Towards leveraging surgery clusters in hospital operating room scheduling

Jochems, Jeff Antonius Johannes Louis

Award date:
2021

TU/e EINDHOVEN
UNIVERSITY OF
TECHNOLOGY

Department of Mathematics and Computer Science
Process Science Charter

Towards leveraging surgery clusters in hospital operating room scheduling

Master Thesis

Jeff Antonius Johannes Louis Jochems

Supervisors:
dr. D. Fahland
dr. S. Hess
dr. D. Roubos
R. Quanjel

Eindhoven, July 2021

Abstract

Efficient operating room scheduling is crucial in order to provide an optimal quality of care in hospitals. We investigate if operational hospital operating room scheduling can be improved by clustering the surgery load. Previous research has developed successful scheduling methods and grouped surgeries to assign operating room capacity to surgical departments but failed to address the potential of surgery clusters in operational surgery scheduling. We developed a scheduling framework leveraging surgery clusters based on their scheduling characteristics. A case study of two dutch surgical hospital departments evaluates this scheduling method. Our research shows cluster-based scheduling can improve the operational operating room schedule and provides an extensible framework for future development.

Contents

Chapter 1

Introduction

Efficient surgery scheduling is important for hospitals trying to provide high-quality patient care and optimize revenue. This thesis studies how patient and surgery characteristics can be used to improve scheduling, leveraging clusters of surgeries. This chapter introduces the research field of surgery scheduling, outlines the research performed for this thesis and, summarizes its findings. Firstly, we explain operating room scheduling and the context of this thesis in section 1.1. The state of the art of this field and the hospitals and departments involved with this research are introduced in section 1.2. Section 1.3 formulates the specific research goals addressed in this thesis and section 1.4 discusses the research approach. Finally, Section 1.5 provides a digest of the obtained results and findings.

1.1 Context

Healthcare is a crucial societal industry, providing medical and care services aiming to result in the best possible health outcome for the population. With a total expenditure of € 80.9 billion in the Netherlands in 2019 it accounts for 10 % of the Dutch gross domestic product [1]. The size and the social importance of this sector are expected to increase even further as a result of the expected increase in healthcare demand associated with an ageing population [2].

Most of these care services are facilitated by hospitals and a major part of this is centered around surgical care. In fact, the operating room (OR) is responsible for the largest portion of hospital revenue, and its efficiency impacts the entire hospital performance [3]. Efficient OR scheduling ensures that as many patients as possible receive adequate care. However, OR management is complicated by scarce resources and the highly interrelated nature of the surgical suite to other hospital facilities. Hence, improving OR planning, simultaneously improves patient care and hospital revenue.

Currently, in many hospitals staff schedules surgeries manually, assigning patients from a priority queue to an operational OR schedule. However, OR scheduling is an established field of study in operations research with the potential of improving operating room efficiency [3]. Additionally, machine learning techniques have been successfully applied to more accurately predict surgery scheduling characteristics [4]. Predicting these surgery characteristics based on patient and surgery characteristics available before scheduling can improve OR scheduling performance. In this thesis, we develop an operating room scheduling method that estimates surgery characteristics based on surgery clusters. First, we explain the context and scope of OR scheduling considered in this thesis. Next, we discuss how surgery clusters can be used to improve OR scheduling.

Operating room scheduling

The operating room scheduling interacts with multiple aspects of hospital care. Hence, effective OR scheduling manages the demand it receives from and submits to other hospital departments. Both the outpatient clinic, clinical wards, and emergency room determine the surgical demand. Patients are prepared for surgery in clinical, outpatient, and radiology departments. Furthermore, patients are admitted to clinical wards after surgery. Managing the effect of the OR schedule on hospital performance is a multidimensional problem and influences: patient care (patient waiting times and postponements), hospital efficiency and revenue, and staff stress (overtime, uncertain schedules).

Multiple levels of OR planning are distinguished in hospital management, each with a different effect on other hospital departments [5]:

1. *Strategic OR scheduling* deals with the long-term responsibility and resource assignment to surgical departments. This level determines which and how many surgeries a department is supposed to perform e.g. on a yearly basis.
2. *Tactical OR scheduling*, however, determines when resources should be available to surgical departments to meet the load prescribed by the strategic level. At this level OR scheduling typically attempts to manage the required postoperative capacity of wards by assigning OR time slots to surgical departments.
3. *Operational OR scheduling* involves the departmental decisions allocating staff and surgeries to the ORs and time slots made available to one surgical department by the tactical level.

This thesis focuses on operational scheduling. We consider the load coming from outpatient, clinical and emergency wards as fixed and focus on managing OR scheduling efficiency and post-operative patient flow.

Leveraging surgery cluster characteristics

To adequately schedule surgeries, the scheduling characteristics of these surgeries have to be available. Currently, when scheduling the OR, hospitals often assume the duration for every surgery type is fixed. However, this duration can depend on various factors, including for the performing surgeon and patient health and age. Recent research shows machine learning methods can help to predict the surgery duration more accurately [6]. Moreover, groups of different surgeries exist with similar scheduling characteristics [7]. Using clustering and classification we aim to show predicted surgery characteristics can improve operational surgery scheduling.

Managing uncertainty is a primary challenge in OR scheduling. Providing more accurate scheduling characteristics improves operation OR scheduling by reducing this uncertainty. Additionally, when the future demand of surgeries is available, this can be taken into account during scheduling as well. Dynamical OR scheduling methods can leverage the expected surgical demand to combine surgeries with compatible characteristics. However, since every surgery has different characteristics, forecasting the characteristics of future surgery demand is often infeasible. By clustering the surgeries performed by a surgical department, future OR scheduling can be improved in two ways. Firstly, surgery clusters allow us to estimate surgery characteristics during scheduling. Secondly, by clustering the surgery population we can summarize the surgery demand to be used in future OR scheduling solutions.

1.1.1 Case Hospitals

The research described in this thesis is performed in collaboration with two case hospitals. These hospitals provide OR scheduling data which we use to evaluate the developed cluster-based scheduling method. Furthermore, the proposed clustering and scheduling method designs require case-specific domain knowledge. To identify case-specific OR scheduling goals and challenges, we interviewed and surveyed hospital stakeholders. Furthermore, we jointly designed and evaluated the surgery clusters and scheduling performance measures. Chapter 3 explains the general research problem more thoroughly and discusses the domain knowledge obtained for both hospital cases in detail. The two investigated case studies are:

1. **Orthopedic surgery at Franciscus Gasthuis & Vlietland:** This department performs surgeries to treat conditions to the musculoskeletal system. Franciscus Gasthuis & Vlietland is a peripheral hospital in Rotterdam. Hospital stakeholders consider the OR schedule used in this department to be efficient.

2. **Cardiothoracic surgery at Maastricht UMC+:** This department performs surgeries to the heart and lungs. Maastricht UMC+ is the academic hospital in Maastricht. Due to the complex and specific nature of the surgeries performed by this department, hospital stakeholders experience a challenging OR scheduling case resulting in frequent last-minute postponements.

1.2 State of the Art

Samudara et al. (2016) provide a comprehensive literature review on *OR* scheduling outlining this topic of study in the academic field of operations research [8]. The field of operational OR scheduling has developed scheduling solutions for a wide range of surgery scheduling settings. This thesis focuses on the allocation of patients to a time and OR in a set of available time slots (block schedules) predetermined at the tactical level. Within this scope, some research focuses on the efficient scheduling of elective (either inpatient [9] or outpatient [10]) surgeries while other studies deal with non-elective surgeries such as emergencies. Scheduling solutions typically optimize specific combinations of performance criteria, often investigating a trade-off between performance measures. Frequently investigated performance criteria include: waiting time (for patient or surgeon), utilization, OR idle time, throughput, and surgery postponements [3]. Some of these solutions also take the capacity in postoperative facilities into account [11].

We distinguish the methodologies proposed for OR scheduling into two categories, static and dynamical scheduling. *Static OR scheduling* employs a scheduling horizon and optimizes some scheduling performance by scheduling a set of surgeries (often from a queue) at the same time, whereas *dynamic scheduling* aims to allocate a single surgery to a (partially filled) schedule. Examples of methodologies successful in static scheduling include *mathematical programming* (MP) [9] [12] and *genetic algorithms* [13]. The extensively studied static approach assumes that all surgeries which are to be scheduled in the considered planning horizon are known in advance. In contrast to static scheduling, dynamical scheduling surgeries allow the expected future surgeries to be taken into account during scheduling as well. Delaying specific surgeries in favour of surgeries with more compatible scheduling characteristics can improve both the currently scheduled and future schedules [14]. Dynamic scheduling is an established method in general appointment scheduling but, due to the large variability in surgery durations, there is a gap in research translating such methodology from the appointment to the surgery scheduling setting [8].

Regardless of the scheduling method used, the allocation of surgeries requires an *estimation of surgery scheduling characteristics* such as surgery durations. Traditionally, hospitals schedule a specific surgery with a duration estimation based on the average duration of surgeries of the same type. Recently, however, machine learning models have been used to more accurately estimate surgery durations based on the wealth of patient and surgery data available for surgeries in the hospital [6, 15]. In fact, Strömblad et al. (2021) show that implementing machine learning model predictions in a clinical scheduling setting reduces the surgery delays, and on-site patient waiting time [4]. This previous research, however, seems to focus on predicting surgery durations specifically. As surgery scheduling affects the demand of related facilities, improving the estimations of other scheduling characteristics, like the postoperative length of stay, can further improve scheduling performance.

Despite the wealth of operational OR scheduling research in academia, its adoption is lacking. In fact, both the hospital cases studied in this thesis employ no specific operational OR scheduling methodology. Both schedule surgeries manually from a priority queue, on a first come first serve basis. However, the tactical level in both hospital cases is optimized to manage postoperative departmental load. Samudara et al. (2016) attribute this lack of adoption to the fact that research rarely targets scheduling practitioners, fails to use practically relevant performance measures, or do not clearly report the setting and method assumptions [8].

1.3 Research objective

This thesis addresses the gap of research investigating the usage of multiple simultaneously estimated scheduling characteristics in operational OR scheduling. Furthermore, we develop a scheduling framework that is adoptable in practice.

This research aims to show *how surgery clustering, based on scheduling characteristics, can be used in operational operating room scheduling.* Specifically, we address how the clustering of the surgeries of the two case hospitals can be used to improve their operational operating room scheduling. To generalize the results of the two case hospital we aim to develop a method that can be adapted to new surgery scheduling cases. This method should improve the operational operating room scheduling while maintaining schedule feasibility by adhering to the case-specific scheduling constraints. By clustering surgeries into groups of surgeries with similar scheduling characteristics, we divide the population of surgeries into groups with scheduling characteristic distributions and historical arrival patterns. Future research could use these arrival patterns as an expected future demand. Section 3.5 divides this research goal into specific sub-tasks.

1.4 Method

To show surgery clusters can be used to improve the hospital operating room scheduling, we develop a scheduling method that estimates the scheduling characteristics of a surgery from surgery clusters found in historic data. This method first clusters historical surgeries based on their scheduling characteristics. Next, this method uses a classification model to predict the surgery cluster to which a new surgery that is being scheduled belongs. We then use the cluster characteristics to estimate the surgery scheduling characteristics. Finally, heuristic scheduling strategies use the estimated surgery duration to schedule the surgeries. We evaluate the proposed scheduling method by identifying the most suitable heuristic for each hospital case and comparing this to the surgery schedules developed by the case hospitals. An overview of this method is provided in chapter 4.

As our approach leverages several data mining methodologies we base our project approach on the *cross-industry standard process for data mining (crisp-DM)* [16]. Chapter 3 explains the business, data understanding, and data preparation obtained in the corresponding *crisp-DM* steps. Next, we give an overview of the necessary modeling and evaluation steps in Chapter 4. The details of these steps are explained further in the subsequent chapters.

1.5 Findings

The developed scheduling method successfully improves the hospital scheduling of the investigated case hospitals and is applicable in practice. Section 1.5.1 summarizes the obtained results and Section 1.5.2 contains an abstract of the conclusion following from these results. Chapter 9 evaluates the results in more detail.

1.5.1 Results

We used agglomerative clustering to distinguish 45 orthopedic surgery clusters and 23 cardiothoracic surgery clusters with distinct scheduling characteristics. Of these clusters, 27 and 11 were used to estimate surgery characteristics. To estimate these characteristics, we trained a classification model with an *accuracy* and *initial cluster accuracy* of 0.96 and 0.99 for orthopedic surgeries. The cardiothoracic surgery cluster classification obtained an *accuracy* of 0.90 and *initial* cluster accuracy of 0.98. This resulted in a scheduling characteristic estimation with a *duration mean absolute error (MAE)* of 20.91 minutes for orthopedic surgeries and 59.08 minutes for cardiothoracic surgeries, a *postoperative length of stay (MAE)* of 28.40 hours for orthopedic surgeries, and 101.38 hours for cardiothoracic surgeries and *postoperative department accuracy* of 0.98 for orthopedic and 0.91 for cardiothoracic surgeries.

When investigating several heuristic strategies we found *shortest processing time first (SPF)* scheduling to be most suitable for orthopedic surgery scheduling. Using this strategy for orthopedic surgery scheduling, we improved the performance of the hospital surgery schedules. The average *utilization* and *undertime* of the simulated method schedules increased by 0.34 and 36 hours, whereas the average *idle time*, *overtime*, on-site *patient waiting time* and number of *postponements* decreased by 57.1 hours, 35.1 hours, 263.4 hours, and 28 surgeries per two weeks compared to the simulated hospital schedules.

Similarly, we found using the *longest processing time first, Bailey-Welch (LPF-BW)* or *largest variance first, Bailey-Welch (LVF-BW)* scheduling results in the most suitable cardiothoracic surgery schedules. The simulated schedules resulting from these strategies could not be compared to simulated hospital schedules but outperform the realized hospital schedules. The average *utilization* and *undertime* of the schedules resulting from the simulated scheduling method increased by 0.14 and 6.3 hours, whereas the average *idle time*, *overtime* and number of *postponements* decreased by 24.1 hours, 20 hours, and 1 surgery per two weeks compared to the realized hospital schedules.

1.5.2 Interpretation

We successfully identified surgeries with distinct scheduling features. However, the clustering method resulted in several small clusters. The classification models trained to predict these clusters for surgeries that are to be scheduled do so accurately enough to develop effective scheduling strategies. Unfortunately, the *initial cluster accuracy* of the classification models for both orthopedic and cardiothoracic scheduling case are not perfect. The *initial cluster accuracy* denotes how well the classification method distinguishes clusters with different scheduling constraints and thus ideally would be 1. For the two studied hospital cases, however, the confused initial clusters do not result in infeasible schedules, so this classification can be used in our scheduling method.

For orthopedic OR scheduling, our scheduling method, using *SPF* scheduling, results in schedules outperforming the schedules developed by the hospital. Similarly, we observe an improved scheduling performance when using the *LPF-BW* or *LVF-BW* scheduling strategies for cardiothoracic OR scheduling. Additionally, we showed how to apply this scheduling method in a practical setting. We can optionally extend this method to manage emergency surgeries and improve the surgery scheduling further by separately predicting surgery durations. Furthermore, we demonstrated how to explain cluster predictions made by the classification model to, for example, scheduling staff.

This thesis presents an extensible framework able to efficiently schedule surgeries based on cluster scheduling characteristics derived from surgery clusters. We showed this framework can be used to improve the OR scheduling of two hospital cases. While the individual models developed for both hospital cases are not ready to be applied clinically, the framework itself can be used in practice. Future research may extend this framework to leverage additional scheduling characteristics to manage for example postoperative patient flow.

Chapter 2

Preliminaries

This thesis studies a scheduling method for scheduling patients based on scheduling characteristics estimated from surgery clusters. This chapter explains the methodology background and provides a general introduction to the used techniques. Section 2.1 explains clustering and the specific method used to cluster surgeries based on surgery characteristics. Next, Section 2.2 discusses classification and the method used to predict the cluster for surgeries during scheduling. Section 2.3 discusses a similar method used to extend the scheduling method with individually predicted surgery clusters. Finally, Section 2.5 explains the method used to compute feature contribution explanations for the classification of surgeries into clusters.

2.1 Clustering

Clustering is, in the context of data mining, the task of discovering groups in data. It is an unsupervised machine learning method and aims to group similar data. We assume a basic familiarity with the concept of clustering as an unsupervised machine learning method and refer to Tan et al. (2016) for a more thorough explanation [17]. The method used to cluster surgeries in this thesis is agglomerative clustering and uses the heterogeneous distance measure: Gower distance. Section 2.1.1 explains this distance measure and Section 2.1.2 explains the agglomerative clustering approach. Section 2.1.3 explains the silhouette method and how it can be used to determine a suitable number of clusters.

2.1.1 Gower distance

Clustering algorithms group data based on similarity. In general, cluster analysis aims to find groups in which data is as similar as possible, while the data from other groups is as dissimilar as possible. In mathematics, several distance measures exist, which express this notion of similarity between data points. In Euclidean space, the similarity between two observations is frequently expressed using Euclidean distance.

However, this assumes all observation features are numerical and in the Euclidean space. In the surgical clustering case, this does not hold as we also cluster surgeries based on categorical features such as *department after surgery*. Gower (1971) proposes a distance measure that is able to express the distance between observations with both numerical and categorical features [18]. To express the distance between the observations $x_i = \{x_{i1}, \ldots, x_{in}\}$ and $x_j = \{x_{j1}, \ldots, x_{jn}\}$ with n features, this distance measure leverages a similarity score s_{ijk} for every feature $1 \leq k \leq n$. Depending on the nature of feature k, this similarity denotes something different:

1. If feature k is numerical s_{ijk} is the scaled difference between the observations in feature k:

$$s_{ijk} = 1 - \frac{|x_i - x_j|}{R_k}, \tag{2.1}$$

where R_k denotes the range of feature k.

2. If feature k is categorical, s_{ijk} denotes whether or not x_i and x_j represent the same category:

$$s_{ijk} = \mathbb{1}\{x_i = x_j\}. \tag{2.2}$$

Furthermore, Gower distance uses the variable δ_{ijk} denoting whether or not the feature k can be compared for observations x_i and x_j. E.g. if feature k is missing for one of the observations, this is not the case and $\delta_{ijk} = 0$, otherwise $\delta_{ijk} = 1$. Combining the score s_{ijk} and variable δ_{ijk}, Gower distance between two observations x_i and x_j is defined as the average of the similarity scores of all n features:

$$d_{gow}(x_i, x_j) = S_{ij} = \frac{\sum_{k=1}^{n} s_{ijk}\delta_{ijk}}{\sum_{k=1}^{n} \delta_{ijk}}. \tag{2.3}$$

2.1.2 Agglomerative clustering

We use the Gower distance measure explained in the previous section to cluster patients using agglomerative clustering. This clustering method is hierarchical in nature and recursively groups data until the desired similarity level is reached [19].

Agglomerative clustering initially considers each data point to belong to a specific cluster. Next, this clustering method iteratively merges the two most similar clusters until the desired number of clusters is found. To decide which clusters to merge, agglomerative clustering uses a linkage criterion and a distance measure. As discussed in Section 6.3, the method used to cluster surgeries employs the average linkage criterion with Gower distance. Average linkage considers the distance between two clusters to be the average distance of all pair-wise distances of all observations in one cluster to all observations of the other cluster.

By clustering with agglomerative clustering, we obtain a partitioning of observations represented by a dendrogram. This dendrogram contains the nested similarity of all observations in the data. By cutting the dendrogram at a specific level, we obtain the clustering with a desired number of clusters. For example, to obtain a clustering with 3 clusters, we cut the dendrogram before the last two merges.

2.1.3 Silhouette method

Section 6.5 discusses how we use the silhouette method to determine the number of clusters per surgery category or initial cluster. We use the number of clusters resulting in the highest silhouette score. The silhouette score is the average silhouette value of all clustered observations. This silhouette value measures how much an observation resembles other observations within that cluster compared to observations in other clusters. Hence, the average silhouette value expresses how closely observations are grouped in the investigated clusters and thus is a measure of clustering performance.

