients is evaluated by looking at the number of patients that is admitted by the OR Scheduling agent and admitted by the surgical ward agent. Both the admitted patients at the surgical ward and the completion date are considered as performance indicators in ABOCSS. The completion date shows when the OR Scheduling agent succeeds in admitting the planned number of patients. When the completion date is before the 31st of December, ABOCSS could process a higher number of planned patients, but did not, because the planning strategy of the OR Scheduling agent is not changed (except in scenario 3e and 3f). Changing the strategy of the OR Scheduling agent is beyond the scope of our research.

In two of the scenarios, we draw a preliminary conclusion on the involved costs in terms of more or less expensive care.

5.2 Evaluation

The initial result set was obtained by performing a simulation run that contains 15 individual simulations. The initial simulation run was executed using the default settings. The settings are selected in such a way, that the settings resemble the current way of working at the CZE. The default settings of ABOCSS can be seen in Figure 5.1. Most of the scenarios have a patient planning of 1800 patients. This means that the OR Scheduling agent must succeed in scheduling 1800 patients before the individual simulation run is complete.

It is important to note that all the simulations were executed on the same computer, an IBM Thinkpad T42. When ABOCSS was executed on a slower computer, ABOCSS performed worse than on the IBM Thinkpad. This is caused by the extensive communication via sockets and between agents, and the amount of messages that the agents must process. The initial results are presented in Table 5.1. The initial evaluation needs until the 17th of November to complete the simulation. In this table, three rows are depicted: admitted,
Figure 5.1: Default settings of ABOCSS.
Table 5.1: Initial run results

<table>
<thead>
<tr>
<th></th>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1803(±1)</td>
<td>910(±2)</td>
<td>886(±3)</td>
<td>245(±10)</td>
<td>117(±7)</td>
<td>1254(±7)</td>
</tr>
<tr>
<td>Bounced</td>
<td>21(±5)</td>
<td>0(±1)</td>
<td>1(±2)</td>
<td>51(±10)</td>
<td>2(±1)</td>
<td>50(±6)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±1)</td>
<td>0(±1)</td>
<td>298(±7)</td>
<td>19(±4)</td>
<td>125(±8)</td>
</tr>
</tbody>
</table>

*bounced* and *rejected*. *Admitted* counts the number of patients that were successfully admitted by an agent. *Rejected* counts the number of patients that were rejected by an agent. *Bounced* counts the number of times that a patient is rejected by an agent after an accept message from the agent where the patients were proposed to. This means that the agent sends a request, receives a positive response for admission and then the agent that has sent the request waits for too long before transferring the patients. Then the request is expired and the beds are occupied by other patients. Furthermore, five columns are depicted: *OR Scheduling, PACU, HC, IC, MC* and *Surgical ward*. Each column contains the *admitted, bounced* and *rejected* patients for the corresponding agent.

### 5.2.1 Evaluation of ABOCSS with a varied patient planning

The waiting list contains more patients than the OR Scheduling agent can admit in one simulation run. The resembles the current hospital practice because in real life, the hospital will never run out of patients to admit. In this evaluation, the planning of patients is varied to see the impact in ABOCSS. First ABOCSS is executed with a planning of 1900 patients (evaluation a). The remainder of the settings is kept the same (opening times, probabilities etcetera). In evaluation b, ABOCSS is set up to process 1600 patients. In evaluation c, the patient planning is 2000 patients. What we would like to show in these scenarios is that ABOCSS behaves the same, no matter the number of patients on the planning. The results are presented in Table 5.3 [5.4](#) and 5.5 [5.5](#) The completion dates of this scenario are presented in Table 5.2 [5.2](#)

The results of evaluation a, b and c show that the number of patients that is
5.2. Evaluation

Table 5.2: Completion dates of the initial evaluation

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Completion date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluation a</td>
<td>6 December</td>
</tr>
<tr>
<td>Evaluation b</td>
<td>11 October</td>
</tr>
<tr>
<td>Evaluation c</td>
<td>20 December</td>
</tr>
</tbody>
</table>

Table 5.3: Simulation runs with an inflow of 1900 patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1902(±2)</td>
<td>960(±3)</td>
<td>936(±4)</td>
<td>256(±16)</td>
<td>119(±5)</td>
</tr>
<tr>
<td>Bounced</td>
<td>23(±4)</td>
<td>1(±0)</td>
<td>1(±1)</td>
<td>45(±5)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>0(±1)</td>
<td>327(±18)</td>
<td>8(±7)</td>
</tr>
</tbody>
</table>

Table 5.4: Simulation runs with an inflow of 1600 patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1602(±1)</td>
<td>802(±1)</td>
<td>793(±2)</td>
<td>214(±8)</td>
<td>104(±9)</td>
</tr>
<tr>
<td>Bounced</td>
<td>10(±3)</td>
<td>0(±0)</td>
<td>1(±2)</td>
<td>23(±8)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>2(±2)</td>
<td>1(±0)</td>
<td>284(±23)</td>
<td>18(±6)</td>
</tr>
</tbody>
</table>

Table 5.5: Simulation runs with an inflow of 2000 patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>2002(±1)</td>
<td>1002(±2)</td>
<td>996(±2)</td>
<td>268(±14)</td>
<td>126(±11)</td>
</tr>
<tr>
<td>Bounced</td>
<td>5(±3)</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>24(±4)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>295(±22)</td>
<td>15(±2)</td>
</tr>
</tbody>
</table>
admitted, bounced or rejected increases or decreases together with the number of patients admitted at the OR Scheduling agent. Further, the completion date is also prolonged in evaluation a and c (where a larger number of planned patients must be scheduled), and reduced in evaluation b (where a lower number of admitted patients must be scheduled). We can therefore say that there is a linear relationship between the results of the OR Scheduling agent and the other agents. This means that ABOCSS is stable and behaves as expected.

5.3 What if-scenarios

In this paragraph we present the results of the execution of three what if scenarios: a change in the planning of resources, a change in the opening time of the PACU and a change in the probabilities of the patient path. Each of these scenarios is presented in a subsection of this section.

5.3.1 What if we changed the planning of resources

The number of beds that the IC agent and the MC agent have available varies per day. On most of the days, 2 beds are available for CTS patients. But there is a probability that 1, 3 or 4 beds can be occupied. In scenario 1a, the IC agent has 2 beds available for CTS patients. In scenario 1b the IC agent has 3 beds available for CTS patient. In scenario 1c, the IC agent has 4 beds available for CTS patients. In these scenarios we would like to show that the more IC beds there are available for CTS patients, the the better the throughput to the IC and the better the overall throughput. The results of these scenarios can be seen in Table 5.7, 5.8 and 5.9. The completion dates of the individual scenarios are presented in Table 5.6.

With respect to the initial results, where the IC has a probability of having 2, 3 or 4 beds available, we can draw the following conclusions: Scenario 1a shows a
### 5.3. What if-scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Completion date</th>
</tr>
</thead>
<tbody>
<tr>
<td>scenario 1a</td>
<td>16 November</td>
</tr>
<tr>
<td>scenario 1b</td>
<td>16 November</td>
</tr>
<tr>
<td>scenario 1c</td>
<td>16 November</td>
</tr>
</tbody>
</table>

#### Table 5.7: Scenario 1a: simulation runs with 2 dedicated IC beds for CTS patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Admitted</strong></td>
<td>1802(±2)</td>
<td>913(±2)</td>
<td>884(±4)</td>
<td>185(±17)</td>
<td>123(±1)</td>
</tr>
<tr>
<td><strong>Bounced</strong></td>
<td>27(±4)</td>
<td>0(±0)</td>
<td>1(±1)</td>
<td>44(±5)</td>
<td>1(±1)</td>
</tr>
<tr>
<td><strong>Rejected</strong></td>
<td>0(±0)</td>
<td>1(±2)</td>
<td>0(±1)</td>
<td>356(±27)</td>
<td>11(±3)</td>
</tr>
</tbody>
</table>

#### Table 5.8: Scenario 1b: simulation runs with 3 dedicated IC beds for CTS patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Admitted</strong></td>
<td>1803(±1)</td>
<td>905(±5)</td>
<td>891(±5)</td>
<td>261(±12)</td>
<td>123(±18)</td>
</tr>
<tr>
<td><strong>Bounced</strong></td>
<td>14(±10)</td>
<td>1(±2)</td>
<td>0(±0)</td>
<td>32(±16)</td>
<td>0(±1)</td>
</tr>
<tr>
<td><strong>Rejected</strong></td>
<td>0(±0)</td>
<td>1(±2)</td>
<td>0(±1)</td>
<td>204(±29)</td>
<td>8(±7)</td>
</tr>
</tbody>
</table>

#### Table 5.9: Scenario 1c: simulation runs with 4 dedicated IC beds for CTS patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Admitted</strong></td>
<td>1802(±1)</td>
<td>905(±1)</td>
<td>891(±4)</td>
<td>354(±11)</td>
<td>136(±10)</td>
</tr>
<tr>
<td><strong>Bounced</strong></td>
<td>14(±6)</td>
<td>0(±0)</td>
<td>1(±2)</td>
<td>26(±6)</td>
<td>0(±0)</td>
</tr>
<tr>
<td><strong>Rejected</strong></td>
<td>0(±0)</td>
<td>1(±2)</td>
<td>1(±1)</td>
<td>210(±11)</td>
<td>1(±1)</td>
</tr>
</tbody>
</table>
decrease in the number of admitted patients of 24.5% at the IC on average. This result is statistically significant at a confidence level of 95%, which is founded by the two-sample KS-test. Statistical significance at a confidence level of 95% holds for all conclusions presented in this chapter. The number of rejected patients at the IC increases with 19.5% on average. Scenario 1b shows an increase of 6.5% on average in the number of admitted patients at the IC with respect to the initial results. Scenario 1c shows an increase in admitted patient of 44.5% at the IC and the number of rejected patients at the IC decreases with 29.5% with respect to the initial results.

In scenario 1b, the number of admitted patients at the IC increases with 41.8% on average with respect to scenario 1a and the number of rejected patients decreases with 42.7% with respect to scenario 1a. In scenario 1c, the increase in admitted patients at the IC is 35.6% and the decrease in rejected patients is 2.9% with respect to scenario 1b. With respect to scenario 1a, 91.4% more patients are admitted to the IC and 41% less patients are rejected.

With respect to the completion date of the individual scenarios we can conclude that no improvement or deterioration can be seen when there are more IC beds available. This result can be explained by the fact that the planning of beds at the IC does affect the planning strategy of the OR Scheduling agent. The throughput of patients, on the other hand, shows a significant improvement. We can see that the inflow of patients stays the same (about 1800 patients), but the outflow of patient increases from (in scenario 1a) 1236, to (in scenario 1b) 1274 and finally (in scenario 1c) to 1332. The improved throughput of patients is also shown by the increased number of admitted patient and the decreased number of rejected patients at the IC agent: in the same period of time, a higher number of patients is admitted and a lower number or patients is rejected by the
5.3. What if-scenarios

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Completion date</th>
</tr>
</thead>
<tbody>
<tr>
<td>scenario 2a</td>
<td>17 November</td>
</tr>
<tr>
<td>scenario 2b</td>
<td>16 November</td>
</tr>
<tr>
<td>scenario 2c</td>
<td>18 November</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario 2a: simulation runs with a prolonged opening hour of the PACU</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OR Scheduling</strong></td>
</tr>
<tr>
<td>Admitted</td>
</tr>
<tr>
<td>Bounced</td>
</tr>
<tr>
<td>Rejected</td>
</tr>
</tbody>
</table>

IC agent. To achieve better throughput, as this scenario depicts, more expensive care is required (more planned IC beds for CTS patients).

5.3.2 What if we changed the PACU opening times

The opening time for the PACU at the CZE is from 12:00 until 22:00. In scenario 3a, we prolong the PACU opening time with one hour, in such a way that the PACU is opened until 23:00. In scenario 2b, the PACU is opened until 4:00. In scenario 2c, we keep the PACU opened for 24 hours a day. With these scenarios, we would like to show that less patients are rejected or bounced and the number of patients admitted to the IC decreases, when the PACU has a prolonged opening time. Overall we expect to see a better throughput. The results of this scenario can be seen in Table 5.11, 5.12, and 5.13. The completion dates of the individual scenarios are presented in Table 5.10.