Rousseeuw (1987) proposes the silhouette value $s(x_i)$ of observation x_i in cluster C_i, such that $x_i \in C_i$, as a measure depending on the dissimilarity of that observation to observations in its own cluster $a(x_i)$ and its dissimilarity to observations in other clusters $b(x_i)$ [20]. Again the similarity between observations is expressed with a distance measure $d(x_i, x_j)$, for which we use the Gower distance explained in Section 2.1.1 in our clustering method. The dissimilarity of observation x_i to the other observations in cluster C_i is defined as the average distance between x_i and the cluster observations, such that:

$$a(x_i) = \frac{1}{|C_i| - 1} \sum_{x_j \in C_i, x_i \neq x_j} d(x_i, x_j). \tag{2.4}$$

Similarly, the dissimilarity of observation x_i to other observations outside cluster C_i is defined as the smallest average distance of x_i to the observations in any other cluster. Hence, this dissimilarity is defined as:

$$b(x_i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{x_j \in C_k} d(x_i, x_j). \tag{2.5}$$

These two dissimilarity measures combine to formulate the silhouette value:

$$s(x_i) = \begin{cases} 1 - \frac{a(x_i)}{b(x_i)} & \text{if } a(x_i) < b(x_i) \\ 0 & \text{if } a(x_i) = b(x_i) \\ \frac{a(x_i)}{b(x_i)} - 1 & \text{if } a(x_i) > b(x_i) \end{cases} \tag{2.6}$$

To determine how well the data is clustered, the silhouette score averages the silhouette value for all observations in the dataset. To determine a suitable number of clusters, we compute the silhouette score for multiple numbers of different clusters, after which we select the number of clusters resulting in the highest silhouette score.

2.2 Classification

In contrast to clustering, classification is a supervised machine learning task. It aims to predict a dependent categorical feature based on a set of independent features. Again, we assume a basic familiarity with the general concept of classification and refer to Tan et al. (2016) for a more thorough explanation [17]. In this thesis, we use Random forest classification to predict surgery clusters for surgeries which are being scheduled. The next section discusses this classification method

2.2.1 Random forest classification

The random forest classification method is an ensemble approach, aggregating results from multiple models (decision trees). Let us first discuss the idea behind decision trees and explain how these models can be combined to obtain an ensemble model with better generalization.

Classification trees, and classification models in general, map input features to a categorical output variable. These decision trees partition the feature space into increasingly smaller regions. In these new regions, the relationships between the smaller feature space and output variable might be more apparent. This is performed recursively until the final subspaces can be used to fit relatively simple models. The 'tree' structure in decision trees represents the recursive partitioning and its leaves correspond to a final partition in the partitioning, containing the simple model which only applies to this partition. The internal nodes of the decision tree represent conditions about the features that are used to define the partitioning. To evaluate the classification of a single input, we trace the classification tree to the leaf node containing the partition to which this input belongs. Next, we use the simple model fit to that leaf partition to classify the input. We traverse decision tree by traveling down the tree, evaluating the subsequent feature conditions in each tree node.

To determine a predicted output when the partition of a particular input instance is found, we evaluate the simple model associated with that partition. Classical classification trees implement the mode of the output variables belonging to partition samples as the predicted output. Consider the following partition of training features and associated output variables:
$\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$, the classic model for this partition would predict the output variable $\hat{y} = \text{argmax}_a |\{y \in \{y_1, \ldots, y_n\} | y = a\}|$.

Computing decision trees involves finding the right splitting rules or conditions on the input features and removing partitions (pruning) that increase overfitting. A popular method for the construction of decision trees is CART (Classification And Regression Trees) [21]. This method recursively splits the training data 'top down'. On each split, the independent variable that best splits the training data in terms of the dependant variable is used. For classification, this method requires a notion of homogeneity of the output class in each subset of the training data that can be used to measure the quality of a split. We use Gini impurity to measure how often a randomly chosen observation from one of the subsets resulting from a split would be incorrectly labeled if it received a random class label corresponding to the distribution of labels in that subset [22]. For C possible class labels in partition D, where $1 \leq i \leq C$ and $p(i)$ is the class probability of class i, the Gini impurity is formalized as:

$$G(D) = 1 - \sum_{i=1}^{C} p(i)^2 \tag{2.7}$$

To identify the best split, we chose the split resulting in the highest Gini gain. This gain is the difference between the original branch Gini impurity and Gini impurities of the branches resulting in from the split weighted by the number of observations in each split. Consider D to be the partition of the data before splitting. When splitting this partition in samples S_1 and S_2 based on feature F, the Gini gain is defined as:

$$\Delta G(F) = G(D) - (\frac{|S_1|}{|D|} G(S_1) + \frac{|S_2|}{|D|} G(S_2)). \tag{2.8}$$

Hence, the best split is the split resulting in the smallest weighted sum of Gini impurities of both resulting samples.

Decision trees by themselves are typically sensitive to the training data and do not generalize well to unseen data. This results in a model that suffers from high variance. Random forests, perhaps unsurprisingly, combine many decision trees to improve predictive power and generalization. This combining of models is called ensemble learning and results in an ensemble model. Random forest classification aggregates a large number of classification trees by taking the most frequently predicted result. Random forest aggregation of decision trees applies two techniques to construct a general predictive model [23]:

1. The method of combining multiple decision trees applied by random forest regression models is 'bagging', short for bootstrap aggregating. Bootstrapping is the resampling of the training data by random sampling with replacement from the training data. Bagging uses this technique to generate n number of bootstrapped datasets and learning a decision tree for every new dataset. These decision trees are then aggregated into the final ensemble model. Since bagging averages the results of many different decision trees it reduces the variance of the aggregated model.

2. To reduce the correlation between the aggregated trees, random forests consider only a subset of the features each time a split is generated. Every time a split is constructed, random forests draw a random sample of input features to consider for this split. Doing so helps prevent the model from overfitting.

2.3 Regression

Where classification aims to predict a dependent categorical feature based on a set of independent features, regression estimates the relationship between those features and a dependent numerical feature. Similarly to classification, regression is a basic machine learning task and we assume a basic familiarity with the general concept. For a more thorough explanation, we refer to Tan et al. (2016) [17]. The general scheduling method proposed in this thesis can be extended by separately predicting scheduling characteristics. To do so, we use random forest regression and Section 2.3.1 explains this regression model.

2.3.1 Random forest regression

Random forest regression is closely related to random forest classification discussed in Section 2.2.1. Again, this method builds an ensemble of decision trees. Random forest regression, however, aggregates the results of multiple regression trees instead of classification trees. These regression trees are similar to classification trees but estimate a numerical feature instead of predicting a categorical one. In fact, regression trees also recursively split the training data until a final partitioning provides simple models to evaluate the prediction for an input sample. To evaluate a predicted output for an input sample, regression trees typically use the average output feature of samples of the partition to which the input sample belongs. For the following partition of training features and associated output variables: $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$, the classic model for this partition would predict the output variable $\hat{y} = \frac{1}{n}\sum_{i=1}^{n} y_n$.

To identify the best split, regression trees cannot use the Gini impurity and require a different measure of split quality. To do so, we use the popular sum of squared error minimization. This method defines the best split cutoff to result in a partitioning with a minimum sum of squared error. The sum of squared error defines the error made by predicting the samples in a partition D using the predicted average output feature in that sample $\hat{y}$:

$$SSE(D) = \sum_{i=1}^{n}(y_i - \hat{y})^2. \tag{2.9}$$

Similar to random forest classification, random forest regression aggregates multiple decision trees to reduce variance and improve model generalization. Instead of using the mode of decision tree predictions to determine the eventual prediction, random forest classification uses the average prediction of the individual trees in the ensemble. To reduce the correlation between the trees in the ensemble and improve generalization, random forest regression also uses bagging and limits the number of features in each individual tree.

2.4 Supervised model training

Both the random forest classification and regression models discussed in the previous sections are supervised models that are trained on surgical data provided by the case hospitals. To obtain accurate prediction models whose performance generalizes to unseen data we use cross-validation to tune the random forest hyperparameters and obtain an unbiased estimate of the final model. Section 2.4.1 discusses the optimization method used to tune the model hyperparameters.

2.4.1 Bayesian optimization

Bayesian optimization is an optimization technique performing an efficient directed search of a global optimization problem. As this optimization method iteratively evaluates a set of parameters to optimize an objective function, it is similar to other exploratory optimization techniques like grid search and random search. Bayesian optimization constructs a surrogate function, a probabilistic model of the objective function, that is explored with an acquisition function to determine which parameter configuration to evaluate next [24]. This surrogate function approximates the mapping of the input variables to the output objective function. Mathematically, it represents the conditional probability $P(f|X)$ of the objective function f given the parameter configuration X. In this thesis, we use a *Gaussian process* to model the interaction between hyperparameters and the objective function, prediction model performance, providing a multivariate Gaussian distribution with mean and covariance properties. To determine which sample to evaluate next, several strategies exist leveraging these distribution properties. To direct the parameter search, the method proposed in this thesis chooses the sample resulting in the highest probability of improvement, most expected improvement, or about which the surrogate model is most uncertain (lower confidence bound) [24].

Bayesian optimization sequentially evaluates a predefined number of parameter configurations and reports the configuration resulting in the best optimization objective. The performance of a machine learning model (e.g. accuracy or MAE) can be optimized by searching the space of hyperparameters. Bayesian optimization allows a directed search over a complex global optimization problem and is thus suitable for the optimization of hyperparameters of a machine learning model.

2.5 Model explainability

To leverage the surgery clusters during scheduling, we predict into which cluster a surgery belongs. Next, a scheduling method may schedule the surgery accordingly. This scheduling system is unlikely to be adopted as a black box in clinical settings. Hospital staff will need to monitor, adjust and bear responsibility for the resulting schedules. In order for the machine learning approach to be used in such a clinical setting, scheduling staff needs to be convinced the predicted clusters correctly provide surgery scheduling characteristics. Hence, the prediction models need to be explainable. Machine learning explainability is an active field of study and various solutions exist to provide interpretable predictions. In the case of surgery cluster prediction, we want to explain the reason a surgery gets assigned to a particular cluster. To address this local explainability, we use Shapley additive explanations. Section 2.5.1 explains the explanations provided by this approach.

2.5.1 Shapley additive explanations

Shapley additive explanations (SHAP) are based on *Shapley values* and explain a prediction by computing the contribution of each feature to that prediction. *Shapley values*, a concept from coalition game theory, provide a way to fairly distribute the payout of a game among its players. In the context of explaining machine learning, we can consider a prediction to be a game and the features leading to that prediction its players. By computing the contribution of one feature to a specific predicted value, *Shapley values* can be used to explain predictions [25]. *SHAP* estimates a linear model and provides additive Shapley feature contributions.

Consider f to be the prediction model that we want to explain. Since we focus on explaining a prediction of a single input x, we explain the prediction of $f(x)$. To do so, SHAP uses an explanation model g. This model explains the prediction for a coalition $z' \in \{0,1\}^M$, where M is the number of features. The coalition z' is a representation of the present (1) and absent (0) features in the input. Since the explanation model is defined as

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i, \tag{2.10}$$

the model is indeed linear and additive. When evaluating the explanation of input instance x, we consider all features to be present and the coalition x' to be a vector of 1's. In that case, the explanation model becomes:

$$g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i. \tag{2.11}$$

This explanation model can be used to explain single feature contributions as ϕ_i represents the effect of feature $1 \leq i \leq M$ on the predicted value.

Lundberg et al. (2017) propose the *kernelSHAP* estimation method used in this thesis to estimate the explanation model g [25]. This method first randomly samples K coalitions $Z = z'_1, \ldots, z'_k$ and maps these coalitions to the input feature space by substituting present features with the input feature and absent features with randomly sampled values of that feature, resulting in the feature vector $h_x(z'))$. The predicted values for this feature vector can be computed by evaluating the prediction model: $f(h_x(z'))$. *KernelSHAP* optimizes a weighted sum of squared error loss $L(f, g, \pi_x)$ to fit the explanation model g. The weight of a coalition z', $\pi_x(z')$, is the weight of that coalition in Shapley value estimation. This weighting results in large weights for the sparse and dense coalitions with a large number of absent and present features, respectively. Optimizing the loss:

$$L(f, g, \pi_x) = \sum_{z' \in Z} [f(h_x(z')) - g(z')]^2 \pi_x(z'), \tag{2.12}$$

fits the explanation model g and estimates the Shapley values ϕ_i for every feature $1 < i < M$. These estimated Shapley values provide feature contributions that can be used to explain the effect each feature had on the predicted value.

Chapter 3

Problem exposition: case studies

As explained in section 1.1.1 of the introduction, this research is performed in close collaboration with two surgical departments of Dutch hospitals: the Cardiothoracic surgery department at Maastricht UMC+ (MUMC+) and the orthopedics department at Franciscus Gasthuis & Vlietland. The operating room scheduling of these departments serve as case studies for this research. We apply a general method developed towards leveraging surgery clusters in operating room scheduling to the challenge of scheduling surgeries in these case departments. By collecting the required domain knowledge and data understanding, in general, and for both hospital cases, this chapter describes the data and business understanding phases in the *crisp-DM methodology* [16]. Furthermore, we also address the data preprocessing step by explaining how we combine and preprocess the available data for the hospital cases. This chapter explains the general scheduling problem considered in this study in section 3.1. The problem of operating room scheduling has different aspects for the two hospital cases. Section 3.2 discusses the method we use to obtain business and data understanding of the two cases. In Section 3.3 we present the understanding resulting from this method. Next, Section 3.4 discusses the data made available by the case hospitals for this research. Finally, we examine the case-specific research goals and method requirements following from these hospital contexts in section 3.5.

3.1 The operating room scheduling problem

This section extends the general scheduling problem introduced in section 1.1 of the introduction. Recall this research studies the possibility of improving the operational operating room schedule within the block scheduling paradigm employed by most Dutch hospitals. The tactical organization level of hospital planning creates a block schedule in which surgery departments are allowed to schedule their elective surgical care [26]. This master surgery schedule (MSS) determines which

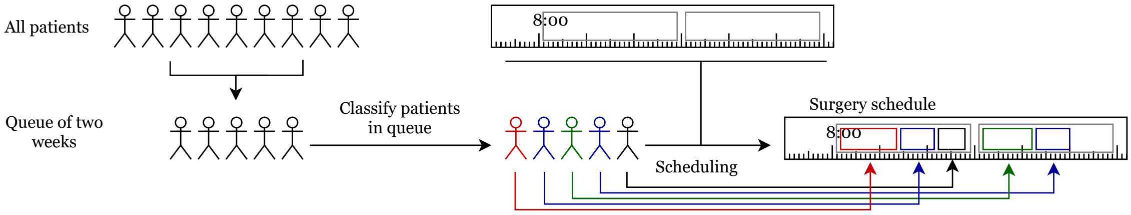


Figure 3.1: The simplified scheduling problem. Surgeries with different scheduling properties are displayed with different colors and should be scheduled within the available time slots accordingly.

operating rooms are available during which time slots for surgery by a hospital department. The specific problem considered in this research is finding a good assignment of surgeries to these available operating rooms and times. Figure 3.1 illustrates the general problem we address.

Finding the best way to plan surgeries in the master surgery schedule comes with a set of common challenges. Patients receiving surgery typically require care with various levels of urgency. Hospitals usually prioritize surgeries with higher urgency by scheduling patients from a priority queue, scheduling the most urgent surgery as soon as possible. Resources required by surgeries are limited and making efficient use of these resources is an important challenge in operating room scheduling. Resources like operating rooms and specialized medical staff are required for all surgeries. Some surgeries, however, need more specific resources like medical imaging tools, surgical robots, or prosthetics. Furthermore, surgical intervention is not without risk and specialized care is required to minimize the risk of complications during and after surgery. Postsurgical resources play a crucial role in the hospital providing this care. During scheduling, hospitals need to manage the availability of resources like beds in nursing wards with sufficient nursing staff and tools like respiratory equipment as well. Finally, an operating room schedule is subject to uncertainties that can result in scheduled surgeries being postponed, canceled, or rescheduled. These uncertainties impact the schedule at multiple levels. First, the surgical care demand is not always available before scheduling. Surgical planners adjust their schedules to surgeries that have to be scheduled on short notice. Schedulers generally do not plan emergencies (surgeries which have to be performed within the next 24 hours) during the scheduling of the elective surgeries but either fit in time reserved for uncertain demand or disrupt the initial schedule when emergencies are required. Moreover, the surgical process itself is uncertain as well [27]. The inherent variability of surgery durations complicates the operating room scheduling problem. Next to the duration of surgeries, other aspects like the availability of the patient and staff at the moment of surgery (due to e.g. late arrival in the hospital or an infection) impact the effectiveness of scheduling systems.

Grouping surgeries with the same scheduling characteristics and leveraging these characteristics during scheduling can help with managing these challenges. Note that many of the challenges are related to resources being required for particular surgeries and uncertainties leading to poor expectations of surgery features (like surgery case duration). Clustering surgeries can help distinguishing groups of surgeries that should be scheduled similarly. Ideally, all surgeries within one cluster have the same expected surgery scheduling properties and need the same resources during and after surgery. This way, knowing to which cluster a surgery belongs during scheduling provides an overview of the required resources and an expectation of the surgery features.

To investigate the potential of classifying patients in clusters based on similar characteristics during scheduling, this research focuses on the general scheduling problem described above to a limited scope that can be evaluated for different hospital cases. In this research, we consider the scheduling of a queue of patients in two weeks (starting on Monday) of available time slots. Before scheduling, the following scheduling components are available:

- The master surgery schedule: the set of operating rooms available for surgery and the time slots these OR's are available.
- The queue of patients, including the surgery and patient characteristics that can be used to predict scheduling characteristics.

Given these components, the scheduling goal is to allocate the biweekly queue of surgeries to start times in the master surgery schedule in the best possible way.

This focused scope is limited but allows the evaluation of schedules in a uniform way. By considering the case of scheduling a queue of selected patients (a queue with a limited number of patients), the effects of unscheduled care and surgery priority, in general, are omitted. By considering two weeks of surgeries as they have been performed in each case hospital, we manage the surgery priorities in the same way as the case hospitals. Furthermore, this way a baseline scheduling is available for evaluation. In section 10.2.1 the limitations of this scope are addressed more thoroughly.

3.2 Case problem definition approach

The previous section outlined the general research problem and showed that domain knowledge is required to adapt this general problem to two cases of hospital scheduling. This section discusses how we build an understanding of operating room scheduling in two case hospital departments. The general method is made clear in section 3.2.1, the specific steps are taken to adapt the challenge of leveraging surgery clusters in operating room scheduling to the department of orthopedics at Franciscus Gasthuis & Vlietland and cardiothoracic surgery in Maastricht UMC+ are discussed in sections 3.2.2 and 3.2.3 respectively.

3.2.1 Method for requirement analysis

To develop an understanding of the operating room scheduling problem and the challenge of making use of surgery clusters with planning characteristics, we investigate two things. First, we perform a literature study to become familiar with the concept of surgery scheduling and its current challenges and advancements. Section 1 summarizes the resulting problem understanding and we expand this with case-specific understanding. Discussions with domain experts such as surgery planners, surgeons, and hospital capacity managers result in additional executive perspectives on the theory in the literature. Furthermore, a Delphi study based on a series of two surveys among a panel of surgery scheduling professionals provides tangible relevance of scheduling goals and challenges.

3.2.2 Orthopedic surgery scheduling at Franciscus Gasthuis & Vlietland

We performed the case study of orthopedic surgery scheduling at Franciscus Gasthuis & Vlietland in close collaboration with the hospital capacity expertise center. This department is challenged with managing and improving hospital resource capacity, operations, and planning. In order to obtain an understanding of the scheduling challenges faced during orthopedic surgery scheduling, we took interviewed multiple experts involved with this scheduling. These experts included capacity managers and medical specialists. Table 3.1 lists the experts that were interviewed for this study and their functions. The in-depth discussions and interviews provided insight in the specific challenges surrounding orthopedic surgery scheduling. Following the interview with the dr. Sjoerd Rutten, we inspected the surgical (scheduling) process in person. We visited the operating rooms available for orthopedic surgery, associated wards, and waiting rooms. Furthermore, we investigated the daily operations and ad hoc adaptation of the surgery schedule as a result of uncertainties in the operating room planning.

Table 3.1: The panel of experts interviewed to investigate the problem of operating room scheduling for the orthopedics department at Franciscus Gasthuis & Vlietland.

Expert	Role
Renee van Houten	Manager capacity expertise centre
Mirjam Kerklaan	Manager admissionplanning and tactical planning
Milou Zwetsloot	Advisor capacity expertise centre
dr. Sjoerd Rutten	Orthopedic surgeon
Pauline Pos	Specialized physician assistant orthopedics, surgery scheduling
Marcel van den Aardweg	Doctor-anaesthesiologist
Suzanne Petra	Director of healthcare

To supplement the qualitative understanding obtained by interviewing the panel of experts, we asked them to participate in a Delphi study. This study, comprised of two surveys, was performed together with the interviewed personnel of MUMC+, discussed in the next section. To arrive at the group opinion of the panel of experts that was interviewed, we structured these surveys following the Delphi method [28]. The first survey consisted of open questions asking experts about the challenges and goals in operating room scheduling. The second survey aggregated the answers collected from the first survey and literature and shared these answers with all participants. In this second survey, we asked the experts to rate the relevance of the answers of all participants.

3.2.3 Cardiothoracic surgery scheduling at Maastricht UMC+

We investigated the cardiothoracic surgery scheduling at MUMC+ in close collaboration with both the Integral Capacity Management (ICM) department and the cardiothoracic surgery (CTS) department itself. Analogous to the method used to collect business understanding employed at Franciscus, we first interviewed a panel of experts involved with cardiothoracic surgery. Table 3.2 lists the interviewed experts.

Table 3.2: The panel of experts interviewed to investigate the problem of operating room scheduling for the cardiothoracic surgery department at MUMC+.

Expert	Role
Nol Visschers	Program manager integral capacity management
drs. Bart Scheenstra	Cardiologist
Jacqueline Scheijen	Staffadvisor integral capacity management
Nicole Beckers	Planner cardiothoracic surgery
Sylvia Frederix	Planner cardiothoracic surgery
dr. Patrique Segers	Cardiothoracic surgeon, head of surgery planning

The discussions and interviews with the experts listed above built an understanding of the specific challenges surrounding cardiothoracic surgery scheduling discussed in section 3.3. Following these interviews, we investigated the surgical process in person. We visited the wards, intensive care unit, and medium care unit available for cardiothoracic surgery, and waiting rooms. The scheduling method was discussed with operating room planners and surgeons. Furthermore, we inspected the daily operations and ad hoc adaptation of the surgery schedule as a result of uncertainties in the operating room planning.

Some cardiothoracic surgeries require specific resources. The scheduling requirements of cardiothoracic surgeries contained specific constraints like surgeries requiring unique resources. An overview of the required resources and possible operating rooms per surgery type performed at the cardiothoracic surgery department was requested.

Finally, we asked the interviewed experts at MUMC+ to participate in the same Delphi study as the experts of Franciscus Gasthuis & Vlietland. This allows the comparison between the relevance of challenges and goals in both hospitals.

3.3 Hospital context

The previous section discusses the method used to define the case hospital-specific research problems. This section summarizes the results from this method and presents the business understanding obtained with it. We discuss the characteristics of the two case hospital departments. Furthermore, we explain and compare the challenges of operating room scheduling for these departments in this section.

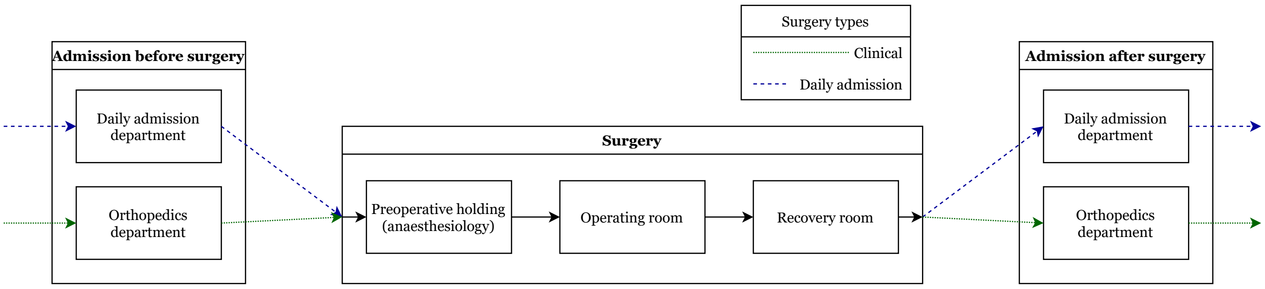


Figure 3.2: The surgical process, describing the patient logistics for orthopedic surgery at Franciscus Gasthuis & Vlietland.

3.3.1 Orthopedic surgery scheduling at Franciscus Gasthuis & Vlietland

Orthopedic surgery treats patients with conditions affecting the musculoskeletal system, including trauma, spinal diseases, sports injuries, infections, tumors and more [29]. The orthopedic surgery department operates using a planning horizon of 9 weeks. This means the department schedules surgeries 9 weeks in advance. Orthopedic surgeries take an average of 72 minutes, which allows an average of 4.2 surgeries to take place in each surgery session. With 14.6 OR sessions available to orthopedic surgery in the master surgery schedule, the orthopedics department is able to perform 61,4 surgeries on average per week. Planners manually schedule patients from a priority queue in the master surgery schedule provided by hospital planners on the tactical level. Scheduling staff fills this priority queue with patients that are prescribed surgery either in the outpatient clinic or while being admitted to the hospital. While scheduling patients in the time slots available in the master surgery schedule, planners aim to schedule patients in order of priority while adhering to a complicated set of scheduling constraints. Furthermore, planners manage the number of beds required in the nursing wards for patients that have received surgery.

After surgery, the hospital admits the patients are to either the orthopedics department or the daily admissions department. The latter houses beds intended for patients who are expected not to stay overnight. In most cases, patients admitted to one of these wards return there after surgery and do not require admission to another surgery ward. When complications occur during the surgery or recovery thereof while the patient was admitted to the daily admission ward, the hospital transfers the patient to the orthopedics ward. Figure 3.2 displays the intended surgical process for orthopedic patients.

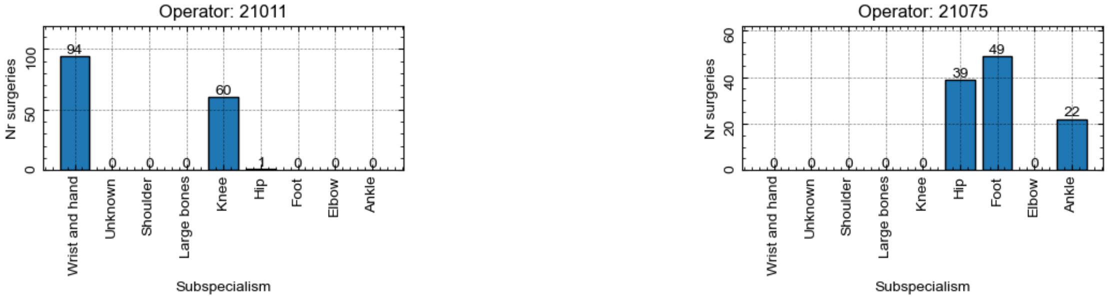
The orthopedic surgery schedule is subject to unforeseen circumstances, like high priority surgeries forcing deviations from the surgical schedule and uncertain surgery durations, and has to manage these during the execution of the schedule. However, the department rarely treats emergencies as these are treated in dedicated time slots in the master surgery schedule. This means emergencies rarely introduce severe problems like last-minute cancellation or rescheduling of surgeries.

Challenges

As discussed in section 3.1 the operating room scheduling problem manages several constraints. From the interviews of scheduling stakeholders, we identify some specific constraints relevant for orthopedic surgery scheduling. To adhere to these constraints, our scheduling method should fulfill the following requirements:

- Scheduling faces the challenge of managing several resources. Specifically, orthopedic surgery scheduling should manage the following resources:

    1. *Operating rooms*: The operating room availability is determined by the master surgery schedule and a limited set of operating rooms is available for surgery day.



(a) The specialization of operator '21011', this surgeon specializes in Knee, wrist and hand surgeries.

(b) The specialization of operator '21075', this surgeon specializes in Hip, Foot and Ankle surgeries.

Figure 3.3: The specialization of specialists '21011' and '21075'. Note that both operators perform surgeries belonging to different surgeries.

2. *Surgical staff*: For every surgery, a sufficiently specialized surgeon needs to be available. Surgeons have different specialisms and the orthopedic surgery department associates each surgery with one of the following subspecialisms: 'wrist and hand', 'shoulder', 'hip', 'knee', 'large bones', 'foot', 'elbow' or 'ankle'. Unfortunately, some surgeries in the provided data could be associated with one of these subspecialisms, these surgeries were assigned to the 'unknown' subspecialism. Figures 3.3a and 3.3b show for two specialists, the number of surgeries per subspecialisms performed by the specialists.
3. *Postoperative ward capacity*: The department to which the patient gets admitted before and after surgery is required to have a bed available for the patient. To minimize the impact of the scheduled surgeries on the capacity in the nursing ward, surgery schedulers minimize the variability of the number of beds required in the ward. Furthermore, patients are admitted either clinically, or as a daily admission. And need to be admitted to the corresponding ward after surgery (orthopedics ward, or the daily admission ward).

- Despite the challenges associated with operating room scheduling for the orthopedics department at Franciscus Gasthuis & Vlietland, the schedules developed by surgery planners are considered to perform well. No major issues arise during the execution of these schedules and the queue of patients waiting is acceptable under normal circumstances. Hence, our scheduling method should not implicate this performance by increasing patient waiting times.

3.3.2 Cardiothoracic surgery scheduling at Maastricht UMC+

Cardiothoracic surgery is the field of surgery focusing on surgery of the heart, lungs, and other organs in the thoracic cavity [30]. This research focuses on the cardiac surgeries performed by the cardiothoracic surgery department at MUMC+. Cardiothoracic surgery takes an average of 243 minutes, which allows an average of 1.5 surgeries to take place in each surgery session. With 10.4 OR sessions available to orthopedic surgery in the master surgery schedule, the orthopedics department is able to perform 15.8 surgeries on average per week. The cardiothoracic surgery department schedules their surgeries for the upcoming two weeks from a priority queue. Before patients are placed on this priority queue, they need to be accepted in a multi-disciplinary meeting (MDO). During this MDO the patient and surgery characteristics are discussed and a decision is made on the required surgery. As patients are being planned two weeks in advance, they also receive notice of the exact date and time of their operation no sooner than 2 weeks before surgery.

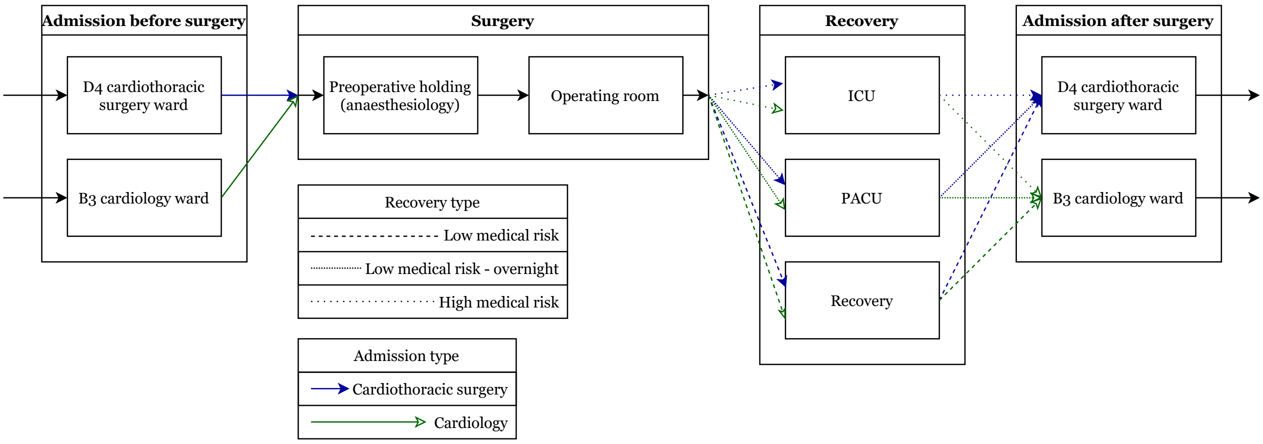


Figure 3.4: The surgical process, describing the patient logistics for cardiothoracic surgery at MUMC+.

Scheduling staff manually schedules cardiothoracic surgeries by assigning patients from the priority queue to the time slots assigned by the hospital in the block scheduling schema (the master surgery schedule). Generally, planners aim to schedule patients in order of priority. However, planning constraints frequently force planners to deviate from this method. Examples of these interfering constraints include a limited set of surgeons or operating theatres being available for surgeries.

After a surgery is scheduled in the upcoming two weeks, the patient gets notified. The department admits a patient the day before surgery and that patient recovers from the surgery in an intensive care unit or ward available to the cardiothoracic department. When patients are referred from a different hospital, the patient is transferred from the referring hospital to MUMC+ and gets transferred back a few days after surgery.

After cardiothoracic surgery patients are admitted either to the post anaesthesia care unit (PACU) or recovery rooms (when there is a low medical risk) or to the intensive care unit (ICU) followed by the coronary care unit (CCU) (when there is a large medical risk). The PACU is designed for patients that have to be in recovery overnight. Ultimately, the hospital moves most patients to the D4 cardiothoracic surgery ward. However, some patients are moved to the B3 ward, specializing in general cardiology. Figure 3.4 displays the patient's logistical process.

The surgery schedule is subject to unforeseen circumstances like emergencies or a lack of available postsurgical resources. Scheduling staff manually adjusts the schedule ad hoc, to manage these circumstances. Rescheduling this way might result in a last-minute surgery postponements. These postponements occur within 24 hours of the scheduled surgery time and increase healthcare costs and patient distress [31] [32]. The cardiothoracic department at MUMC+ canceled and rescheduled 13.1% of the surgeries scheduled between January 2017 and June 2019. These postponements were due to process-related factors in 71,4% of the canceled cases. Overtime of previous surgeries, other patient emergencies and lack of ICU capacity are the most frequent process-related factors resulting in postponements.

Challenges

Similarl to the orthopedic surgery scheduling case, several challenges exist for cardiothoracic surgery scheduling. Again, during the stakeholder interviews, we identified several case-specific constraints that result in requirements for our scheduling method. The following requirements are relevant for cardiothoracic surgery scheduling:

- Similar to the case of orthopedic surgery scheduling, a scheduling method should efficiently manage the resources required for surgery. For cardiothoracic surgery the following resources pose scheduling constraints:

1. *Operating rooms*: The limited availability of operating rooms is provided by the master surgery schedule. For cardiothoracic surgery, however, not every surgery can take place in every operating room. Some surgeries require specific resources and can take place in specific operating rooms. Specifically, a surgical robot used for minimally invasive direct coronary artery bypass (MIDCAB) surgery and 3D reconstruction tools used in minimally invasive surgeries are both available in just one operating room (OR 15 and OR 18 respectively). Furthermore, one operating room is designed for hybrid surgeries (in which cardiothoracic surgeons and cardiologists cooperate) requiring radiography equipment and hence these surgeries should take place in this specific room (OR 16). Furthermore, surgical staff specializes in particular surgeries. This means that scheduling a surgery requires staff to be both available and specialized in that surgery. The cardiothoracic department at MUMC+ has approximately 2 operating rooms available every day.
2. *Surgical staff*: Again, sufficiently specialized staff needs to be available to perform the scheduled surgeries. In contrast to orthopedic surgery, this department does not have subspecialisms in which surgeons specialize. Here, specific surgeries are performed by a select few surgeons. Unfortunately, no data is available regarding the availability of this specialized staff.
3. *Postoperative ward capacity*: Not only the capacity of operating rooms determines the viability of an OR schedule, the resources that will be occupied by patients returning from surgery also have limited capacity. After surgery patients move to either a recovery room, post-anesthesia care unit (PACU), intensive care unit (ICU), medium care unit/coronary care unit (MCU/CCU), or the nursing ward. The capacity of these departments is limited and patients often require specific facilities available only on these departments, i.e. respiratory equipment. When treatments are planned that need to move patients to such departments while there is no capacity available on these departments, the surgery cannot be performed. Every morning, the ICU staff discusses the planned surgeries with the cardiothoracic surgery department. If insufficient ICU capacity is available surgeries need to be postponed.

- Uncertainty in care demand is the primary reason the cardiothoracic OR schedule needs to be revised. Unfortunately, some surgeries are urgent and need to happen within the upcoming two weeks. A scheduling solution needs to be able to manage this unplannable care. Currently, the scheduling staff does not place these surgeries in the priority queue but rather schedules them within the (possibly full) current schedule. Emergency surgeries need to be treated within 24 hours and can thus never be planned in advance. The cardiothoracic department of MUMC+ performs emergencies at least once a week (frequently requiring a last-minute rescheduling of other surgeries). The cardiothoracic surgery planners do not reserve space in the OR schedule for emergencies directly but do reserve time slots for surgeries of clinical patients (patients occupying a bed in the hospital).

3.4 Available data

Solving the general problem described in section 3.1 requires specific data to be available. Specifically, we need to consider the following data: the master surgery schedule, surgery scheduling characteristics, and patient and surgical process features. However, the case hospitals do not have this data readily available. This section explains the required data and how we prepare the data which is available for both case hospitals to obtain the data required for our scheduling method. By preparing the data for the scheduling method discussed in the next chapter, this section describes the *crisp-DM* data preparation phase.

The master surgery schedule denotes the available operating room time slots for surgery. The surgery scheduling characteristics are the features that will be estimated during scheduling and are used to define the surgery classes. The patient and surgical process features correspond to the historic patient and surgery information available a priori. This data will be used to identify to which surgery class a surgery belongs before scheduling.

Section 3.4.1 and 3.4.2 describe how to extract and combine the data available for orthopedic and cardiothoracic surgery, respectively. Next, Section 3.4.3 summarizes the data which will be used in this research and compares the data of the two hospital cases.

3.4.1 Orthopedic surgery at Franciscus Gasthuis & Vlietland

The data available for the case study of orthopedic surgery is provided by the capacity expertise center of Franciscus Gasthuis & Vlietland. This data consists of multiple datasets related to the surgery scheduling in 2018 and 2019. For this research, we collect the required data from four provided datasets. The provided datasets explain operating room time slots, admissions, surgeries, or patients.

Since the scheduled start and end times are available for analysis in this dataset, the original planning made by the hospital is available. This allows us to evaluate the schedules found with our method by comparing them to the hospital schedule.

In general, the relevant surgery characteristics are collected and aggregated in a single dataset of surgeries. We one-hot encode the categorical features in this dataset and standardize continuous features to be distributed normally with a mean of 0 and unit variance.

Master surgery schedule

The master surgery schedule is directly available as a single dataset. The operating room time slot file lists the time slots in which operating rooms are available to the orthopedics department.

Surgery scheduling characteristics

In contrast to the master surgery schedule, the scheduling characteristics unfortunately are not available as a single dataset. The surgery dataset contains the scheduled and actual start and end times of surgeries (the moment the patients come in and out of the OR respectively), required to obtain the surgery durations. This dataset also provides the surgery description needed to determine surgery subspecialisms This surgery scheduling information is supplemented with the admission dataset providing the admission information (admission types, associated department, room, bed, and start- and end-times). We use these admissions to determine the length of stay of the patient after surgery and to identify whether or not a patient is admitted to the orthopedic or daily admission ward after surgery.

Retrieving this from the respective datasets is no trivial task. Some surgery features require additional data preprocessing to be extracted. For example, we define surgery times (both scheduled and realized), as the difference between start and end times.

Additionally, the postoperative length of stay feature requires multiple datasets to be combined and preprocessed to be used in the scheduling method. We define postoperative length of stay as the time between the end of the surgery and the end of the last admission to a nursing ward following the surgery. This features requires the duration of all department admissions after surgery and we collect it by creating an event log of surgery events and admissions to wards before and after surgery. In creating the event log, the admission in which a surgery takes place determines which admissions are associated with that surgery. We associate all subsequent admissions with a gap in hospital admission no larger than 24 hours with the surgery. Finally, we retrieve the postoperative length of stay of every surgery by computing the time expired between the end of the surgery and the end of the last admission associated with that surgery.