<table>
<thead>
<tr>
<th>Scenario 2b: simulation runs with a PACU opening time until 4:00 hours</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>OR Scheduling</strong></td>
</tr>
<tr>
<td>Admitted</td>
</tr>
<tr>
<td>Bounced</td>
</tr>
<tr>
<td>Rejected</td>
</tr>
</tbody>
</table>
Table 5.13: Scenario 2c: simulation runs with a PACU opening duration of 24 hours

<table>
<thead>
<tr>
<th></th>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1803(±1)</td>
<td>889(±6)</td>
<td>882(±3)</td>
<td>193(±9)</td>
<td>116(±12)</td>
<td>1325(±13)</td>
</tr>
<tr>
<td>Bounced</td>
<td>31(±5)</td>
<td>17(±4)</td>
<td>0(±0)</td>
<td>7(±3)</td>
<td>1(±1)</td>
<td>77(±5)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>10(±4)</td>
<td>1(±1)</td>
<td>150(±16)</td>
<td>16(±7)</td>
<td>200(±6)</td>
</tr>
</tbody>
</table>

In scenario 2c, we can see that the PACU agent admits slightly less patients as shown in the initial results and the results of the preceding scenarios, though the PACU is opened 24 hours a day. This is caused by the fact that PACU patients can stay for at most 24 hours at the PACU and after at least 22 hours, the next patients are scheduled to the PACU. The PACU agent has to decline these proposals, because there are no beds available. One remarkable result is the fact that the IC agent admits 21.2% less patients on average. This is caused by the fact that when the PACU has a prolonged opening time, PACU patients can heal longer at the PACU and less patient need IC or MC treatment after visiting the PACU ward.

Scenario 2a and 2b show no significant improvement or deterioration on the IC results. The number of rejected patients in scenario 2c at the surgical ward is increased with 104% with respect to scenario 2a.

With respect to the completion date of the individual scenarios we can conclude that no improvement or deterioration can be seen when the PACU opening time is prolonged. This result can be explained by the fact that the PACU opening hours do not affect the planning strategy of the OR Scheduling agent. With respect to the throughput of patients, we can conclude that a prolonged opening time until 4:00 shows a significant improvement in the throughput of patients. An opening time of 24 hours shows a significant decrease of admitted patients at the IC and an equal increase in the throughput as scenario 2b. To achieve
5.3. What if-scenarios

better throughput, an lesser amount of expensive IC beds as in the initial evaluation are needed, but extra capacity is required at the PACU (for the prolonged opening times). A possible addition to this scenario would be the planning of IC beds as presented in scenario 1. This way, a trade off could be made between the costs of a prolonged opening duration of the PACU and the number of IC beds, and the better throughput.

5.3.3 What if the probabilities of patient path change

In this scenario, the probability that a patient belongs to the PACU variant of the patient path is varied. In the initial simulation results, probabilities of 60% HC and 40% PACU are used. In scenario 3a, the probability of a patient being of the PACU variant of the patient path is changed to 50%. In scenario 3b, this probability is set to 30%. The change in probabilities does not mean that there are less PACU patients than HC patients on the waiting list (in the case of PACU 30% and HC 70%); it means that from the almost infinite patient population, 30% is a PACU patient and 70% a HC patient.

Next in scenario 3c and 3d, the probabilities of the care levels after the PACU or the HC (thus from the IC or MC) in the patient path are varied. In the initial results, the probabilities for HC patients were IC: 10% MC: 5% and for PACU patients IC: 20% MC: 10%. In scenario 3c, the probability that a PACU patient will go to the IC is set to 20% and the probability that a PACU patient will go to MC is set to 10%. In scenario 3d, the probabilities for the HC variant of the patient path are: IC 10% and MC 5%.

The final scenarios (3e and 3f) have an identical setup as scenario 3a and 3b, but now the OR Scheduling agent will schedule more patients from the variant of the patient path that has a higher probability. Through these scenarios, we would like to show that the model is able to deal with changes in the environment. The results can be seen in Table 5.15, 5.16, 5.17, 5.18, 5.19 and 5.20. The completion dates of the individual scenarios are presented in Table 5.14.
Table 5.14: Completion dates of scenario 3

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Completion date</th>
</tr>
</thead>
<tbody>
<tr>
<td>scenario 3a</td>
<td>16 November</td>
</tr>
<tr>
<td>scenario 3b</td>
<td>17 November</td>
</tr>
<tr>
<td>scenario 3c</td>
<td>16 November</td>
</tr>
<tr>
<td>scenario 3d</td>
<td>15 November</td>
</tr>
<tr>
<td>scenario 3e</td>
<td>29 December</td>
</tr>
<tr>
<td>scenario 3f</td>
<td>15 November</td>
</tr>
</tbody>
</table>

Table 5.15: Scenario 3a: simulation runs with a 50% PACU patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>CTC_HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1803(±2)</td>
<td>909(±2)</td>
<td>899(±4)</td>
<td>250(±13)</td>
<td>120(±3)</td>
</tr>
<tr>
<td>Bounced</td>
<td>20(±7)</td>
<td>0(±0)</td>
<td>0(±1)</td>
<td>45(±12)</td>
<td>3(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>1(±1)</td>
<td>0(±1)</td>
<td>278(±15)</td>
<td>15(±5)</td>
</tr>
</tbody>
</table>

Table 5.16: Scenario 3b: simulation runs with a 30% PACU patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>CTC_HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1802(±1)</td>
<td>899(±11)</td>
<td>891(±4)</td>
<td>243(±7)</td>
<td>121(±11)</td>
</tr>
<tr>
<td>Bounced</td>
<td>13(±11)</td>
<td>2(±4)</td>
<td>0(±1)</td>
<td>28(±3)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>4(±6)</td>
<td>2(±1)</td>
<td>304(±15)</td>
<td>15(±4)</td>
</tr>
</tbody>
</table>

Table 5.17: Scenario 3c: simulation runs with a change in the path of PACU patients: IC 20%, MC 10% and Surgical Ward 70%

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1802(±1)</td>
<td>906(±6)</td>
<td>891(±6)</td>
<td>260(±5)</td>
<td>148(±5)</td>
</tr>
<tr>
<td>Bounced</td>
<td>14(±12)</td>
<td>0(±0)</td>
<td>1(±1)</td>
<td>37(±2)</td>
<td>3(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>1(±1)</td>
<td>329(±7)</td>
<td>24(±4)</td>
</tr>
</tbody>
</table>

Table 5.18: Scenario 3d: simulation runs with a change in the path of HC patients: IC 10%, MC 5% and Surgical Ward 85%

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1802(±1)</td>
<td>903(±3)</td>
<td>893(±3)</td>
<td>263(±12)</td>
<td>84(±8)</td>
</tr>
<tr>
<td>Bounced</td>
<td>7(±3)</td>
<td>0(±1)</td>
<td>1(±1)</td>
<td>20(±4)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±1)</td>
<td>1(±1)</td>
<td>339(±19)</td>
<td>7(±3)</td>
</tr>
</tbody>
</table>
5.3. What if-scenarios

Table 5.19: Scenario 3e: simulation runs with 30% PACU patients and 3 OR’s and IC’s for PACU patients and 5 OR’s and IC’s for HC patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1802(±1)</td>
<td>773(±20)</td>
<td>1024(±19)</td>
<td>255(±13)</td>
<td>128(±9)</td>
</tr>
<tr>
<td>Bounced</td>
<td>264(±54)</td>
<td>0(±1)</td>
<td>0(±1)</td>
<td>36(±10)</td>
<td>1(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>270(±10)</td>
<td>13(±3)</td>
</tr>
</tbody>
</table>

Table 5.20: Scenario 3f: simulation runs with 75% PACU patients and 5 OR’s and IC’s for PACU patients and 3 OR’s and IC’s for HC patients

<table>
<thead>
<tr>
<th>OR Scheduling</th>
<th>PACU</th>
<th>HC</th>
<th>IC</th>
<th>MC</th>
<th>Surgical Ward</th>
</tr>
</thead>
<tbody>
<tr>
<td>Admitted</td>
<td>1803(±1)</td>
<td>1198(±66)</td>
<td>597(±66)</td>
<td>273(±8)</td>
<td>105(±5)</td>
</tr>
<tr>
<td>Bounced</td>
<td>123(±105)</td>
<td>0(±0)</td>
<td>0(±0)</td>
<td>51(±19)</td>
<td>2(±1)</td>
</tr>
<tr>
<td>Rejected</td>
<td>0(±0)</td>
<td>3(±4)</td>
<td>0(±1)</td>
<td>346(±28)</td>
<td>12(±3)</td>
</tr>
</tbody>
</table>

As we can see in Table 5.19, scenario 3a shows no significant improvement or deterioration with respect to the initial results. This is caused by, as stated in the description of this scenario, the fact that a change in probabilities does not mean that there are less PACU patients than HC patients on the waiting list; it means that from the almost infinite patient population, 50% is a PACU patient and 50% a HC patient. The same conclusion can be drawn on scenario 3b.

With respect to the initial results, scenario 3c shows an increase in the number of admitted patients by the MC agent of 26.5%. The IC agent admits 6.1% more patients and rejects 10.4% more patients. At the surgical ward, 6.1% more patients are admitted and 31.2% more patients are rejected. In scenario 3d, the IC agent admits 7.3% more patients than in the initial results. The MC agent rejects 94.7% less patients.

In the last two scenarios, the planning is changed. In scenario 3e, 3 OR’s and PACU beds are available for PACU patients and 5 OR’s and HC beds for HC patients. The probability of a patient being of the PACU variant of the patient
path is set to 30%. Scenario 3e shows an increase in the number of bounced patients at the OR Scheduling agent. Furthermore, 15% less patients are admitted to the PACU and 15.6% more patients are admitted at the HC. This is due to the fact that for each run, $\frac{3}{8}$ of the patients is a PACU patient and $\frac{5}{8}$ is a HC patient. We can also see that the IC agent rejects 29.4% less patients and that the surgical ward agent bounces 25.6% less patients. In scenario 3f, $\frac{5}{8}$ of the patients is a PACU patient, $\frac{3}{8}$ is a HC patient and the probability of a patient being of the PACU variant of the patient path is set to 75%. Scenario 3f shows an increase in the number of bounced patients at the OR Scheduling agent and an increase in admitted and rejected patients at the IC with respect to the initial results.

With respect to the completion date of the individual scenarios we can conclude that a deterioration can be seen when the planning of the PACU surgeries and OR’s are changed. In scenario 3e, ABOCSS needs more than a month extra time to complete the simulation runs. In the other scenarios, no significant improvement or deterioration is shown. This result can be explained by the fact that the changes in the environment as in scenario 3a, 3b, 3c and 3d do not affect the planning strategy of the OR Scheduling agent. In Scenario 3f no significant improvement or deterioration is shown with respect to the completion date while the planning of resources is changed. With respect to the throughput of patients, we can conclude that ABOCSS responds very well to scenario 3e and 3f both in admitted and rejected patients at the surgical ward. The downside of scenario 3e and 3f is the high number of bounced patients at the OR Scheduling agent. A conclusion on costs cannot be drawn, because this scenario implies a change in the environment and not on quantifiable aspects. From the results of scenario 3e and 3f, we could draw a conclusion on the costs of the changed
planning of OR’s and beds, but this still would imply that the percentages of the variants of the patients path, that we cannot control, change. Therefore, this conclusion is left out.

5.4 Conclusion

In this chapter, the preliminary results of the initial evaluation and three what if scenarios are presented. These results are preliminary in the sense that further study is needed to improve the quality of the results. In this chapter we have based conclusions on the results of three what if scenarios, that can be used as a lead for further research. The used measures were the number of admitted, rejected and bounced patients for each agent, the completion date of a simulation run and the throughput that could be seen when looking at the inflow (at the OR Scheduling agent) and the outflow (at the surgical ward agent). The completion date indicates not only when the OR Scheduling agent succeeds in admitting the planned patients, but also if there is time left until the 31st of December to admit patients. No more patients than planned are admitted, because this would require the OR Scheduling agent to change its planning strategy. This however, was beyond the scope of our research.

The initial evaluation corresponds to the current way of working at the CZE. The results of the initial evaluation were analyzed to see the effects in throughput of patients. Further in the initial evaluation, the planned number of patients is varied. We have seen that these results do not differ significant from the initial results.

In scenario 1, the probability of the number of IC beds is changed. First in scenario 1a, 2 beds were available for CTS patients, then in scenario 1b, 3 bed were available for CTS patients and finally in scenario 1c, 4 beds were available for CTS patients. No improvement or deterioration with respect to the completion date can be seen when there are more IC
beds available. With respect to the throughput of patients, we have seen that the outflow of patient increases and thus can conclude that this scenario is responsible for a better throughput. Related to costs, we can state that more expensive care is required (more planned IC beds for CTS patients).