Finally, in order to assign surgeries to an initial clustering associated with a subspecialism and expected department after surgery, we need to preprocess that data. No categorical surgery feature, indicating the subspecialisms of surgeries is present, so we manually extract this data from unstructured text in the surgery dataset using basic natural language processing. We determine the categorical feature subspecialism by defining tokens that are associated with specific subspecialisms. The surgery policy (a description of the surgery that is to be performed) is available in the surgery dataset as free text. We consider each surgery policy word multiset (bag of words) to belong to a specific subspecialism if it contains a token of that type. The subspecialisms and their associated tokens are set up using domain knowledge. The (case-insensitive) tokens used to determine subspecialisms based on surgery policy are available in appendix A. In contrast to the subspecialism feature, the expected department after surgery is readily available. Admissions are labeled as associated with a 'clinical' or 'daily' admission. Surgeries corresponding to an admission that is labeled 'clinical' are supposed to be admitted to the orthopedics department after surgery, while 'daily' patients move to the daily admission department. This allows us to directly retrieve the expected postoperative department.

Patient and surgical process features

Similarly, to the scheduling characteristics, the a priori features used to predict surgery clusters, do not originate from a single source. These features combine surgery information known before scheduling available in the surgery, patient, and BMI datasets. These datasets provide generic surgery and patient information like surgery type, expected surgery duration, patient age, and body mass index (BMI). The expected duration is computed by subtracting the scheduled surgery start time from its scheduled end time.

3.4.2 Cardiothoracic surgery at MUMC+

Surgery data is maintained by two departments in MUMC+. The integral capacity management department keeps data on which surgeries are scheduled and performed. This dataset also contains for every surgery the associated patient and admission information. However, the patient data collected by the ICM department is limited and does not contain an exhaustive set of surgery characteristics. Patient and surgery information relevant to the medical specialists in MUMC+ is collected by the surgical departments themselves. As such, the cardiothoracic surgery department has a business information management (BIM) team that maintains surgery *MDO* form data. This *MDO* form is used to collect data of the patient and surgery departments to decide on the surgery policy. In order to use this dataset containing relevant medical patient and surgery information, we enrich the ICM surgery data with the cardiothoracic surgery *MDO* data for this research. This research used two years of operating room scheduling data, collected in 2018 and 2019.

The data provided by ICM department contains multiple datasets. Admission data is stored as the times per patient when he or she was admitted to a particular department. The surgery data contains, for every surgery, which patient is treated in which operating room at what times. Due to the MUMC+ not storing scheduling history or planned times, the times surgeries were scheduled to be performed are not available. This complicates the evaluation of our method, as will be further discussed in section 10.2. The general patient data available through the ICM dataset contains basic patient information including the age, sex, and BMI of patients receiving surgery. Finally, an operating room session dataset is available.

Again, the relevant surgery characteristics are collected and aggregated in a single dataset of surgeries. We one-hot encode the categorical features in this dataset and standardize continuous features to be distributed normally with a mean of 0 and unit variance.

Master surgery schedule

Similar to the orthopedic scheduling case, the master surgery schedule is provided in a single operating room sessions file. This dataset of operating time sessions lists the start and end times of time slots in which an operating room is available for surgery by a specified department.

Surgery scheduling characteristics

Several surgery characteristics are relevant to adhere to the constraints posed on cardiothoracic surgery scheduling. The surgery duration is available in the general ICM surgery dataset. We define the surgery duration as the difference between the OR start and end times (the moment the patient is brought into the operating room and is moved out of the OR after surgery).

Similarly to the preprocessing for orthopedic surgery we define the postoperative length of stay as the time between the end of the surgery and the end of the last admission to a nursing ward following the surgery. Again, we collect this scheduling characteristic by creating an event log of surgery events and admissions to wards before and after surgery. Next, we retrieve the postoperative length of stay of every surgery by computing the time expired between the end of the surgery and the end of the last admission associated with that surgery from the event log.

To adhere to the constraints posed by surgeries requiring ICU care or a specific operating room, we preprocess the data further. No Boolean surgery feature, indicating whether or not a patient is expected to recover in the ICU or feature denoting the operating theaters in which the surgery can take place is available. So, analogous to how we obtained orthopedic subspecialisms, we extract this data manually from unstructured text in the *MDO* form provided by the cardiothoracic surgery department using basic natural language processing. Specific surgery types require a particular operating room or are expected to require intensive care capacity. We determine the categorical feature surgery type by defining tokens that are associated with types of surgery. We consider each surgery policy word multiset (bag of words) to be of a particular surgery type if it contains tokens of that type. To link surgery types to the requirement of ICU availability and possible ORs (and in defining the tokens associated with surgery types), we use domain knowledge obtained from interviews with surgery stakeholders. The tokens used to determine surgery types based on surgery policy are available in appendix A.

Unfortunately, the surgeons specialized to perform a surgery was not possible to extract from the available data for every surgery. Knowing which specialists are available for each surgery is necessary in order to reschedule a surgery in a different slot (as one of these specialists needs to be available in that time slot). Since our method only reschedules surgeries in the same time slot, the missing information on available specialists is not problematic.

Patient and surgical process features

The list of a priori surgery characteristics, used to predict surgery properties like associated clusters, is extensive for cardiothoracic surgery. The general patient features like age and BMI provided by the ICM department are supplemented with the surgery *MDO* form which collects the data and surgery policy decisions made by medical specialists before accepting the patient in the surgery queue. This *MDO* form contains structured (categorical or continuous) data about the surgical process, patient and nature of the surgery. Surgical process data includes the referring care centre and urgency. Furthermore, patient information like medical patient features such as kidney and left ventricular function are available. Finally, comorbidities (additional diseases the patient suffers) are collected and aggregated patient health statuses like the NHR ('Nederlandse Hart Registratie' or Dutch Heart Registration) logistic score and Euroscore2 are collected in the MDO form. These scores are standardized scores used to rate patient health.

3.4.3 Exploratory statistics

In the previous subsections, we explained the available data for the orthopedic and cardiothoracic surgery scheduling cases and discussed how we prepare this data for our scheduling method. Here, we present an exploratory analysis of the case hospital data.

Orthopedic surgeries

The surgery schedule statistics indicative of surgery scheduling for orthopedics at Franciscus Gasthuis & Vlietland are provided in this section. Some surgery statistics describing the surgeries scheduled for orthopedics are provided in table 3.3. A more exhaustive list of surgery characteristics used in this research (including features used in classification) is available in table B.1 in appendix B.

Table 3.3: Characteristics of the surgeries performed by the orthopedic department at Franciscus Gasthuis & Vlietland.

Characteristic	**(N=9452)**
Duration (min)	71.7±33.9
Post operative lenght of stay (hours)	41.1±70.4
Surgery type - no./total no. (%)	
Clinical	6781/9452 (71.7)
Daily admission	2671/9452 (28.3)

Figure 3.5b shows the distribution of the number of surgeries performed by the orthopedics department on a weekday in Franciscus Gasthuis & Vlietland. We show the typical duration, length of stay in the nursing ward, and expected department after surgery in figures 3.5a, 3.5c, 3.5d respectively.

Note the relatively large number of surgeries performed, short surgery duration, and postoperative length of stay for surgeries in the orthopedic surgery schedules (compared to the cardiothoracic surgery scheduling at MUMC+).

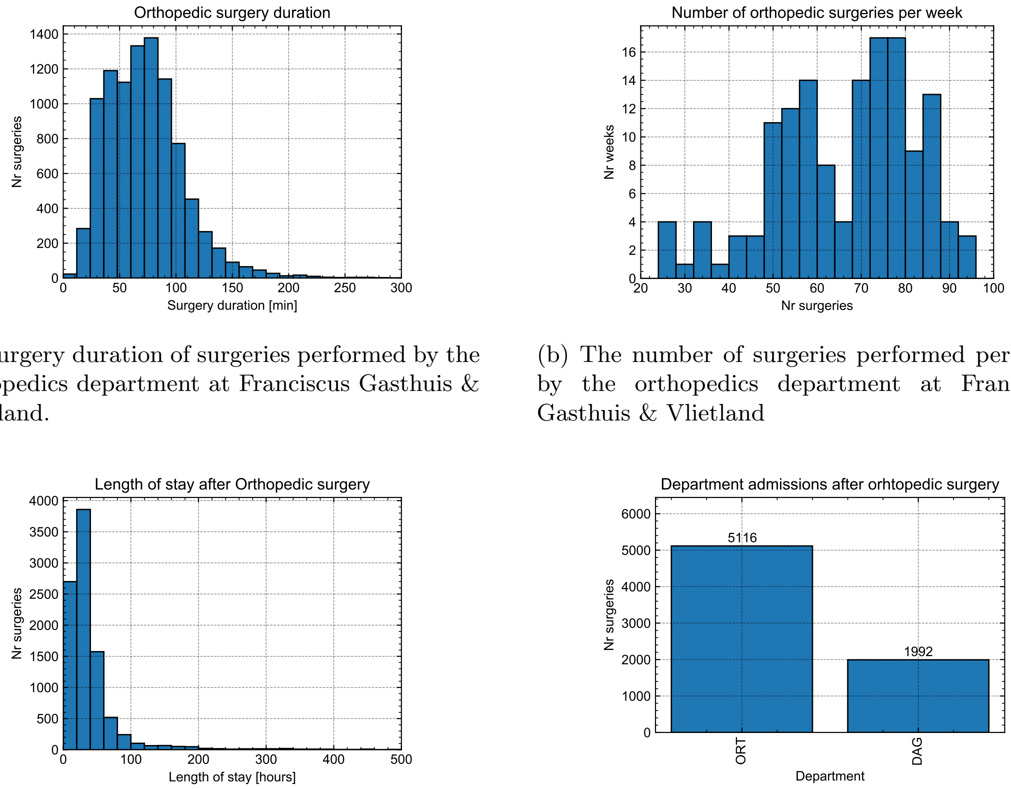
Cardiothoracic surgeries

The surgery schedules of the cardiothoracic surgery department at MUMC+ are investigated in this section. Table 3.4 lists some important statistics, describing the surgeries used in this research. We provide a more exhaustive list of surgery characteristics, including the features used in classification, in table B.2 in appendix B.

Table 3.4: Characteristics of the surgeries performed by the cardiothoracic surgery department at MUMC+

Characteristic	**(N=1643)**
Duration (min)	242.7±132.8
Post operative lenght of stay (hours)	175.4±176.3
Possible OR - no./total no. (%)	
All	1290/1643 (78.5)
VH-OK16	171/1643 (10.4)
VH-OK15	107/1643 (6.5)
VH-OK18	75/1643 (4.6)
Postoperative IC required - no./total no. (%)	1619/1643 (98.5)

Figure 3.6b shows the distribution of the number of surgeries performed by cardiothoracic surgery department on a weekday in MUMC+. We display the typical duration, length of stay in the



(a) Surgery duration of surgeries performed by the orthopedics department at Franciscus Gasthuis & Vlietland.

(b) The number of surgeries performed per week by the orthopedics department at Franciscus Gasthuis & Vlietland

(c) The length of stay after surgeries performed by the orthopedics department at Franciscus Gasthuis & Vlietland

(d) The departments where patients recover from orthopedic surgery in Franciscus Gasthuis & Vlietland

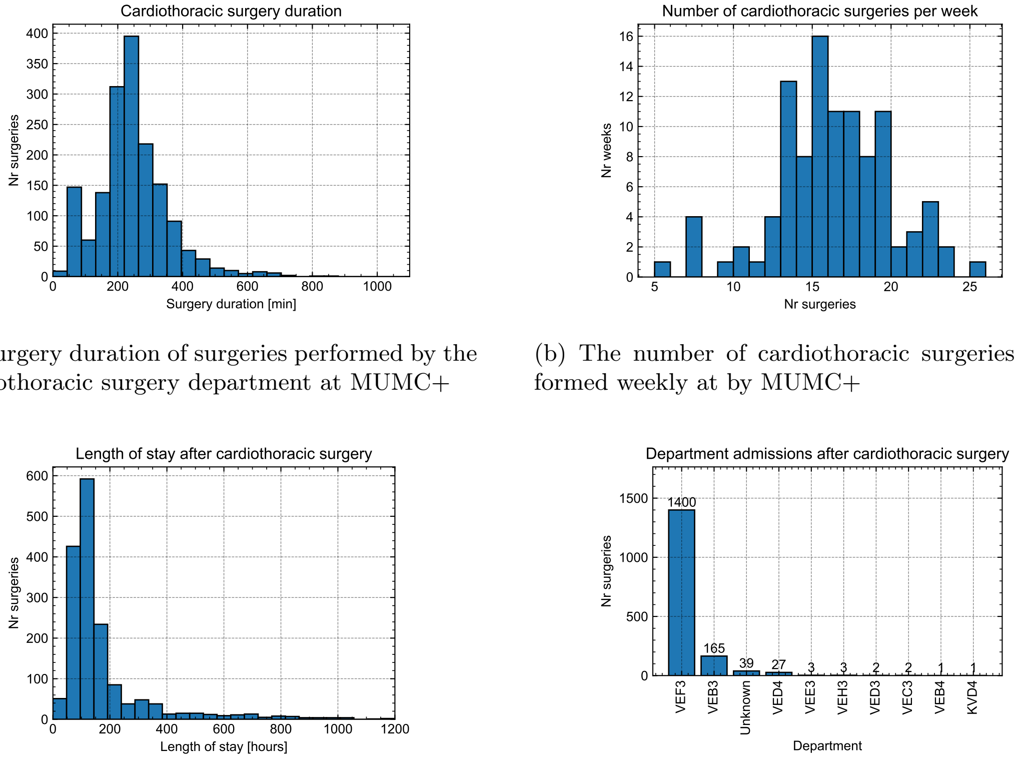
Figure 3.5: Scheduling characteristics of surgeries performed by the orthopedics department at Franciscus Gasthuis & Vlietland

nursing ward and expected department after surgery in figures 3.6a, 3.6c, 3.6d respectively.

Note the relatively long surgery duration, long postoperative length of stay, and small throughput of the cardiothoracic surgery schedule (compared to the orthopedics surgery scheduling in Franciscus Gasthuis & Vlietland) indicative of the complicated surgeries and scheduling problem.

Case hospital comparison

When comparing the two case hospital statistics, the different nature of the scheduling cases becomes apparent. First, the duration of cardiothoracic surgeries is longer than that of orthopedic surgeries. This corresponds to the complex surgeries performed by the cardiothoracic department. Also, the long duration of cardiothoracic surgeries explains why only a small number of surgeries are performed in available the time slots. Also, the orthopedic surgery department performs more surgeries each week and has a smaller postoperative length of stay. Scheduling more but shorter surgeries helps to combine surgeries with compatible characteristics in a schedule. This emphasizes the difficult scheduling challenge faced by the cardiothoracic surgery department.



(a) Surgery duration of surgeries performed by the cardiothoracic surgery department at MUMC+

(b) The number of cardiothoracic surgeries performed weekly at by MUMC+

(c) The length of stay after cardiothoracic surgery in MUMC+

(d) The departments where patients recover from cardiothoracic surgery in MUMC+.

Figure 3.6: Scheduling characteristics of surgeries performed by the cardiothoracic surgery department at MUMC+. The departments in figure d are care units associated with specific departments, providing different levels of care. Departments 'VEF3', 'VEE3' and 'VED3 are intensive care units, 'VEH3', 'VEC3' and 'KVD4' are medium care units belonging to the cardiothoracic ward and departments 'VEB3' and 'VEB4' are nursing wards associated with the cardiology department.

3.5 Research goals and requirements

As discussed in section 3.1, there are similarities between the surgery scheduling problem in different hospitals and hospital departments. A general method used to find clusters, one to predict them for surgeries being scheduled and a scheduling method making use of these clusters is developed and discussed in chapter 4. This general method serves as a framework and lacks concrete implementations and specific goals. In order to answer how surgery clusters can be used in operating room scheduling, some intermediate challenges, corresponding to research sub-goals, need to be solved:

SG1 While the general scheduling problem is similar for different hospital cases, each case also has individual challenges and scheduling goals. Before determining how surgery clusters can be used in scheduling, it is necessary to determine what constitutes better planning. To this end, we require a set of scheduling objectives with which surgery schedules can be compared. Each case hospital might prioritize different scheduling objectives as their challenges differ. We investigate the universal scheduling objectives relevant to our scheduling problem in chapter 5.

SG2 To leverage surgery clusters in scheduling, first, meaningful groups of surgeries need to be discovered. Section 3.3 discusses the differences in the surgical process in case hospitals. Each of these cases has different scheduling requirements corresponding constraints posed on specific surgeries. Orthopedic surgeries belong to subspecialisms and move to a specific postoperative department while cardiothoracic surgeries can take place in specific operating rooms and might require intensive care. We address these constraints by grouping surgeries with similar constraints. We discuss the way surgeries can be discriminated in clusters, based on these surgical process characteristics, in chapter 6.

SG3 Not only the surgical process but also the patient and surgery information available before surgery is different among hospital departments. To be able to adhere to the constraints for both hospital cases (identified in Section 3.3), the clusters identified in the previous sub-goal need to be accurately predicted. Chapter 7 discusses how for surgeries that are to be scheduled, their surgery cluster can be predicted.

SG4 The predicted surgery clusters are associated with surgery process characteristics that can be used in scheduling. To leverage these clusters, we need to develop a method able to schedule surgeries based on their predicted schedules. This method needs to be able to both leverage the surgery clusters and fulfill the patient waiting time and uncertain care demand requirements discussed in section 3.3. Additionally, it needs to maintain the schedule feasibility by adhering to the scheduling constraints posed on every surgery. Depending on which scheduling objectives (discussed in chapter 5) are prioritized, a different scheduling method could result in the preferred schedule. Chapter 8 explains how these surgery clusters and their expected surgical processes can be used in scheduling. Furthermore, this section shows how our surgery scheduling method can be used to improve the operating room scheduling in the two hospital cases.

Chapter 4

Overview scheduling approach

This chapter discusses the method used in this research to schedule surgeries making use of surgery clusters. Chapters 1 and 3 introduced the problem of surgery scheduling and discussed it in the context of two case hospitals. The overview provided in this chapter expands the general method explained in section 1.4 and discusses how we address each subproblem defined in section 3.5.

The scheduling method first clusters surgeries based on their scheduling characteristics. Next, it predicts the surgery cluster to which a surgery that is being scheduled belongs. We use these clusters to estimate the surgery scheduling characteristics of the surgeries being scheduled. Finally, heuristic scheduling strategies use the estimated surgery duration to schedule the surgeries. Figure 4.1 illustrates this general approach. The following Chapters 6, 7 and 8 elaborate the specific steps in this general approach. Hence, these chapters explain the modeling phase of *crisp-DM*. Finally, we evaluate this scheduling approach for the orthopedic and cardiothoracic surgery scheduling cases in Chapter 9, corresponding to the evaluation phase of *crips-DM*.

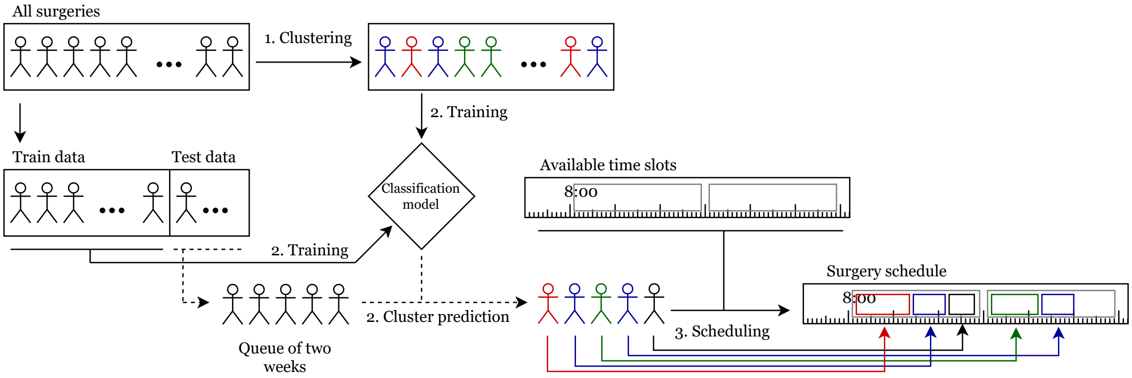


Figure 4.1: The general scheduling approach. First, we obtain a ground truth clustering. Next, this clustering is predicted using a classification model. Finally, the predicted surgery clusters are used to schedule two weeks of surgeries in the available time slots.