In the second scenario, we have changed the opening times of the PACU. In scenario 2a, the PACU was opened until 23:00. This scenario caused no significant effect. In scenario 2b, the PACU was opened until 4:00. This scenario caused a decrease in admitted, rejected and bounced patients at the IC and the MC. This decrease is caused by the fact that a patient can now stay (and heal) longer at the PACU and therefore it is less likely that a patient needs IC or MC afterwards. In scenario 2c, the PACU was opened 24 hours. The results of this scenario were similar to the results of scenario 2b.

With respect to the completion date of the individual scenarios we have seen no improvement or deterioration when the PACU opening time is prolonged. With respect to the throughput of patients we have seen that a prolonged opening time of 4 hours shows a significant improvement. An opening time of 24 hours shows a significant decrease of admitted patients at the IC. To achieve better throughput when the PACU is opened until 4:00, an equal amount of expensive beds is needed as in the initial evaluation. When the PACU is opened for 24 hours a day, a lesser amount of beds is needed at the IC. In both cases, extra resources are required at the PACU.

In the last scenario, the probabilities of the variants of the patient path were varied. In scenario 3a, 50% of the patients belonged to the PACU variant of the patient path. This scenario had no significant effect with respect to the initial results. The same conclusion holds for scenario 3b. In scenario 3c, the probability of the PACU variant of the patient path is varied. This caused an increase in admitted patients by the MC agent. This can be explained by the fact that there is a greater probability that PACU patients need to
go to the MC. Scenario 3d contains a change in the probabilities of the HC variant of the patient path. Since the probabilities in the patient path implicate that more patients must be admitted directly to the surgical ward, more patients are admitted and rejected by the surgical ward agent.

Scenario 3e and 3f contain a change in planning. In scenario 3e, 3 OR’s and PACU beds were available for PACU patients and 5 OR’s and HC beds for HC patients. The probability of a patient belonging to the PACU variant of the patient path was set to 30%. This lead to an increase in the number of bounced patients at the OR Scheduling agent. Furthermore, the number of bounced and rejected patients decreases at the IC, MC and surgical ward. In scenario 3f, 5 OR’s and PACU beds were available for PACU patients and 3 OR’s and HC beds were available for HC patients. This lead to an increase of rejections at the IC and a decrease of rejections at surgical ward. The completion date of scenario 3e shows a deterioration of more than a month. In the other scenarios, no significant improvement or deterioration is shown. ABOCSS improves the throughput of patients in scenario 3e with respect to the initial results and in scenario 3f. The downside of scenario 3e and 3f is the high number of bounced patients at the OR Scheduling agent.
6.1 Conclusions

In this master thesis we have evaluated patient planning at the CTS of the CZE by means of an agent-based simulation system for the planning of the CTS patient processes. Our research was focused on two main questions:

- Can an agent-based (decentralized) planning system contribute to the patient planning process at the CTS department of the CZE?
- Can an agent-based simulation system for the planning of the CTS patient processes be built, to evaluate the patient planning in terms of throughput of patients?

The answer to both research questions is yes: an agent-based planning system for the planning of the CTS patient processes can be built and an agent-based planning system can contribute to the patient planning process. Throughout this thesis we have seen a process analysis, a data analysis, a model analysis and design, and an implementation of an agent-based planning system, that was called ABOCSS. The main results of the first part (process and data analysis) of this thesis were the definition of two variants of the patient path, HC and PACU. Regular heart surgery patients are denoted by HC and a short-track...
version of regular heart surgery patients by PACU. Using these variants of the patient path, it becomes possible to plan postoperative care. The patient path consists of a number of postoperative care levels and per care level the probability that a patient will visit this care level. Further, the patient path contains a mean LoS and standard deviation of the mean LoS.

Using ABOCSS, an initial result set (the initial evaluation) is created using parameters that resemble the current way of working at the ICU. Furthermore, preliminary experiments in the form of what if scenarios are executed to see the impact on the throughput of patients and the behavior of ABOCSS. The results of the experiments are described in Chapter 5. In Chapter 5 we distinguish two performance indicators, the completion time and the throughput of patients. The completion time is the time that the OR Scheduling agent needs until the planned number of patients is scheduled. The throughput of patients can be seen when looking at the number of admissions by the OR Scheduling agent and the number of admissions by the surgical ward agent. We haven’t seen a significant improvement or deterioration with respect to the completion time, because this requires a change in the planning strategy of the OR Scheduling agent. This was beyond the scope of our research.

We have seen that better throughput of patients can be reached in two ways: 1) plan the number of IC beds and 2) change the PACU opening times.

The first alternative to reach better throughput is to plan the number of IC beds for the CTS specialism. From the experiments we have seen that when the planned number of IC beds increases, the throughput of patients shows a significant improvement. In terms of cost, we can conclude that this scenario requires a higher capacity of expensive beds than the initial evaluation.

The second alternative to reach better throughput is a prolonged opening time of the PACU.
We can conclude that a prolonged opening time until 4:00 shows a significant improvement in throughput of patients with respect to the initial evaluation. The same conclusion about throughput can be drawn when the PACU is open for 24 hours a day. Furthermore, an opening time of 24 hours shows a significant decrease of admitted patients at the IC. This is caused by the fact that when the PACU is opened for a longer period, PACU patients can heal longer at the PACU and less patient need IC or MC after visiting the PACU. In terms of cost, we can conclude that a PACU opening time of 24 hours a day requires extra capacity at the PACU and less capacity at the IC.

6.2 Recommendations and further work

Throughout this thesis, a lot of data from hospital information systems was used. This was not always an easy assignment because some data was difficult to retrieve. Gladly, we could rely on the expert opinion to get insight in the data.

The restrictions of ABOCSS are the limited negotiation function, the limited implementation of costs and the implementation of just one medical specialism. In this version of ABOCSS, we choose to implement a negotiation technique which is based on the amount of resources available to an agent. When the agent has enough resources available for the proposed patients, the agent will respond positive, else negative. This represents the current way of working at the ICU. Future ABOCSS versions should include a mechanism that is more intelligent, and base the outcome of the agents’ judgement on a utility function and patient schedules. Furthermore, the simulation model could be benchmarked with genetic and evolutionary algorithms to get more insight in the most optimal number of resources, given certain circumstances (like one of the what if scenarios).

The implementation of costs in this version of ABOCSS is very limited. For further evaluation, the costs per resource should be implemented in such a way that a trade off could be made between for example, the costs and the assets of planning an extra IC bed.
The different surgical specialisms and the different patient paths are another aspect for further work. In the current version of ABOCSS, only the CTS specialism with the HC and PACU variation of the patient path are considered, but as can be seen in Figure 4.8 on page 48, there are many more specialisms that make use of the ICU.