Figure 4.1 shows the first step in our approach is to group the surgeries based on their scheduling characteristics and to do this, we use a clustering method. This corresponds to the second research sub-goal *SG 2* from Section 3.5, which states we need to be able to group surgeries with similar scheduling characteristics and constraints. We tackle this sub-task using the unsupervised machine learning method *clustering*. This task is explained in detail in 2.1 and aims to group observations based on features. This task readily translates to the context of grouping surgeries based on scheduling features and clustering thus is uniquely suited to apply to this problem. In the clustering method used in this thesis, we first manually assign surgeries to initial clusters of surgeries with different scheduling constraints. Next, we employ a data-driven clustering method to distinguish these clusters based on their scheduling characteristics. This two-step clustering method is explained further in Chapter 6.

After the clusters of surgeries with distinct constraints or scheduling constraints are found, we need to determine to which cluster a surgery that is being scheduled belongs. This prediction task corresponds to the research sub-goal *SG 3* from Section 3.5. We address this task, predicting clusters based on surgery and patient features, with the supervised machine learning method of *classification*. This method, explained in more detail in section 2.2, predicts a categorical dependent feature of an observation based on a set of independent features belonging to that observation. As we consider the surgery cluster to be a categorical feature of a surgery, classification is a suitable prediction method for this sub-goal. Hence, Figure 4.1 shows we use a classification model to predict the cluster for surgeries that are being scheduled. This classification model is trained on a dataset dedicated for training. Similarly, a dataset is reserved for the evaluation of the classification and scheduling method. In Chapter 7 we explain the specific classification method and discuss how we evaluate its performance.

When we have grouped the surgeries with different constraints and scheduling characteristics, we need to develop a scheduling method able to schedule surgeries based on their cluster characteristics. This challenge corresponds to the research sub-goal *SG 4* from section 3.5. To do so, we employ *heuristic scheduling strategies* that schedule two weeks of available time slots. Chapter 8 explains how we use scheduling heuristics to schedule surgeries based on their predicted cluster. This chapter also discusses how we evaluate the schedules resulting from these heuristics.

Chapter 5

Objectives in operating room scheduling

To be able to evaluate operating room schedules and therewith the scheduling method proposed in this research, clear performance indicators are required. Section 3.5 described that, in order to leverage surgery clusters in operating room scheduling, several sub-problems need to be solved. The first sub-problem is to define a set of concrete scheduling objectives that can be used to compare OR schedules and evaluate scheduling methods. This section addresses that subproblem in two steps. First, we survey a panel of operating room scheduling stakeholders to decide on the relevant scheduling performance factors. Section 5.1 describes the Delphi study employed to retrieve these factors. Finally, in section 5.2, we translate the scheduling performance factors relevant to our research scope into concrete scheduling performance measures.

5.1 Scheduling performance factors

The desired concrete operating room scheduling objectives investigated in this chapter are based on factors impacting the OR schedule performance. We identify these factors, which are used to compare surgery schedules, in this section. We use a Delphi study to achieve a consensus of universal performance factors among a panel of hospital stakeholders. This section describes the Delphi study research method and its results.

5.1.1 Delphi study survey

Designing technological solutions, like the scheduling method studied in this research, can benefit from stakeholder expertise [33]. We involve the stakeholders at the investigated case hospitals to identify their system requirements and objectives [34]. The Delphi method is a research method designed to build a consensus among subjective individual judgments [28] and can be an effective participatory design approach to capture the collective opinion of a wide range of stakeholders [35]. We use a Delphi study to arrive at the collective opinion regarding OR scheduling challenges, goals and, consequences of a panel of hospital stakeholders. The scheduling performance factors obtained from this collective opinion serve as the basis for the scheduling performance measures used to compare OR schedules.

In the case of data-driven operating room scheduling support, several stakeholders are available with a lot of problem understanding. These stakeholders have been discussed in sections 3.2.2 and 3.2.3 and participate in the design of the scheduling objectives used in this research. The participants are listed in Tables 3.1 and 3.2. Note that most of these stakeholders are experts in the field of operating room scheduling. Their expertise is not limited to the effects of the surgery schedule, but they have experience with scheduling surgeries themselves. Hence, we aim to identify performance factors with these stakeholders that generalize to other surgical scheduling settings (like other hospitals).

The Delphi study performed for this research arrives at a consensus of the scheduling challenges and goals in two steps. With an initial survey, we collect the individual opinions of the stakeholders. Section C.1 in the appendix provides more information about this survey and its questions. In a second survey, we ask the stakeholders to rate the relevance of their answers provided to the first survey. This way, each stakeholder evaluates his or her opinion based on the answers of his or her peers. We provide the complete results of the second survey in section C.2 in the appendix. The rated relevance of the answers to the Delphi survey approximates the collective opinion of the panel of surgery scheduling stakeholders.

The Delphi study was supplemented with individual interviews discussed in sections 3.2.2 and 3.2.3 in order to obtain case-specific performance factors.

5.1.2 Delphi study results

We designed one question in the Delphi study specifically to obtain the relevant scheduling performance factors. Figure 5.1 displays the rated relevance of answers provided to this question: 'What constitutes a good or bad operating room schedule?'. The stakeholders rated these answers to have a large positive, positive, neutral, negative, or large negative effect.

Additionally, the interviews with scheduling stakeholders for the case of cardiothoracic scheduling identified the number of surgeries that are postponed last-minute as a relevant performance factor. Recall that in the case of cardiothoracic surgery, last-minute postponements are a common problem. These postponements are surgeries that have to be rescheduled within 24 hours of their scheduled surgery time. In the case of orthopedic surgery scheduling, however, surgeries rarely need to be postponed this way.

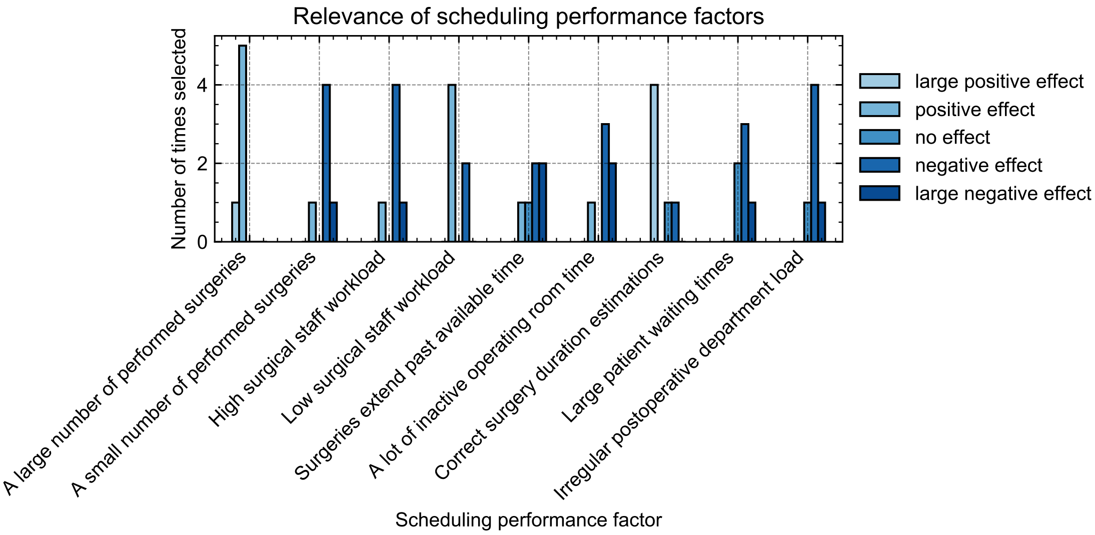


Figure 5.1: The rated relevance of the performance factors identified by the OR scheduling stakeholder panel.

5.1.3 Relevant scheduling performance factors

To translate the scheduling requirements resulting from the Delphi study into scheduling performance measures we need to identify the performance factors considered relevant by the stakeholders. The results of the Delphi study question presented in Figure 5.1 propose a set of performance factors which can be used to evaluate schedules. A large *number of performed surgeries* is considered to have a positive effect on the schedule performance whereas a small *number of performed surgeries* has a negative effect. In contrast to the number of performed surgeries, a high *staff workload* has a negative effect on the scheduling performance, whereas a smaller *workload* has a positive effect. *Correctly estimated surgery durations* are considered to have a positive effect, while the *time surgeries exceed the available time slots* has a negative effect. Moreover, the *time in which the OR is not in use* has a negative effect on the schedule performance. Similarly, larger *patient waiting times* and *irregular postoperative department loads* are associated with a negative effect on scheduling performance. Note that *patient waiting time* is ill-defined and can refer to multiple things. *Patient waiting time* can indicate the time that patients have to wait for surgery in the queue (a scope of potentially months) and the time which patients have to wait when their surgery is delayed (a scope of hours).

Not all of the aforementioned performance factors are relevant to the scope of the scheduling problem studied in this research. We identify distinguish following performance factors:

1. *Number of performed surgeries*: This factor is not relevant in our scope. Recall that the proposed method schedules patients within the time slots in which they actually received surgery. This means the same number of surgeries is performed using our proposed method.

2. *Staff workload*: Again, this factor is not relevant to our scope as the same number of surgeries are performed each day.

3. *Estimation of surgery durations*: This factor is affected by estimating the durations from surgery clusters. However, we do not evaluate this as a schedule performance measure. Rather, Section 9.3 discusses the duration estimation performance of our method separately.

4. *Time exceeding time slots*: This factor is affected by more efficiently scheduling the surgeries that took place and is thus relevant in our scope.

5. *Unused time in time slots*: Similarly to the time exceeding the time slots this factor is relevant in our scope.

6. *Patient waiting time*: We distinguished two levels of patient waiting time. The first, denoting the time the patients are waiting for surgery at home, is not changed with our problem scope. The second level of patient waiting time, the time the patient is waiting while admitted in the hospital, is relevant.

7. *Postoperative patient load*: As patients are generally admitted to postoperative departments for multiple days, this factor is not relevant to our scope. By scheduling the surgeries on the same day they were scheduled, we do not change the daily load to postoperative departments.

8. *Last-minute postponements*: By more efficiently scheduling the surgeries that are originally scheduled, last-minute postponements as a result of overtime can be avoided. Hence, this factor is relevant within our scope.

Table 5.1 lists the relevant abstract performance factors and the associated performance measures that will serve as a metric for the performance of schedules in terms of the relevant factors. These concrete performance measures are explained in Section 5.2.

Table 5.1: The performance factors and corresponding performance measures relevant to the scheduling scope of this thesis.

Abstract performance factor	Concrete performance measures
4. Time exceeding time slots	overtime
5. Unused time in time slots	idle time, undertime, utilization
6. Patient waiting time	patient waiting time
8. Last minute postponements	number postponements, factor postponements

5.2 Concrete performance measures

The previous section defined the abstract relevant performance factors that determine the surgery scheduling performance within the problem scope considered in this research. This section translates these performance factors in concrete performance measures which can be computed serve as a metric for the performance of a two-week surgery schedule. Section 5.2.1 explains how a general model can be adapted to formulate the concrete measures. Section 5.2.2 discusses what performance measures are typically used for the factors described in the previous section and presents the concrete performance measures used for the rest of this thesis.

5.2.1 Scheduling performance model

Analogous to Khaniyev et al (2020) we use a surgery scheduling model with uncertain surgery durations [36]. In this scheduling model, we incorporate the fixed preparation time, used to prepare ORs for the next surgery, in the surgery duration for simplicity. The available time slots for surgery are defined in the master surgery schedule. This schedule contains for all rooms R, $r = 1, \ldots, R$ on every day d, which surgery time slots (TS) are available. We denote this formally with $MSS_{r,d} = \{TS_1, \ldots, TS_s\}$, where s is the last time slot available for surgery in room r on day d.

Every surgery time slot TS_i has a start time s_i, end time e_i, fixed moment of acceptable overtime o_i^{acc} and a set of scheduled surgeries (or treatments) T_i . The *acceptable overtime* defines which surgeries need to be postponed. If the expected end time of a surgery exceeds the acceptable overtime, it should be postponed. We thus define a surgery time slot with $TS_i = (s_i, e_i, o_i^{acc}, T_i)$. The set of surgeries T_i is an ordered set of m_i scheduled surgeries, such that $T_i = \{t_{i,1}, \ldots, t_{i,m_i}\}$.

To formulate the scheduling performance measures, we define an auxiliary surgery parameter: *planned surgery duration* $\delta_{i,j}$. The parameter $\delta_{i,j}$ (like the assigned surgery duration in [36]) represents the difference between the scheduled start time and end time of surgery $t_{i,j}$. The *planned surgery duration* $\delta_{i,j}$ contains the entire time the OR is scheduled to be in use for surgery $t_{i,j}$.

Finally, each surgery could start earlier than their original scheduled start time, so long as the patient is available at that point. The time patients are available before surgeries varies per hospital and department but is typically a fixed amount of time. We define this time as the *pre-surgery availability* t^{pre}.

The planned surgery duration allows us to formulate additional surgery attributes that can be used to evaluate schedules. Particularly the *planned* and *realized start and end times* of surgeries are defined this way. We define the *scheduled start time* of surgery $t_{i,j}$ as $\tau_{i,j}^{SS}$, the *actual start time* as $\tau_{i,j}^{AS}$, the *scheduled end time* as $\tau_{i,j}^{SE}$ and the *actual end time* as $\tau_{i,j}^{AE}$. Figure 5.2 provides an example of the realization of a planned schedule within this model. Assuming the first scheduled surgery starts at the start of the day, then by definition of the planned surgery duration, the scheduled start time of all surgeries in the daily schedule is:

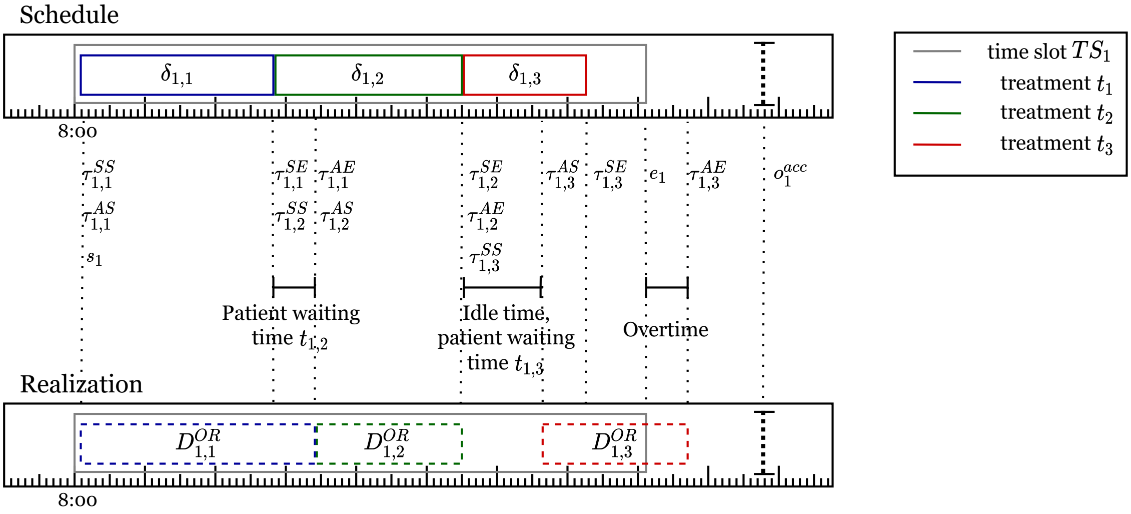


Figure 5.2: An example of the realization of a surgery schedule time slot TS_1 with uncertain surgery durations. The example includes the formulation of the model used to define the scheduling performance measures. Note that the idle time is equal to the patient waiting time of surgery t_3. We adapted this figure to this research problem from a figure by *Khaniyev et al* (2020) [36].

$$\tau^{SS}_{i,j} = \begin{cases} 0 & \text{if } j = 1 \\ \sum_{k=1}^{j-1} \delta_{i,k} & \text{if } j = 2, \ldots, m_i \end{cases} . \tag{5.1}$$

The time patients are available before surgery, allowing surgeries to start sooner than expectd, introduces for every surgery $t_{i,j}$ a possible start time: $\tau^{PS}_{i,j} = \tau^{SS}_{i,j} - t^{pre}$. Similarly to the scheduled start time, we define the scheduled end time as the sum of scheduled durations:

$$\tau^{SE}_{i,j} = \sum_{k=1}^{j} \delta_{i,k}. \tag{5.2}$$

Assuming the first surgery starts at the beginning of the day ($\tau^{AS}_i = 0$ if $j = 1$), subsequent surgeries start after the first surgery is completed and the possible scheduled start time has passed.

$$\tau^{AS}_{i,j} = \max\{\tau^{AE}_{i,j-1}, \tau^{PS}_{i,j}\}, \tag{5.3}$$

The duration of surgery $t_{i,j}$ was previously defined as a realization of a random variable. Hence, we define the actual end time as

$$\tau^{AE}_{i,j} = \tau^{AS}_{i,j} + D^{OR}_{i,j}, \tag{5.4}$$

where $D^{OR}_{i,j}$ denotes the realized surgery duration. Note that when $\tau^{AE}_{i,j} > o^{acc}_i$, the surgery is recognized as a surgery which should be postponed.

5.2.2 Performance measures selected from literature

The previous section provides a formal model to define the performance measures that will be used to evaluated surgery schedules. This section translates the performance factors identified in

Section 5.1.3 to performance measures found in literature and formalizes these with the model provided by the previous section. In the next chapters, we use these formal performance measures to quantify surgery scheduling performance.

The four scheduling performance factors identified in the previous section correspond to several specific performance criteria typically used in operating room scheduling. Samudara et al. (2016) developed an extensive overview of operating room scheduling performance measures in their literature review [8]. The first relevant performance factor identified with the Delphi study, the *time exceeding the time slots* available for surgery where operating rooms are in use, corresponds to *overtime.* Overtime is defined as the time from the end of the scheduled time slot until the end of the last surgery in the schedule.

We capture the *unused time in time slots* of the operating room, with the performance measure *idle time.* An additional performance measure, closely related to *idle time*, is also relevant to this performance factor. If the last surgery ends before the end of the time slot, we define this measure, the *undertime*, as the time between the end of the last surgery and the end of the time slot. Note that if the surgery would end after the end of the time slot, we would consider this time *overtime.* Furthermore, a final performance measure associated with the time the OR is in use within the available time slot is *utilization.* We define *utilization* as the time in the surgery time slots divided by the time the operating rooms are in use.

The third performance factor, *patient waiting time*, is directly defined as a performance measure. It denotes the time between the scheduled start time of the surgery to the actual start time.

Finally, the *number of postponements* is also trivially translated into a performance measure. This performance measure is interesting by itself, but the *factor of postponed surgeries* is of interest as well. When this factor is known for a large set of schedules, this approximates the probability of rescheduling a surgery. We define the performance measure *factor of postponements* as the fraction of the throughput which is postponed. This *throughput* is the number of surgeries performed in a schedule and can also be considered a performance measure. The identified performance measures are included in Table 5.1.

Throughput

In order to compare two schedules in terms of the number of surgeries that are performed in these schedules, we use the *throughput.* This measure is defined as the number of surgeries m_i performed in one time slot TS_i with a set of surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m_i}\}$. The *throughput* of a schedule of two weeks is the sum of the *throughput* of all surgery time slots available in those two weeks.