Dynamic OR scheduling is also a consideration for future work. This means that a scheduling agent schedules patients based on the capacity of OR’s and the availability of postoperative care and on the different waiting lists. Each medical specialism should get its own scheduling agent. In the results of the experiments, we have seen that the completion date tends to stay the same (around the 16th of November). This leaves about one and half months of time available for the admission of patients. To utilize the remaining time, the OR Scheduling agent should be extended.

Besides these comments on further work for the implementation of ABOCSS, the definition and execution of new scenarios is an interesting consideration. A possible addition to the scenario 2, changing the opening times of the PACU, would be the planning of IC beds as presented in scenario 1. This way, a trade off could be made between the costs of a prolonged opening duration of the PACU and the number of IC beds, and the better throughput. Furthermore, we recommend performing extra experiments for more in-depth results.
Bibliography


Appendix A

Process Analysis

The legend of the process models can be found in Appendix A.0.1. A description on how to read the models can be found on Appendix A.0.2.
Figure A.1: The main process flow of CTS patients
Figure A.2: The HC sub process
Figure A.3: The IC planning process
Figure A.4: The IC emergency process
A.0.2 Description of the process models

The models are described according to the HC postoperative care sub process (Figure A.2). These process models should be read as follows: At the start of the process, a patient must
be admitted to the HC ward. First a check whether a bed is available is performed. Now there are two options to take:

1. When a bed is available for the patient (the ”y” side is followed), the patient will go into the OR to get its surgery performed. Thereafter, the patient will be admitted to a HC bed and the subprocess ends.

2. When there is no bed available for the patient (i.e. all beds are occupied by other patients), the medical staff will admit the patient to a higher postoperative care level (MC or IC). Two options again can happen:

   a. An IC or MC bed is available and the patient can be admitted. The process continues with [1]

   b. All IC and MC beds are occupied. Then the following happens:

      i. It is up to the creativity of the ICU planner to find a solution. If the ICU planner finds a solution, [1] is followed again. If the ICU planner does not succeed in finding a bed, the operation is postponed until a bed can be made available.
Appendix B

Data Analysis

In this Chapter, several data sources are used. The different sources are indicated in the caption of the table or Figure as follows:

1. EZIS
2. ICIS
3. Cardio2000
4. Admission Office Data
Table B.1: Surgical Interventions Legend with number of patients per category (2005)

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Dutch meaning</th>
<th>English meaning</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>Algemene Chirurgie</td>
<td>General Surgery</td>
<td>6351</td>
<td>16,11%</td>
</tr>
<tr>
<td>CAR</td>
<td>Cardiologie</td>
<td>Cardiology</td>
<td>102</td>
<td>0,26%</td>
</tr>
<tr>
<td>CTC</td>
<td>Cardio-Thoarocale Chirurgie</td>
<td>thoracic surgery</td>
<td>3245</td>
<td>8,23%</td>
</tr>
<tr>
<td>DER</td>
<td>Dermatologie</td>
<td>Skin Surgery</td>
<td>853</td>
<td>2,16%</td>
</tr>
<tr>
<td>GYN</td>
<td>Gynaecologie</td>
<td>Gynaecology</td>
<td>1905</td>
<td>4,83%</td>
</tr>
<tr>
<td>KAA</td>
<td>Kaakchirurgie</td>
<td>Jaw Surgery</td>
<td>685</td>
<td>1,74%</td>
</tr>
<tr>
<td>KIN</td>
<td>Kinder chirurgie</td>
<td>Paediatrics</td>
<td>40</td>
<td>0,10%</td>
</tr>
<tr>
<td>KNO</td>
<td>Keel- neus- en oor chirurgie</td>
<td>oto rhino laryngology</td>
<td>2156</td>
<td>5,47%</td>
</tr>
<tr>
<td>LON</td>
<td>Long chirurgie</td>
<td>respiratory surgery</td>
<td>17</td>
<td>0,04%</td>
</tr>
<tr>
<td>NCH</td>
<td>Neurochirurgie</td>
<td>Neurosurgery</td>
<td>95</td>
<td>0,24%</td>
</tr>
<tr>
<td>OOG</td>
<td>Oog chirurgie</td>
<td>Ophthalmology</td>
<td>2876</td>
<td>7,29%</td>
</tr>
<tr>
<td>ORT</td>
<td>Orthopedie</td>
<td>Orthopaedics</td>
<td>4264</td>
<td>10,81%</td>
</tr>
<tr>
<td>PCH</td>
<td>Plastische chirurgie</td>
<td>Plastic surgery</td>
<td>2753</td>
<td>6,98%</td>
</tr>
<tr>
<td>URO</td>
<td>Urologisch chirurgie</td>
<td>Urological surgery</td>
<td>14085</td>
<td>35,72%</td>
</tr>
</tbody>
</table>

Table B.2: Number of patients per category for the OR (1998 - 2005)

<table>
<thead>
<tr>
<th>Category</th>
<th>Priority</th>
<th>1998</th>
<th>1999</th>
<th>2000</th>
<th>2001</th>
<th>2002</th>
<th>2003</th>
<th>2004</th>
<th>2005</th>
</tr>
</thead>
<tbody>
<tr>
<td>Waiting list</td>
<td>1 and 2</td>
<td>50%</td>
<td>54%</td>
<td>54%</td>
<td>55%</td>
<td>55%</td>
<td>56%</td>
<td>55%</td>
<td>54%</td>
</tr>
<tr>
<td>Urgent</td>
<td>3</td>
<td>43%</td>
<td>40%</td>
<td>40%</td>
<td>39%</td>
<td>39%</td>
<td>40%</td>
<td>40%</td>
<td>41%</td>
</tr>
<tr>
<td>Emergency</td>
<td>4</td>
<td>7%</td>
<td>6%</td>
<td>6%</td>
<td>6%</td>
<td>6%</td>
<td>4%</td>
<td>5%</td>
<td>5%</td>
</tr>
</tbody>
</table>
Table B.3: Number of patients per category for the IC (2002 - 2006) [2]