Idle time

To measure the time the OR is not in use but is available, we use *idle time.* This measure, relevant to the second performance factor, is the sum of the time in between surgeries and the time the OR is not in use before the end of the schedule. We define it as the difference between the end times of surgeries and the start time of the subsequent surgeries or, in the case of the final surgery of the day, the end of the schedule. Similarly, when the first surgery begins after the start of the schedule, the difference between the actual starting time and the start of the schedule is *idle time.* Hence, we define the *idle time* in one time slot $TS_i i$ in which the set of surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m}\}$ are performed as follows:

$$\text{idle time}_i = \sum_{j=2}^{m_i} (\tau_{i,j}^{AS} - \tau_{i,j-1}^{AE}) + \max\{0, \tau_{i,1}^{AS} - s_i\} + \max\{0, e_i - \tau_{i,m}^{AE}\}. \tag{5.5}$$

The *idle time* of a schedule of two weeks is the sum of the *idle time* of all surgery time slots in those weeks.

Patient waiting time

We define the *patient waiting time* as the difference between the scheduled start time (when the patient expected the surgery to start) and the actual start time. For one surgery time slot $TS_i i$ in which the set of surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m_i}\}$, we formalize the *patient waiting time* as:

$$\text{patient waiting time}_i = \sum_{j=1}^{m_i} \max\{0, \tau_{i,j}^{AS} - \tau_{i,j}^{SS}\}. \tag{5.6}$$

For a surgery schedule of two weeks, we define the *patient waiting time* as the sum of the patient waiting time of all surgeries in those two weeks.

Over-/Under- time

We define *overtime* as the time an operating room is in use outside after working hours. Similarly, *undertime* is the time the operating room is not in use between the last surgery and the end of the surgery time slot. We define both of these performance measures as the difference between the end of the last surgery and the end of the surgery time slot. When a surgery exceeds the time available to that time slot, we consider the access time *overtime.* Conversely, if the surgery ends before the surgery time slot, the time between the last surgery and the end of the time slot is *undertime.* For a single time slot TS_i in which the surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m_i}\}$ are performed, we formulate the *overtime* and *undertime* as:

$$\text{overtime}_i = \max\{0, \tau_{i,m_i}^{AE} - e_i\}, \tag{5.7}$$

$$\text{undertime}_i = \max\{0, e_i - \tau_{i,m_i}^{AE}\}. \tag{5.8}$$

Overtime can happen in different operating rooms and time slots, while in other operating rooms or time slots undertime occurs. To aggregate these metrics for a surgery schedule, we define the schedule *overtime* and *undertime* as the sum of *overtime* and *undertime* observed in all surgery time slots available in that schedule.

Note that this definition allows a schedule to have both over and *undertime* at the same time. In contrast, a single surgery time slot has either *undertime* or *overtime.* Since we define the *over* and *undertime* of a schedule as the sum of the *over* and *undertime* of the time slots in that schedule respectively, if some time slots have *overtime* and some have *undertime*, both of these performance measures are positive for the schedule.

Utilization

We denote the fraction of time that is available in a time slot that is actually used for surgery by the *utilization.* We define this performance measure as the time used for surgeries in a surgery time slot divided by the time available in that time slot. Since we aim to measure if the available time is used efficiently, we consider the overtime not part of the time the operating room is in use within the time slot. Furthermore, this ensures the utilization cannot exceed is in the range [0,1]. If we would consider the total time the OR was in use, a schedule with a lot of *overtime* and *idle time* could result in the same *utilization* as the same schedule without *idle time.* For a time slot TS_i, with surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m_i}\}$ and end time e_i, we define the *utilization* as:

$$D^{OR}_{i,j} = \begin{cases} \tau^{AE}_{i,j} - \tau^{AS}_{i,j} & \text{if } \tau^{AE}_{i,j} \leq e_1 \\ e_i - \tau^{AS}_{i,j} & \text{if } \tau^{AE}_{i,j} > e_1 \text{ and } \tau^{AS}_{i,j} \leq e_1 \\ 0 & \text{if } \tau^{AS}_{i,j} > e_1 \end{cases} \tag{5.9}$$

$$\text{utilization}_i = \frac{\sum_{j=1}^{m_i} D^{OR}_{i,j}}{e_i - s_i}. \tag{5.10}$$

where ($D^{OR}_{i,j}$) denotes the duration of a single surgery within the time slot. To determine the *utilization* of a schedule of two weeks of surgery, the total time used for surgery and available time in those two weeks need to be divided. For a surgery schedule of two weeks $MSS = \{TS_1, \ldots, TS_s\}$, with s time slots TS_i in which m_i surgeries are performed, we define the total *utilization* as follows:

$$\text{total utilization} = \frac{\sum_{i=1}^{s} \sum_{j=1}^{m_i} D^{S}_{i,j}}{\sum_{i=1}^{s}(e_i - s_i)}. \tag{5.11}$$

Note that the number of surgeries in each time slot m_i is not fixed and may very per time slot.

Number postponements

The final performance measures used in this research are the *number of postponements* and the *factor of postponements*. We define the *number of postponements* as the number of surgeries in a surgery time slot for which the expected end time $\tau^{AE}_{i,j}$ exceeds the maximal *acceptable overtime* o^{acc}_i of that time slot. The *factor of postponements* of a surgery time slot denotes the fraction of surgeries performed in a surgery time slot that are considered to be postponements. We hence formulate the performance measures for a time slot i, with surgeries $T_i = \{t_{i,1}, \ldots, t_{i,m}\}$ as follows:

$$\text{postponements}_i = \sum_{j=1}^{m_i} \mathbb{1}\{\tau^{AE}_{i,j} > o^{acc}_i\} \tag{5.12}$$

$$\text{factor postponements}_i = \frac{\text{number postponements}_i}{\text{throughput}_i} = \frac{\sum_{j=1}^{m_i} \mathbb{1}\{\tau^{AE}_{i,j} > o^{acc}_i\}}{m_i}. \tag{5.13}$$

The *number of postponements* of two weeks of surgery schedule is the sum of the *number of postponements* of all time slots within those weeks. The total *factor of postponements* of two weeks of schedules however cannot be aggregated this easily. The total factor of postponements is the total *number of postponements* of those two weeks divided by the *throughput* of those weeks. Consider a surgery schedule of two weeks $MSS = \{TS_1, \ldots, TS_s\}$, with s surgery time slots TS_i in which m_i surgeries are performed. Again, the number of surgeries in each time slot m_i is not fixed and may very per time slot. The total *factor of postponed surgeries* in that case is:

$$\text{total factor postponements} = \frac{\text{total postponements}}{\text{total throughput}} = \frac{\sum_{i=1}^{s} \sum_{j=1}^{m_i} \mathbb{1}\{\tau^{AE}_{i,j} > o^{acc}_i\}}{\sum_{i=1}^{s} m_i}. \tag{5.14}$$

The performance measures discussed above provide scheduling objectives with which schedules can be compared and evaluated. The measures are based on performance factors identified in cooperation with hospital stakeholders. Furthermore, we formalized these performance measures in order to be able to compute unambiguous scheduling measures in the evaluation of the scheduling method. In the following chapters, we use these scheduling measures to compare schedule performances.

Chapter 6

Clustering surgeries based on scheduling characteristics

To leverage clusters during scheduling, the clusters we use to provide expected surgery characteristics need to discriminate the surgeries correctly. In the previous section, the scheduling performance measures have been defined. With these objectives in place, the rest of this research focuses on using clusters in scheduling and evaluating the scheduling. Chapter 4 explains why we chose to use clustering to identify groups of surgeries with similar scheduling characteristics. Identifying patients groups that should be scheduled similarly corresponds to the research sub-goal *SG 2* discussed in section 3.5. This chapter tackles this research subproblem by discriminating surgeries in clusters with distinctive scheduling properties. Figure 6.1 shows the general scheduling method and highlights the clustering method discussed in this chapter.

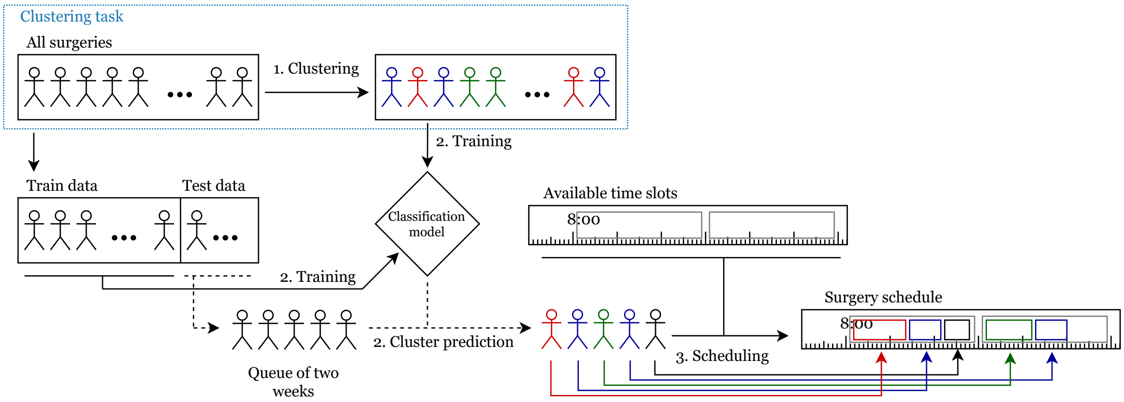


Figure 6.1: The general scheduling method with the highlighted clustering task.

In order to find the clusters that can be used in scheduling, first the relevant surgery scheduling characteristics are identified in Section 6.1. Next, the first step of the clustering method, manually creating the initial clusters required for scheduling constraints, is discussed in Section 6.2. We describe the unsupervised clustering method, the second step of the general clustering method, in Section 6.3. Next, Section 6.4 develops a measure of distance between surgeries that is required for the clustering method. In Section 6.5 we discuss how the clustering method is used to obtain a suitable number of subclusters per initial cluster. Finally, Section 6.6 discusses how we select the meaningful clusters from the clustering obtained with the proposed clustering method.

6.1 Clustering surgery characteristics

In order to obtain a clustering that can be used during scheduling, the surgery characteristics on which surgeries are clustered need to be defined. The scheduling characteristics used for this clustering can be used to schedule surgeries. In section 3.3 we identified the scheduling challenges and requirements for both hospital cases identified in this research. In this section, we show how to translate these requirements into clustering features.

First, we identify which requirements constitute scheduling constraints that determine whether or not a schedule is feasible. If surgeries can only be performed with specific resources, that are known before scheduling, we should distinguish these clusters before performing a data-driven clustering. This *initial clustering* ensures that surgeries are not scheduled as surgery of the wrong cluster. We explain this manual clustering and how to identify the initial cluster features in Section 6.2. In this section, we show how to identify the features that can be used to efficiently schedule the surgeries.

In order to find the cluster characteristics, we use the requirements of each scheduling case that do not determine schedule feasibility but determine scheduling performance. For both hospital cases, we find the same features to be relevant. As discussed in 3.3, in the case of orthopedic and cardiothoracic scheduling cases, the operating room capacity and postoperative ward capacity need to be managed efficiently. The rest of the requirements, including the availability of staff and specific operating rooms, determine whether or not a schedule is feasible. By identifying which requirements are used to improve scheduling performance, rather than ensure schedule feasibility, we find out which requirements can be translated into clustering characteristics. Next, we select the features that can be used to fulfill these requirements from the available data for a hospital case. We explain the available data of the orthopedic and cardiothoracic surgery scheduling cases in Section 3.4.

To adapt the proposed method to a new hospital case, we need to identify which requirements need to be tackled with scheduling and what data we can use to do so. For orthopedic and cardiothoracic surgery we identify the same relevant features and Table 6.1 lists these scheduling characteristics.

Table 6.1: The scheduling characteristics used to cluster orthopedic and cardiothoracic surgeries. The requirements refer to the resource management constraints identified as requirements in Section 3.3

Requirement	Scheduling feature
1. Efficient OR management	duration
3. Postopertave ward capacity management	postoperative length of stay postoperative department

6.2 Initial clustering

In order to find meaningful surgery clusters, the next step is to manually allocate surgeries to initial clusters associated with strict scheduling constraints. The scheduling requirements identified in Section 3.5 correspond to either strict scheduling requirements or features that are allowed to have some estimation error. As discussed in the previous section, strict scheduling requirements may not be confused during scheduling. Some surgery types might, for example, only be possible in a subset of the available ORs. In that case, surgeries with different surgery types can not be placed in the same cluster. To distinguish surgeries with different strict scheduling requirements and surgeries with different flexible characteristics, this research performs the clustering in two steps. First, we manually perform an initial clustering by assigning surgeries with the same strict scheduling requirements to separate clusters. Next, we cluster the set of surgeries in each initial cluster based on the flexible scheduling characteristics in the unsupervised machine learning fashion (discussed in Section 2.1). Figure 6.2 displays the two stages of the clustering method.

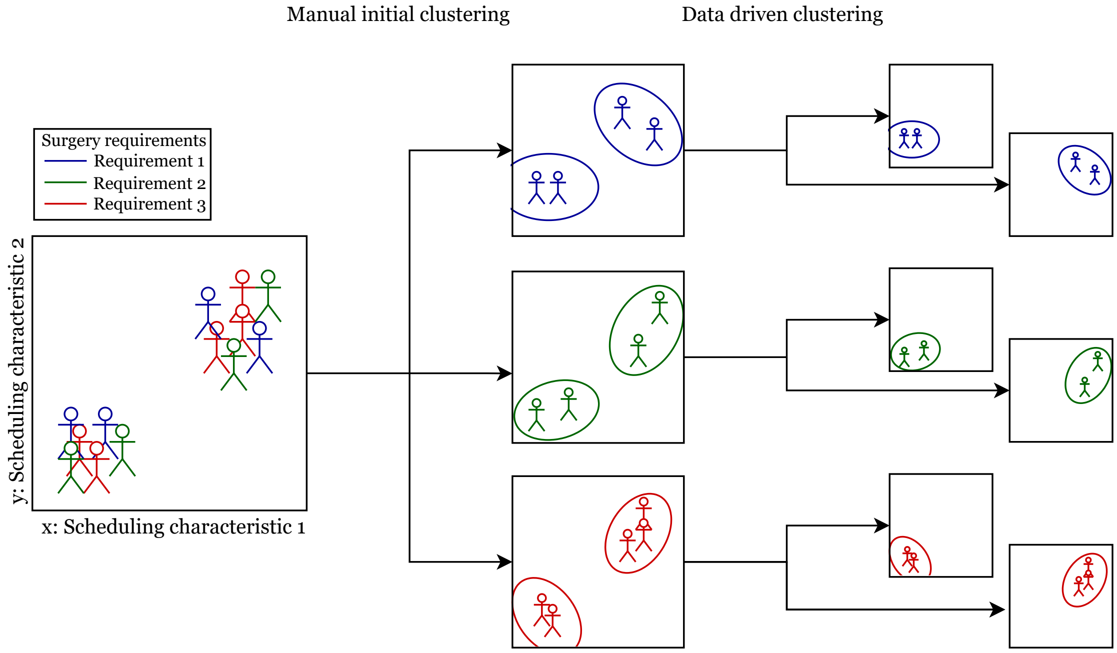

Manual initial clustering
Data driven clustering
Surgery requirements
Requirement 1
Requirement 2
Requirement 3
y: Scheduling characteristic 2
x: Scheduling characteristic 1

Figure 6.2: The clustering method used to categorize surgeries both on surgery requirements and scheduling characteristics. Note that we first distinguish surgeries with different requirements by manually assigning them to different initial clusters. Subsequently, we learn the clusters in other scheduling characteristic domains in an unsupervised fashion from the data.

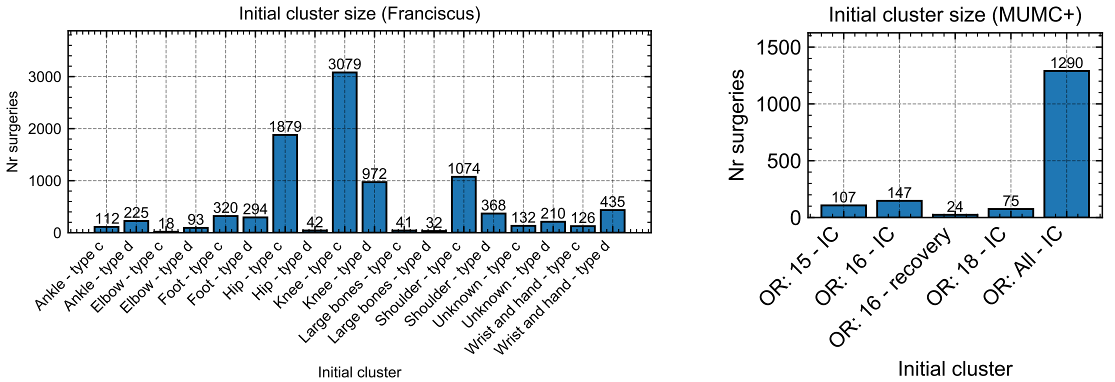
To assign surgeries to the initial clusters, we first identify which scheduling requirements correspond to strict constraints and obtain the surgery features determining these constraints. We explain the scheduling requirements in Section 3.5 and discuss the available data for the investigated hospital cases in Section 3.4. Table 6.2 lists the identified initial cluster features for orthopedic surgery scheduling. In Table 6.3 we present the identified initial cluster features for cardiothoracic surgery scheduling. Adapting this clustering method to a new hospital case would require determining a similar mapping of scheduling constraints to requirements and available features. Next, we manually assign the surgeries with the categorical features described in these tables to *initial clusters* to perform the first part of our clustering method.

Table 6.2: The initial clustering features used to distinguish orthopedic surgeries with different strict scheduling constraints. The requirements refer to the resource management constraints identified as requirements in section 3.3

Requirement	Initial cluster feature	Possible values
2. Surgical staff availability	subspecialism	wrist and hand, shoulder, hip, knee, large bones, foot, elbow, ankle or unknown
3. Postoperative ward capacity management	surgery type	clinical (c), daily admission (d)

Table 6.3: The initial clustering features used to distinguish cardiothoracic surgeries with different strict scheduling constraints. The requirements refer to the resource management constraints identified as requirements in section 3.3

Requirement	Initial cluster feature	Possible values
2. OR availability	possible OR	OR: 15, OR: 16, OR: 18 or all OR's
3. Postoperative ward capacity	intensive care required	yes (IC), no (recovery)



(a) Initial cluster size orthopedic surgery

(b) Initial cluster size cardiothoracic surgery

Figure 6.3: The size of the manually assigned initial clusters of surgeries that happened in 2018 or 2019.

Figure 6.3 depicts the sizes of the initial clusters identified using this method. We show the initial cluster sizes of the orthopedic surgeries in Figure 6.3a, whereas Figure 6.3b shows the sizes of the cardiothoracic surgery clusters. In these figures, the number of surgeries in each initial cluster is plotted as a bar.

These initial clusters found both hospital cases effectively capture the information required to adhere to the scheduling constraints identified in Section 3.3. We associate every initial cluster of orthopedic surgeries with a subspecialism and admission type and the initial cardiothoracic surgery clusters with the available operating rooms and ICU requirements.

By assigning the surgeries to the initial clusters described above, we find clusters with interesting sizes. Particularly, the distribution of surgeries among the initial clusters is skewed. The orthopedics department performs a lot of knee and hip surgeries, explaining the larger sizes of the clusters. Furthermore, most of these hip surgeries are clinical as hip surgeries are severe and generally require overnight care. This is reflected by the size of the clinical hip surgery cluster compared to the size of the cluster hip surgeries with daily admissions. For cardiothoracic surgery at MUMC+ in Figure 6.3b, it stands out that nearly all surgeries require ICU capacity. Interestingly, just one initial cluster with patients that recover from surgery in the recovery ward instead of the ICU is observed. Since the only surgeries that do not require intensive care after surgery are femoral Transcatheter Aortic Valve Implantations (TAVI's) and these surgeries can only occur in operating room 16, this result is to be expected.

6.3 Surgery clustering

In the previous sections of this chapter, we construct the initial clusters and explain the scheduling features that are relevant to group surgeries. In order to arrive at the clustering that can be used during scheduling, the surgeries of the initial clusters need to be allocated to subclusters with distinct scheduling features. This section discusses the agglomerative clustering method that performs this subclustering.