<table>
<thead>
<tr>
<th>Category</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTS</td>
<td>7474</td>
<td>57,96%</td>
</tr>
<tr>
<td>respiratory surgery</td>
<td>222</td>
<td>1,72%</td>
</tr>
<tr>
<td>general surgery</td>
<td>2189</td>
<td>16,98%</td>
</tr>
<tr>
<td>orthopaedics</td>
<td>211</td>
<td>1,64%</td>
</tr>
<tr>
<td>traumatology</td>
<td>101</td>
<td>0,78%</td>
</tr>
<tr>
<td>urological surgery</td>
<td>123</td>
<td>0,95%</td>
</tr>
<tr>
<td>gynaecology</td>
<td>473</td>
<td>3,67%</td>
</tr>
<tr>
<td>neurosurgery</td>
<td>1</td>
<td>0,01%</td>
</tr>
<tr>
<td>other surgery</td>
<td>131</td>
<td>1,02%</td>
</tr>
<tr>
<td>cardiology</td>
<td>331</td>
<td>2,57%</td>
</tr>
<tr>
<td>respiratory</td>
<td>608</td>
<td>4,72%</td>
</tr>
<tr>
<td>digestive</td>
<td>242</td>
<td>1,88%</td>
</tr>
<tr>
<td>renal</td>
<td>182</td>
<td>1,41%</td>
</tr>
<tr>
<td>CNS</td>
<td>182</td>
<td>1,41%</td>
</tr>
<tr>
<td>endocrine</td>
<td>98</td>
<td>0,76%</td>
</tr>
<tr>
<td>haematological</td>
<td>11</td>
<td>0,09%</td>
</tr>
<tr>
<td>skin/muscle/bone</td>
<td>14</td>
<td>0,11%</td>
</tr>
<tr>
<td>intoxication</td>
<td>238</td>
<td>1,85%</td>
</tr>
<tr>
<td>other acute</td>
<td>54</td>
<td>0,42%</td>
</tr>
</tbody>
</table>

Table B.4: Number of CTS patients per variant of the patient path. (2002 - 2006) [2]

<table>
<thead>
<tr>
<th>Category</th>
<th>Number</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>HC</td>
<td>4428</td>
<td>59,25%</td>
</tr>
<tr>
<td>PACU</td>
<td>3029</td>
<td>40,53%</td>
</tr>
<tr>
<td>Recovery</td>
<td>17</td>
<td>0,22%</td>
</tr>
</tbody>
</table>
Table B.5: LoS in days of PACU patients staying at the IC

<table>
<thead>
<tr>
<th>Mean</th>
<th>Standard Deviation</th>
<th>Maximum</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.02</td>
<td>2.38</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>

Table B.6: LoS in days of PACU patients staying at the MC

<table>
<thead>
<tr>
<th>Mean</th>
<th>Standard Deviation</th>
<th>Maximum</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.02</td>
<td>1.605</td>
<td>9</td>
<td>0</td>
</tr>
</tbody>
</table>

Table B.7: LoS in days of HC patients staying at the IC

<table>
<thead>
<tr>
<th>Mean</th>
<th>Standard Deviation</th>
<th>Maximum</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.75</td>
<td>2.38</td>
<td>8</td>
<td>0</td>
</tr>
</tbody>
</table>

Table B.8: LoS in days of HC patients staying at the MC

<table>
<thead>
<tr>
<th>Mean</th>
<th>Standard Deviation</th>
<th>Maximum</th>
<th>Minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.43</td>
<td>0.92</td>
<td>7</td>
<td>0</td>
</tr>
</tbody>
</table>
Figure B.1: OR duration in minutes per specialism [1]
Figure B.2: Scatterplot of the relation between the LoS at the ICU and the OR duration of 2005 data [1]
Figure B.3: Mined process model of 2005 and 2006 data. [4]
The causal relations of the \( \alpha \)-algorithm are defined as follows: Let \( W \) be an event log over \( T \), i.e., \( W \subseteq T^* \). Let \( a, b \in T \):

1. \( a >_W b \) iff there is a trace \( \sigma = t_1 t_2 t_3 \ldots t_n \) and \( i \in \{1, \ldots, n-1\} \) such that \( \sigma \in W \) and \( t_i = a \) and \( t_{i+1} = b \),
2. \( a \rightarrow_W b \) iff \( a >_W b \) and \( b \nRightarrow_W a \),
3. \( a \#_W b \) iff \( a \nRightarrow_W b \) and \( b \nRightarrow_W a \), and
4. \( a \|_W b \) iff \( a >_W b \) and \( b >_W a \).

Let \( W \) be a workflow log over \( T \) then \( \alpha(W) \) is defined as follows:

1. \( T_W = \{ t \in T \mid \exists _{\sigma \in W} t \in \sigma \} \),
2. \( T_I = \{ t \in T \mid \exists _{\sigma \in W} t = \text{first}(\sigma) \} \),
3. \( T_O = \{ t \in T \mid \exists _{\sigma \in W} t = \text{last}(\sigma) \} \),
4. \( X_W = \{ (A, B) \mid A \subseteq T_W \land B \subseteq T_W \land \forall a \in A \forall b \in B a \rightarrow_W b \land \forall a_1, a_2 \in A a_1 \#_W a_2 \land \forall b_1, b_2 \in B b_1 \#_W b_2 \} \),
5. \( Y_W = \{ (A, B) \in X_W \mid \forall (A', B') \in X_W A \subseteq A' \land B \subseteq B' \Rightarrow (A, B) = (A', B') \} \),
6. \( P_W = \{ p_{(A, B)} \mid (A, B) \in Y_W \} \cup \{ i_W, o_W \} \),
7. \( F_W = \{ (a, p_{(A, B)}) \mid (A, B) \in Y_W \land a \in A \} \cup \{ (p_{(A, B)}, b) \mid (A, B) \in Y_W \land b \in B \} \cup \{(i_W, t) \mid t \in T_I \} \cup \{(t, o_W) \mid t \in T_O \} \), and
8. \( \alpha(W) = (P_W, T_W, F_W) \).
Appendix C

Simulation Design
Figure C.1: The negotiation model of ABOCSS
D.1 Preface

This user guide is intended for the ABOCSS Simulation runner v1.0 that was developed as a master project at the Eindhoven University of Technology. The ABOCSS Simulation Runner is a simulation system, intended to simulate the planning of the CTS patient processes. In the remainder of this guide, the simulation parameters and the output of the simulation runs are described.

D.2 Prerequisites

The ABOCSS Simulation Runner is an application build in Java. To run Java applications, the Java Runtime Environment (JRE) must be installed on the system. If this is not the case, the ABOCSS Simulation Runner will not start. Get JRE 1.5 from http://www.java.com. The ABOCSS Simulation Runner relies on RMI as a communication protocol. Therefore, the RMIRegistry must be started before a simulation run is executed. You can find the RMIRegistry at the bin directory of the JRE installation folder.