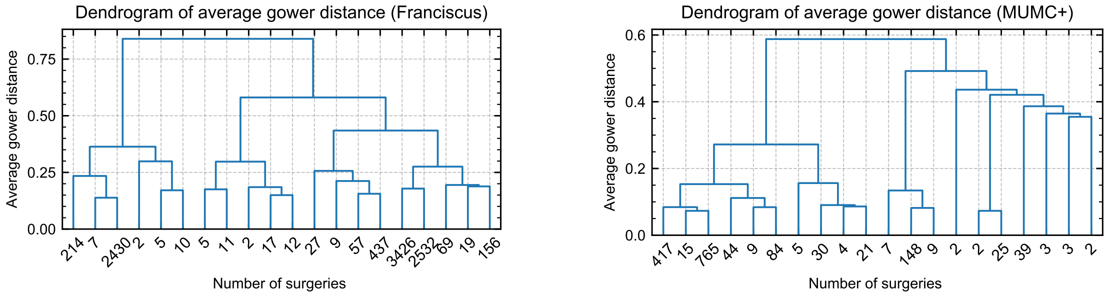
Agglomerative clustering is a hierarchical clustering method that greedily assigns samples to clusters. We discuss this method more extensively in Section 2.1.2. In the case of surgery clustering, it constructs a cluster hierarchy in a bottom-up fashion. First, we assign all surgeries to their own subcluster. Next, we merge the two most similar subclusters until a hierarchy of surgery subclusters is formed. The constructed subcluster hierarchy can be used to determine the initial surgery subclusters. Depending on the number of clusters that need to be identified, we traverse the hierarchical subclustering tree until only the desired number of subclusters can still be merged. These subclusters make up the eventual clusters that can be used in scheduling.

The similarity of two subclusters is defined using a linkage criterion. In this research, we use the average linkage criterion. A linkage criterion is a function determining the distance (or similarity) of two sets of observations. The average linkage criterion is the average pairwise distance between all observations of two subclusters. Consider a general pairwise distance $d(a, b)$ between two observations a and b. The average linkage L^{avg} between subcluster $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_m\}$ is defined as $L^{avg}(A, B) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} d(a_i, b_j)}{nm}$. Hence, in every step of the agglomerative clustering used in this research, we merge the two subclusters with the smallest average pairwise distance. We chose to use the average linkage for this method since it serves as a compromise between the sensitivity to outliers of complete linkage and the tendency of single linkage clustering to form counter intuitive chain-like cluster structures [37]. The pairwise distance $d(a, b)$ is not trivial and the distance measure used in this thesis is discussed in the next section. We discuss the results of clustering surgeries with agglomerative clustering after explaining the employed distance measure.

6.4 Surgery distance

As discussed in the previous section, in order to find clusters of similar surgeries, we require a measure of surgery similarity (surgery distance). The concept of similarity of two surgeries is not trivial. In the field of unsupervised machine learning and clustering, a distance measure is a function quantifying the similarity between two objects [38]. This distance measure thus is a function of two sets of features and indicates the distance between the objects to which these features belong. In the case of surgery clustering considered in this research, the distance measure denotes the similarity based on the set of relevant features previously defined in Section 6.1: surgery duration, postoperative department, and length of stay. Note that, in contrast to surgery duration and length of stay (which are continuous values), the postoperative department is a nominal attribute. Distance measures are typically used to find reasonable distances in applications with continuous values. Heterogeneous feature value settings, in which features of both continuous and nominal values are considered, pose an additional challenge and are a field of ongoing research [39]. This section discusses the heterogeneous distance measure employed to define similarity between surgery clusters based on scheduling characteristics in this research. Also, this section summarizes the clustering obtained with the agglomerative clustering method using this distance measure.

The heterogeneous distance measure used to measure the similarity of surgeries is called Gower distance. In Section 2.1.1 in the preliminaries, we explain the Gower distance in detail. Recall that the Gower distance is defined as an average of differences between individual features. For the continuous features surgery duration and postoperative length of stay, this difference is the interval scaled difference of the two observed values. The difference in the nominal feature postoperative department, however, is either 0 when the patient moves to the same department after surgery or 1 when the postoperative departments are different. Since both, the interval scaled differences of continuous variables and the binary similarity of nominal variables result in individual feature differences between 0 and 1, the Gower distance denotes the similarity between two surgeries in



(a) The Gower distance dendrogram of orthopedic surgeries.

(b) The Gower distance dendrogram of cardiothoracic surgeries.

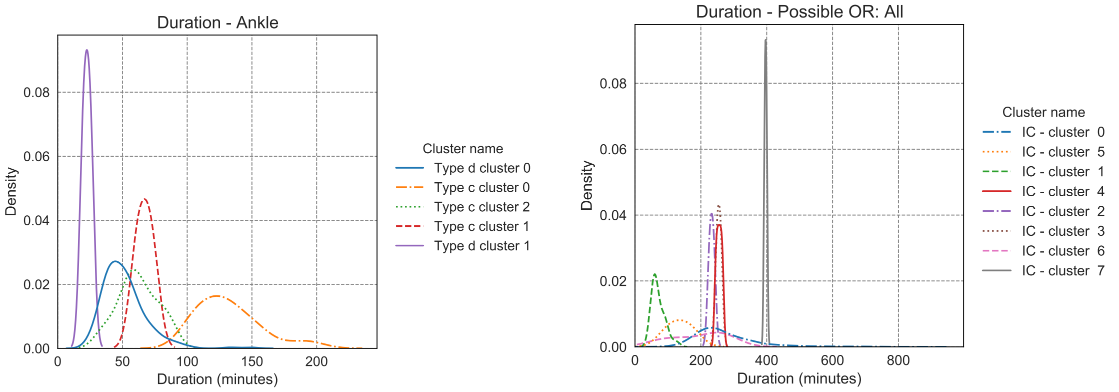
Figure 6.4: The average Gower distance dendrograms showing the hierarchical similarity between surgeries in 2018 and 2019. The size of the group of surgeries denoted by the leaves in the dendrogram is displayed on the x-axis.

that range as well.

Figures 6.4a and 6.4b display the hierarchical Gower distance similarity of the orthopedic and cardiothoracic surgeries, respectively. These dendrograms show the average Gower distance between subgroups in the total set of surgeries. The average distance between subgroups connected with horizontal lines is denoted on the y-axis, as the height of the horizontal line. Furthermore, the size of one subgroup is the sum of the size of all its subgroups. The size of the smallest level of subgroups in the dendrograms (the leaves) is provided on the x-axis.

These dendrograms allow us to discriminate surgeries in terms of the previously defined scheduling characteristics. Note that for orthopedic surgeries the initial split produces two subgroups with an average Gower distance of over 0.8. Approximately the same number of surgeries that are present in the subgroups found with this split are admitted to either the daily admission or orthopedic department after surgery. A similar observation stands out for cardiothoracic surgery. The initial split produces two subgroups, one with approximately 1400 surgeries and one that is much smaller with a lot of difference in its subgroups. This corresponds to the large number of patients who are admitted to the ICU after cardiothoracic surgery and the relatively small number of patients being admitted to various different departments. This suggests that the Gower distance is more sensitive to differences in categorical features compared to differences in numerical features. That observation is supported by the fact that a difference in a categorical feature results in an individual feature distance of 1, which is the maximal distance possible for numerical features. As discussed further in 10.2, this, unfortunately, results in small clusters for infrequent categorical features suggesting an adjusted distance measure might help find better clusters.

To find the clusters which will be used in scheduling, we use the Gower distance with average linkage of subclusters in the clustering method. Figure 6.4 shows this average Gower distance subcluster hierarchy of the entire dataset (instead of a particular initial cluster). Figure 6.5 shows the distribution of surgery durations among the clusters identified in initial clusters using the agglomerative clustering method described above. In Figure 6.5a, we display the distribution of all clusters found in the initial clusters corresponding to surgeries with subspecialism ‘Ankle’. This contains 3 clusters found in the initial cluster of surgeries with the clinical admission type and 2 clusters found in the initial cluster of surgeries with the daily admission type. Figure 6.5b displays the distribution of durations of surgeries in the 8 clusters identified in the initial cluster of surgeries that are allowed in all operating rooms and are expected to require intensive care after surgery. These figures show these durations as Gaussian kernel density estimations of the empirical distributions. We use kernel density estimations to provide interpretable feature distributions in the form of continuous probability density curves. Figure 6.5 only shows the duration for a limited set of initial clusters. We evaluate all features used in scheduling in Section 9.1. Furthermore, the full set of cluster characteristics found using the clustering method used during scheduling is provided in Appendix D.



(a) Duration distribution of ankle subclusters.

(b) Duration distribution of subclusters of surgeries possible in all ORs.

Figure 6.5: The density estimations of the duration distributions of subclusters identified using agglomerative clustering with average Gower distance linkage of example initial surgery clusters. Figure 6.5a shows the duration of the initial clusters *Ankle - type c* and *Ankle - type d*. Figure 6.5b shows the duration of the initial cluster *Possible OR: All - IC*.

Figure 6.5 shows that agglomerative clustering generates surgery clusters with different scheduling characteristics. Specifically, it shows that surgery clusters with distinct surgery durations are found. For example, in Figure 6.5a we find two clusters in the daily admission initial cluster with ankle surgeries. The first cluster (Type d cluster 0) takes on average approximately 50 minutes whereas the second cluster (Type d cluster 1) takes on average approximately 20 minutes. Furthermore, we observe a larger variability in duration for the first cluster. Similarly, in Figure 6.5b we find various distinct clusters in terms of surgery duration for cardiothoracic surgeries. Note that one cluster in particular (IC - cluster 7) seems to have very small variability. The small variability is an artifact of the uneven cluster sizes. This cluster is limited in size and hence also has smaller variability in terms of surgery duration, resulting in a distribution with high kurtosis. Since this shows the potential of discriminating surgeries in clusters with different scheduling characteristics, the next section discusses how we can determine a suitable number of clusters to be constructed using agglomerative clustering.

6.5 Selecting the number of clusters

We computationally determine the number of clusters that result in a suitable clustering per initial cluster. The agglomerative clustering method, discussed in the previous section, produces suitable clusters (with distinct scheduling characteristics). However, it does this for a predetermined number of clusters. The number of clusters that best distinguishes the set of surgeries in distinct clusters varies. For every initial cluster, we need to establish a suitable number of subclusters to obtain the actual clustering which is used during scheduling. This section explains the method with which the suitable number of subclusters is determined.

To determine an appropriate number of clusters to allocate the surgeries to with agglomerative clustering, we employ the silhouette method. This method uses a clustering performance measure, the silhouette score, to evaluate the clustering that results from a specific number of target clusters. By testing a range of number of clusters and selecting the number of clusters which results in the best (highest) silhouette score, we determine the number of clusters to be used in scheduling. The silhouette score is the mean silhouette value of all the data samples that are being clustered. The silhouette value represents the similarity of one surgery to the other surgeries in its own cluster versus the similarity to the surgeries in other clusters. We explain the silhouette value in more detail in Section 2.1.3. The similarity used in this silhouette analysis is the same pairwise Gower distance used in the clustering method and explained in Section 6.4.

The silhouette method is evaluated formally in section 9.1. We performed the silhouette method for every initial cluster, computing its silhouette score for a multiple number of cluster. The number of clusters resulting in the highest silhouette score is used to construct the clusters for that initial cluster to be used in our scheduling method.

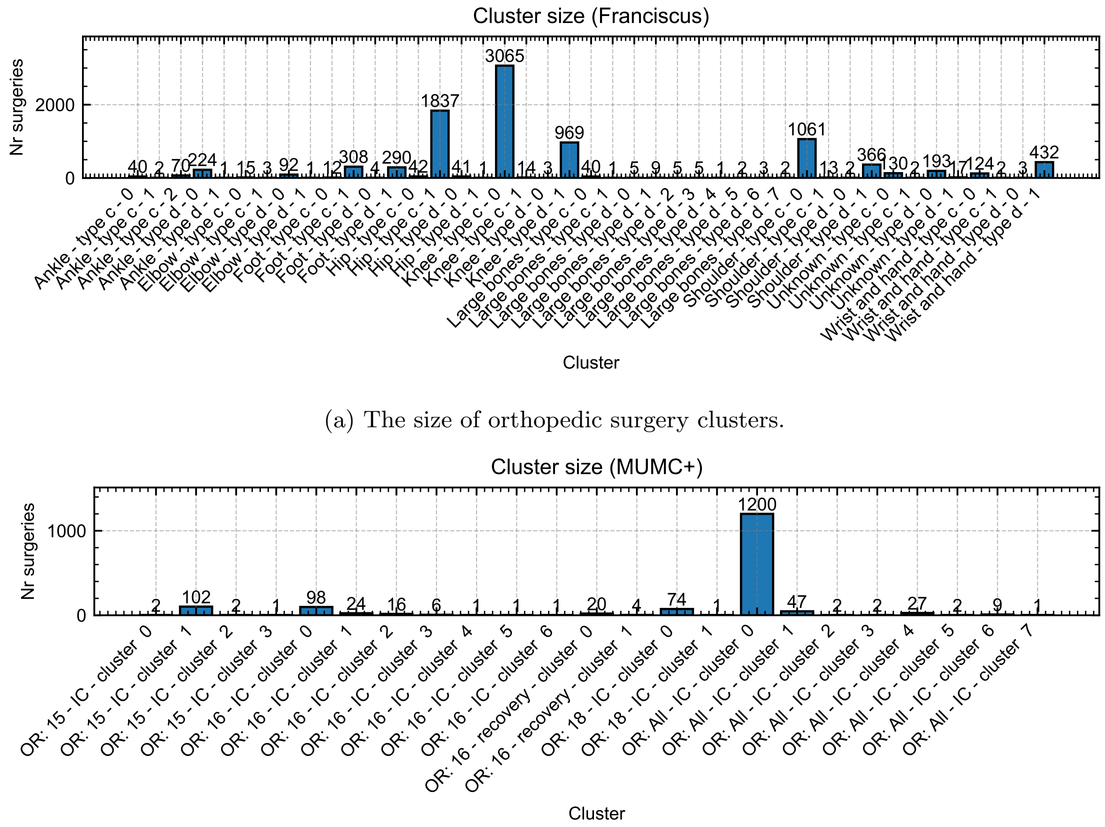
Table 6.4: The number of clusters resulting in the best silhouette coefficient.

Franciscus		**MUMC+**	
Initial cluster	Best nr clusters	Initial cluster	Best nr clusters
Ankle - type c	3	OR: 15 - IC	4
Ankle - type d	2	OR: 16 - IC	7
Elbow - type c	2	OR: 16 - recovery	2
Elbow - type d	2	OR: 18 - IC	2
Foot - type c	2	OR: ALL - IC	8
Foot - type d	2		
Hip - type c	2		
Hip - type d	2		
Knee - type c	2		
Knee - type d	2		
Large bones - type c	2		
Large bones - type d	8		
Shoulder - type c	2		
Shoulder - type d	2		
Unknown - type c	2		
Unknown - type d	2		
Wrist and hand - type c	2		
Wrist and hand - type d	2		

Table 6.4 lists the best number of clusters identified per initial cluster. Note that for most initial clusters of orthopedic surgeries we identify the minimal number of 2 clusters to result in the highest silhouette score.

Figure 6.6 shows the size of the surgery clusters following from the clustering method described in this section using the best number of clusters identified with the silhouette method. Note that the size of the subclusters is very skewed. In most initial clusters a subcluster is found with the bulk of all surgeries performed in that subspecialism, whereas the other subclusters contain a relatively small number of surgeries.

The cluster sizes resulting from this clustering method are skewed. The smallest clusters introduce problems with classification and scheduling. Specifically, small clusters are less likely to be distributed evenly over train/test splits and have few examples to train a classification model with. Furthermore, the sample sizes of the distributions of scheduling characteristics for these small clusters might be too small to produce accurate estimations. The small clusters suggest that our clustering method is too sensitive and a smaller number of clusters per initial subcluster could be more appropriate. Additionally, the tendency of categorical features to outweigh numerical clustering features in the Gower distance measure might promote these skewed cluster sizes. The problems with small subcluster sizes are discussed further in section 10.2. To address this problem



(a) The size of orthopedic surgery clusters.

(b) The size of cardiothoracic surgery clusters.

Figure 6.6: The size of subclusters found using agglomerative clustering with average Gower distance linkage. The number of subclusters per initial cluster is determined using the silhouette method.

of small clusters we determine which clusters are meaningful in the following section, before using these clusters in scheduling.

6.6 Selecting the meaningful clusters

In order to use the clusters identified with the clustering method discussed in this chapter, we need to make sure we group the surgeries in meaningful clusters. This means the clusters actually represent groups of surgeries with distinct scheduling characteristics and contain enough surgeries to represent a different group of surgeries. In this section, we manually compare the clusters found in the previous section, to identify the ones that are meaningful for scheduling.

To identify label a cluster as meaningful, we assert two things:

1. The cluster is distinct from the other surgery clusters. We determine this by manually inspecting the scheduling characteristic distributions of the clusters belonging to the same initial cluster. A cluster from a different initial cluster has different scheduling constraints so we always considered those distinct. Two clusters from the same initial cluster are distinct when the distribution of one of its scheduling features is different.
2. The cluster contains enough surgeries to represent a different group of surgeries. To prevent the method from being too sensitive to outliers, we only consider the surgeries with at least 5 surgeries to be 'meaningful'.

In Section 9.1 we evaluate the clusters thoroughly and manually inspect whether or not the clusters have distinct scheduling characteristic distributions. We find that each discovered cluster in every initial cluster does have at least one distinct scheduling feature. However, as discussed in the previous section, some clusters contain a small number of surgeries. Selecting only the clusters with at least 5 surgeries results in the meaningful clusters listed in Table 6.5.

Table 6.5: The identified meaningful clusters for the orthopedic and cardiothroacic surgery scheduling cases.

Franciscus	**MUMC+**
Ankle - type c - 0	OR: 15 - IC - 1
Ankle - type c - 2	OR: 16 - IC - 0
Ankle - type d - 0	OR: 16 - IC - 1
Elbow - type c - 1	OR: 16 - IC - 2
Elbow - type d - 0	OR: 16 - IC - 3
Foot - type c - 0	OR: 16 - recovery - 0
Foot - type c - 1	OR: 16 - IC - 3
Foot - type d - 1	OR: 18 - IC - 0
Hip - type c - 0	OR: All - IC - 0
Hip - type c - 1	OR: All - IC - 1
Hip - type d - 0	OR: All - IC - 4
Knee - type c - 0	OR: All - IC - 6
Knee - type c - 1	
Knee - type d - 1	
Large bones - type c - 0	
Large bones - type d - 0	
Large bones - type d - 1	
Large bones - type d - 2	
Large bones - type d - 3	
Shoulder - type c - 0	

Continued on next page

Table 6.5 – *Continued from previous page*

Franciscus	**MUMC+**
Shoulder - type c - 1	
Shoulder - type d - 1	
Unknown - type c - 0	
Unknown - type d - 0	
Wrist and hand - type c - 0	
Wrist and hand - type d - 1	

Unfortunately, the clustering method results in small clusters that are unlikely to represent meaningful clusters. However, regardless of the cluster sizes, the clusters identified with this method have distinct scheduling characteristics. Hence, during the scheduling of the surgeries, these surgery clusters can be leveraged to infer scheduling characteristics. The next chapters will use the clustering developed in this chapter to improve the operating room scheduling. The identified meaningful clusters serve as class labels to train a prediction model. This model predicts for surgeries that are being scheduled into which scheduling cluster they belong. This prediction is explained in Chapter 7.

Chapter 7

Predicting the surgery clusters

The cluster to which a surgery that is being scheduled belongs is not known before scheduling. So, in order to leverage the cluster characteristics during scheduling, the cluster to which a surgery belongs has to be predicted. We introduced this sub-goal *SG 3*, classifying surgeries into clusters, in Section 3.5 and we address it in this chapter. Figure 7.1 illustrates the general scheduling method and highlights the sub-task of predicting the cluster to which surgeries belong.