D.3 The Menu

When the ABOCSS Simulation Runner is started, a main window is shown on the screen like in Figure D.1 From that, you have the following menu options:

1. File
   (a) Quit - Quit the ABOCSS Simulation Runner.

2. Setup
   (a) Paths - Define the folders where configuration files are stored.
   (b) Simulation - Set up a the parameters for a simulation run.
   (c) Costs - Set up costs for a simulation run.
3. Resources

(a) Setup - Set up the number of resources per agent.

(b) Probabilities - Set the probabilities for IC and MC resources.

4. Simulation

(a) Run - Run a simulation.

(b) Results - Show the results of a simulation.

5. About
D.3.1 Setup

If Setup is selected, one of the three items listed below this menu can be picked. The first one is Paths. In this screen, three paths can be specified that are used for the configuration of a simulation run. The first path is to specify the config file. This config file is an xml file that contains network information like ports and ip-addresses. The default configuration file, which is provided with the installation, can be found at:

```
\config\parameters.xml
```
of the installation folder. The next option is the location of the \textit{patient profiles} file. In this file, the variations of the patient path are specified. The default \textit{patient profile} file can be found at

\texttt{.\config\PatientProfiles.xml}

of the installation folder. The third option is the \textit{waiting list} file. This file contains patients that are waiting for surgery. This file is generated when a new simulation run is started (except when specified otherwise). The \textit{waiting list} can be found at

\texttt{.\config\waitinglist.xml}

of the installation folder. The second menu item in the \textit{setup} menu is \textit{simulation}. In this screen, simulation settings like opening days and times and the number of runs can be specified. A special box is the \textit{number of patients needed to complete the simulation run}. When this box is left empty, the simulation is run for one year, otherwise the number of patients is admitted by the OR Scheduling agent. The verbose run box implies that all messages communicated from and to agents are shown in the console. The checkbox labeled \textit{Generate new waitinglist} enables the generation of a new waiting list, each time a new simulation run is started. The generation of a new waiting list can take some time. When this box is not checked, the same waiting list is used for every run. This is only recommended for test purposes and not for actual evaluation.

The last item in the \textit{setup} menu is \textit{costs}. Here you can specify resource costs per hour for each care level. These costs are calculated while the simulation runs and presented in the output files.

\subsection*{D.3.2 Resources}

(Figure [D.3])

The first menu item in the \textit{resources} menu is \textit{setup}. Here, the number of resources per agent can be set. Besides this, the probability that a certain number of beds will be
available for CTS at the IC or the MC can be specified. The last setting is done in the
probabilities item of the resources menu.

D.3.3 Simulation

(Figure D.4)

In this menu item, only one choice is available, namely run. This enables the ABOCSS
Simulation Runner to start a simulation run. As can be seen when selecting this menu
item, the setup and the resources menu are disabled in such a way that these settings
cannot be adjusted any more. When one of these settings has to be changed, the ABOCSS
D.3. The Menu

The menu simulation runner has to be restarted. When the run item is selected, a screen like the one in Figure D.4 is shown.

![Figure D.4: ABOCSS Simulation Runner Simulation Screen](image)

D.3.4 Waiting List

(Figure D.5)

In this input screen, you can specify the percentage of PACU patients for a simulation run.
D.3.5 Running simulations

When the run button in the simulation runner is selected, ABOCSS will first generate a patient list (unless the system is instructed not to generate a new waiting list). This can take some time. When the waiting list is generated, a simulation run is started. In the simulation runner screen, the different agents can be seen while performing actions. The beds are marked with colors. White means an empty bed, yellow means that a patient belonging to the HC variant of the patient path is occupying a bed, green means that a patient belonging to the PACU variant of the patient path is occupying a bed and red means that a bed is unavailable. All agents show the beds with the LoS for a patient staying in it,
except for the OR Scheduling agent. The OR Scheduling agent shows the patient id on a resource. The surgical ward agent shows the LoS per resource, except for patients being admitted before surgery, then the patient id is lead by an A.

The current day and time are shown in the upper right corner. The number of PACU and HC patients on the list are shown in the upper left corner. The communication between agents can be followed through the communication lines, that appear when an agent sends a proposal or response. The ABOCSS simulation runner in action can be seen in Figure D.6.

Figure D.6: ABOCSS Simulation Runner running a simulation
D.3.6 Results

When the execution of a scenario is completed, the result item becomes available in the simulation menu. When clicked, a screen like the one in Figure D.7 shows up. Here one can see the result of the simulation run: the number of admitted patients per agent together with the number bounced and rejected patients. The costs can be viewed in the bottom of the screen.

The patient log can be found in the log directory of the installation. This directory holds three files per simulation run. Each file is marked with the date and time of the start of the simulation run. The first file in the log directory is called log-dateandtime-.csv. This is a csv file that can be read with a spreadsheet program. Each entry in the log represents a transaction of an agent. The next file in the log directory is called log-dateandtime-.txt. This file gives a summary of a simulation run and contains all the patient id’s that are admitted, rejected and bounced by an agent. The last file in the log directory is the logsummary-dateandtime-.csv. This file is a summary of the log-dateandtime-.txt file. The difference between the log-dateandtime-.txt file and the logsummary-dateandtime-.csv file is that the logsummary-dateandtime-.csv file holds the total number of admitted, rejected and bounced patients in a file format that can be processed with a spreadsheet or database application.

D.3.7 Processing results

When the execution of a scenario is completed, the results can be processes. In this research, we used Microsoft Access 2003 to process the results. This is done in the following way:

1. Create a blank database.

2. Select file then Get external data finally import.
3. Select text files (*.txt; *.csv; *.tab; *.asc) in the files of type drop down list.

4. Select the first logsummary-dateandtime-.csv file in the logs directory.

5. Now a wizard is shown. Follow the steps through (leave the default settings) until step 4, the field options is reached. Enter the following information:

   (a) rename the first column to agent;

   (b) rename the second column to admitted;

   (c) rename the third column to rejected;

   (d) rename the fourth column to bounced.
6. In the next step, select *No primary key*.

7. Give the table a name in the final step.

For all other output files, related to this simulation run, follow the same procedure, besides the third step of the wizard. Since the table is already been created, select *In an existing table* and choose the table name from the drop down box. When all files for a scenario are imported in a table, Microsoft Access can be used to query and analyze the data.