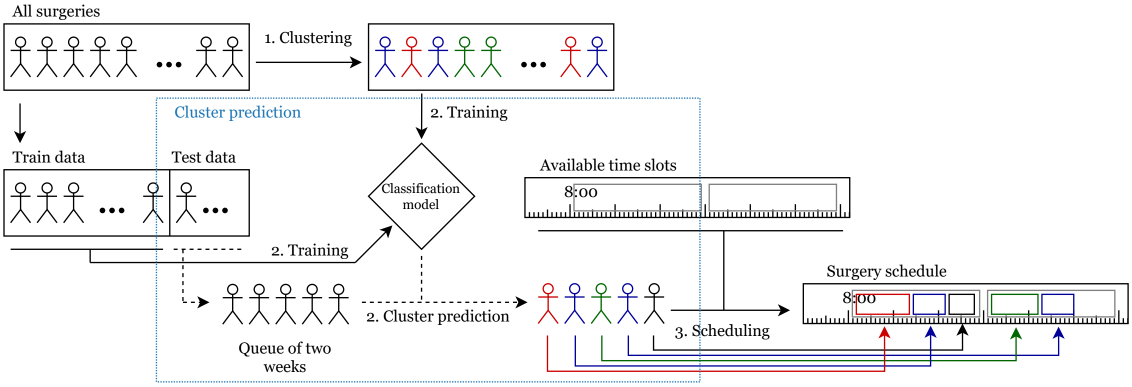


Figure 7.1: The general scheduling method with the highlighted cluster prediction task.

In the previous chapter, we discover the surgery clusters which can be used in scheduling. This chapter first discusses how we split the available data in datasets that can be used to train and evaluate an unbiased predictor in section 7.1. Next, we explain how surgery clusters can be predicted using a classification model in section 7.2. The classification model explained in that section uses some parameters which determine the performance of that model. In Section 7.3, we determine a suitable parameter setting for the classification model. Finally, Section 7.4 addresses the explainability of the classification model used to predict surgery clusters.

7.1 Cross-validation

In order to develop a prediction method that can predict surgery clusters in practice, we need to develop a model that generalizes to unseen data. To develop an unbiased estimator, we need to make sure we evaluate the performance of our prediction model on data that is not used in training. Additionally, we also experiment with model configurations during model training to identify which one results in the best model. Again, we need to evaluate each configuration on data that is not used during training of that configuration. To make sure we develop an unbiased prediction model, we employ a rolling cross-validation method. This section explains the used *cross-validation* method in the context of surgery cluster prediction.

Cross-validation is a machine learning approach we use to ensure a prediction model is accurate not only for data that is used in training but also for unseen data. The *rolling cross-validation* setup, used in this research splits the data into several partitions. First, the preprocessed data is split into two datasets: training and test data. The training data is used to fit and configure the prediction models. Next, the test data is used to evaluate whether or not these models generalize to observations withheld during training. No optimal train-test split data ratio is available, but research suggests dedicating 40-80% of the full dataset to training produces the best results [40]. Furthermore, for smaller sample sizes a larger portion of data reserved for training is recommended [40]. The data available for this research is limited to two years of surgeries, wherefore an 80/20 split is chosen (80% training, 20% testing data).

Additionally, we use *cross-validation* during hyperparameter configuration to estimate the generalized model performance and ensure we do not pick a set of hyperparameters optimal for a single test set. We use *rolling time series cross-validation* to maintain the temporal relationship in schedules. Since the surgery scheduling process is a dynamic process in hospitals, procedural changes might alter the relationships the prediction models are trained to capture. This makes the prediction models prone to *concept drift.* In order to be able to detect *concept drift* during the training of the prediction models, we use rolling time series cross-validation [41]. This *cross-validation* method partitions the training data multiple times in a training and test set and averages the performance over these multiple splits to obtain an unbiased performance measure. Each split, the training data is extended with the previous test data and the next observations are used as the new test data. When, during training, the model performance suddenly drops in the k'th *cross-validation* split, this points to a change in the surgery process. The *cross-validation* setup splitting the data to prevent overfitting is depicted in figure 7.2.

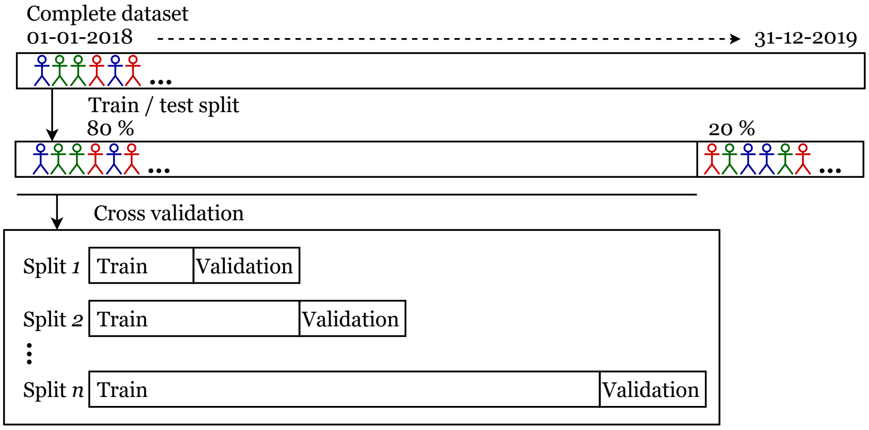


Figure 7.2: The train/test and cross-validation splitting used to minimize overfitting and facilitate the evaluation of the prediction method. Note there is a temporal relationship in the data which is maintained in the 80/20 train/test split and the expanding window cross-validation method.

The *cross-validation* method displayed in Figure 7.2 is used to evaluate the classification model discussed in this section twice. We use this cross-validation during the parameter configuration discussed in Section 7.3. Furthermore, the unbiased estimator trained using this *cross-validation* is developed in section 7.2. The performance of the prediction model at different stages of the cross-validation method is discussed in Section 9.2.

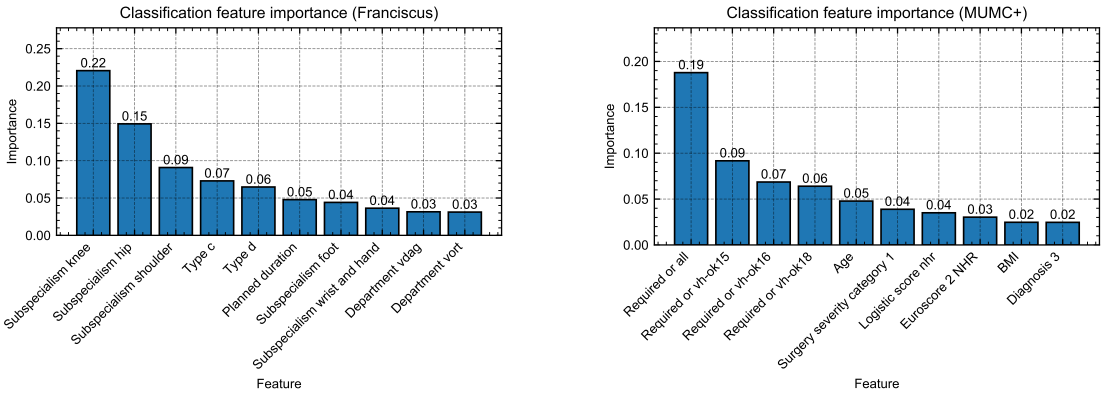
7.2 Random forest classification

To estimate surgery characteristics from clusters during scheduling, the cluster of a surgery that is to be scheduled has to be predicted. Classification is the problem of discovering to which set of observations a new observation belongs. In the context of surgery scheduling based on surgery schedules, this is the problem of identifying the cluster to which a surgery belongs. In order to develop a method that is able to leverage surgery clusters in operating room scheduling, we developed a classification model that assigns surgeries to the clusters identified in the previous chapter. This classification is based on the set of features that is available a priori (before scheduling). This section discusses the random forest classification model that is used to predict the cluster that is used to estimate surgery characteristics.

To classify surgeries into clusters, we use random forest classification. This method is a supervised learning algorithm that builds an ensemble of decision trees that predict a class for some feature input [22]. In section 2.2.1, we explain random forest classification in more detail. Each decision tree in the random forest ensemble is a decision tree classifier and the most frequently predicted output class label is the output of the entire ensemble. Decision trees are constructed top-down, by recursively splitting the feature set of the training data. Each iteration of the construction algorithm chooses an independent variable in the feature space that best splits the datasets in terms of the dependent variable (output class) [21]. To do so, the method requires a notion of homogeneity of the output class in each subset of the training data that can be used to measure the quality of a split. The Gini impurity is the splitting criterion used in the original *CART* algorithm [42] and is explained in Section 2.2.1 as well. Other splitting criteria such as information gain are available but are computationally slightly more expensive. To stay close to the original classification tree implementation and limit computational complexity we use Gini impurity to determine which features are used to split the dataset and construct the decision trees. Random forests are considered accurate general learning approaches since they are able to receive both low variance and bias by averaging individual trees with high variance and low variance [22]. Furthermore, random forests require little configuration. For these reasons, we use the random forest classifier to predict surgery clusters in this research.

Restricting which splits are used in the construction of the individual decision trees and using only a random subset of the features for each tree, helps random forests to form ensembles with small variance. We set these model characteristics with hyperparameters that need to be configured before the random forest classifier can be trained. In section 7.3 we discuss the method that was used to determine some suitable model hyperparameters. Furthermore, to be able to evaluate the model performance, we split the data in training, validation and test sets. Ultimately, we use the test data to evaluate the model and scheduling performance. The validation and training split ensure we use an unbiased estimate to determine the model hyperparameters. We use the cross-validation method discussed in section 7.1, to evaluate the classification and scheduling performance in this research.

We measure the performance of the classification method with a few basic measures. These do not provide insight into what kind of surgeries are classified correctly and what clusters are correctly assigned to surgeries. We perform a thorough evaluation of the classification model in Chapter 9. Section 9.2 evaluates the performance of the classification model by itself and the joint performance of the clustering and classification model to estimate scheduling characteristics, respectively. In the previous chapter, we developed the initial clusters that are required in order to adhere to scheduling constraints. In order to actually adhere to these constraints, surgeries that belong to a subcluster of one initial cluster may only be classified into a cluster belonging to that same initial cluster. To measure whether or not our classifier is able to classify surgeries to subclusters of the right initial clusters, we design the performance measure *initial cluster accuracy. Accuracy* denotes the fraction of surgeries that are classified correctly and is frequently used to assess the quality of a classification model. The *initial cluster accuracy* is the fraction of surgeries that are



(a) Classification feature importance of orthopedic surgeries.

(b) Classification feature importance of cardiothoracic surgeries.

Figure 7.3: The classification feature importance of the 10 features with the highest feature importance.

classified to the correct initial cluster. Table 7.1 provides these basic performance measures of the random forest classifiers used to predict clusters for orthopedic and cardiothoracic surgeries.

Table 7.1: The random forest classification performance measures.

Hospital case	Accuracy	Initial cluster accuracy
Franciscus	0.96	0.99
MUMC+	0.90	0.98

To examine how the classification models assign clusters to surgeries, we investigate the feature importance. We do so, by studying the reduction of the Gini impurity by all splits of a single feature. The normalized reduction induced by that feature for a single decision tree is defined as the Gini importance. The importance of a feature in an ensemble of trees is the average Gini importance of all trees in the ensemble. Figure 7.3 displays the feature importance of the 10 features with the highest feature importance for the classifiers trained to predict surgery clusters.

The feature importances of the random forest classifiers provided in figure 7.3 indicate some interesting classification behavior. In general, the features determining the initial clusters are important for the classification models. For orthopedic surgery, most of the features with high feature importance determine the surgery subspecialisms and admission types. Also, the planned duration, which is the expected duration provided by the hospital, has high feature importance. This is to be expected as the expected duration can be used to approximate a feature used in clustering: the surgery duration. For cardiothoracic surgeries, the only available features that are also used in determining the initial clusters are the required surgeries. Again, these features have the highest importance. However, in this case, some additional features have considerable importance. Age and BMI are general patient descriptions that apparently can be used to determine clusters. Also, surgery risk scores like The NHR logistic score and euroscore 2 are important.

Table 7.1 lists an accuracy of 0.96 and 0.90 for orthopedic and cardiothoracic classification, respectively. This suggests the classifier is predicting the clusters decently. However, due to class imbalance, these results are misleading. Furthermore, we need to assess the predictive performance of each individual class to determine the actual performance of the classification model. We present a thorough evaluation of this model in section 9.2.

7.3 Hyperparameter tuning

To configure the random forest classification model discussed in the previous section, a suitable set of hyperparameters needs to be found. These hyperparameters configure the classification model and affect its performance. Random forest classification has a few parameters which can be tuned:

- First, the maximal depth (*Max depth*) of each decision tree in the ensemble can be specified to limit the tree depth and prevent overfitting.
- The hyperparameter maximum number of features that are considered when looking for the best splits (*Max features*) can be set to create an ensemble of trees that consider different features in their splits. Doing so reduces the variance of the random forest and consequently reduces the chance a random forest will over-fit.
- The minimum number of samples required to allow the split of a subset in the creation of the decision trees (*Min samples leaf*) is a hyperparameter that manages the size leaves of the individual trees. Again, limiting the number of splits decision trees can help prevent overfitting.
- The final hyperparameter that requires configuration is the number of estimators used to set the number of trees that are incorporated in the forest ensemble (*Number estimators*). Increasing this hyperparameter will result in predictions that are averaged over more trees, reducing the variance even further.

For each of these parameters, a suitable value for the problem of cluster classification needs to be found. To do so, we split the dataset into training and validation sets, for both hospital cases, that are used to identify the performance of a configuration of parameters. We obtain the random forest classification results presented in section 7.2 with the best hyperparameter configuration found with this tuning method. This section discusses the Bayesian hyperparameter optimization which explores the space of parameter configurations and provides the set of parameters that produced the best average model accuracy of all cross-validation splits.

We find the suitable parameter configuration for the random forest classification model used in this research using Bayesian optimization. Bayesian optimization is a technique that allows for an efficient directed search of a global optimization problem. It constructs a surrogate function, a probabilistic model of the objective function, that is iteratively searched over with an acquisition function to determine which next sample to evaluate [24]. We explain this optimization method in more detail in Section 2.4.1. Bayesian optimization can be applied to hyperparameter tuning as the performance of a (classification) model can be considered an objective function to be maximized by exploring the space of hyperparameters. The bayesian hyperparameter optimization used in this research uses a Gaussian process as the surrogate function and chooses one of the acquisition functions: lower confidence bound, negative expected improvement, or probability of improvement, using the GP-hedge strategy [43]. Evaluating the performance of the random forest with a set of hyperparameters requires training and evaluating the performance of the model for every cross-validation split, being an expensive operation. Since Bayesian optimization is suited for complex and intractable objective functions like hyperparameter optimization [24], it is a suitable method for this problem.

To obtain a suitable hyperparameter configuration, for each of the previously discussed parameters we use a range of values to define the possible feature space that is explored with Bayesian optimization. First, we explore the maximum depth of the trees in the [3, 20] interval. For the second hyperparameter, the maximum number of features, we test three configurations. Either we consider all n features, just $\sqrt{n}$ features, or $\log_2 n$ features in the construction of decision trees. We search the range [1, 10] to determine the minimal number of samples in each decision tree leaf. Finally, the interval of the number of decision trees making up the random forest we explore is [100, 500]. Table 7.2 lists the parameter configurations within these ranges that were found to produce the best classification models.

Table 7.2: The best random forest classification hyperparameters found using bayesian optimization.

Hospital case	Max depth	Max features	Min samples leaf	Number estimators
Franciscus	14	sqrt	1	500
MUMC+	13	sqrt	1	100

We use the best parameter configurations identified using Bayesian optimization, described in table 7.2, to obtain the classification model performance described in the previous section. Section 9.2 discusses the accuracy of the classification within each iteration of the optimization method and shows that further optimization iterations are unlikely to find a better hyperparameter configuration. So, the classification models could still be improved, but different random forest hyperparameter configurations are unlikely to produce better results.

7.4 Cluster explanations

Before a case hospital adopts a data-driven scheduling tool it needs to be convinced the scheduling method is robust and fair. When discussing the classification method with the hospital stakeholders, they noted that the scheduling approach is not transparent. The rest of the scheduling model is transparent as the clusters contain distinct interpretable distributions. Furthermore, the heuristic scheduling strategies studied in this research behave in a fixed intuitive manner, transparent to operating room scheduling staff. The random forest classification model, however, performs as a black box. Aside from the general feature importance of the classification model discussed in section 7.2, surgery schedulers have no insight into why a surgery is classified into a particular cluster. To provide insight into the classification of a surgery that is to be scheduled into a cluster, we compute explanations of these individual predictions. This section discusses the *Shapley additive explanations (SHAP)* used to provide local classification explanations.

To explain the predictions made by the classification model used to predict clusters for surgeries to be scheduled, we use *SHAP*. *Shapley additive explanations* aim to explain the prediction of an observation by determining the degree to which each feature contributed to that prediction [25]. *SHAP* decomposes the difference between a class probability computed by a classification model and the probability that class is predicted at random into contributions made by each individual feature. Section 2.5.1 explains the local explanation of machine learning methods with *SHAP* in more detail. Computing *SHAP* feature contributions allows the explanation of individual black-box model predictions. For example, Lundberg et al. (2018) use *SHAP* feature contributions to explain predictions of the risk of hypoxemia during surgery [44]. Similarly, by computing the *SHAP* feature contribution for the surgery classifications into clusters, we explain the surgery and patient attributes that contributed to the assignment of that surgery to a particular cluster.

Figure 9.9 displays a force plot visualizing the *SHAP* feature contributions for an incorrect prediction of a surgery belonging to the cluster *Knee - type c - cluster 0*. Force plots illustrate the feature importance for individual predictions by visualizing the effect of a feature as the length of a horizontal bar in the force plot. Each prediction has a base value that, in the case of classification, corresponds to the probability the predicted class is picked randomly from the observed data. In *SHAP*, each feature increases or decreases that probability, and adding up all feature contributions results in the predicted class probability. The length of a feature bar in a force plot corresponds to the increase or decrease of the class probability as a result of the value of that feature.

The *SHAP* feature contributions illustrated in force plots allow us to explain the opaque classifica-

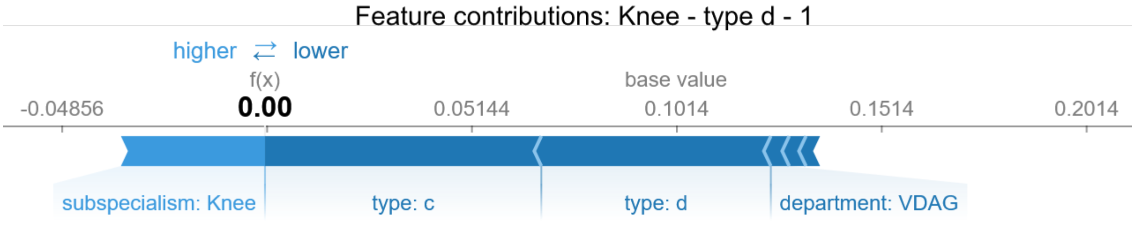


Figure 7.4: SHAP force plot visualizing the feature contributions for an incorrect prediction made for a surgery belonging to the cluster *Knee - type c - cluster 0*.

tion of surgeries into the clusters used in scheduling. Figure 7.4 displays the feature contributions of the incorrect prediction of a surgery belonging to the cluster *Knee - type c - cluster 0* to the cluster *Knee - type d - cluster 1*. This surgery is not predicted to belong to the latter cluster, reflected by the predicted class probability of 0.0. The base value of the probability surgeries belong to the cluster *Knee - type d - cluster 1* is 0.1014. The one-hot encoded feature *subspecialism: Knee* increases the probability of this cluster being correct, which is understandable as the surgery does belong to a cluster of knee surgeries. The other one-hot encoded features offset this increase in predicted class probability, illustrated by their negative feature contribution 'pushing' back the class probability. These features are associated with the surgery type and department to which the patient is admitted, which are in fact different for surgeries belonging to the cluster decrease the predicted class probability. When evaluating the classification of a surgery, a force plot allows us to discover which features contributed to this classification. It should be noted that the *SHAP* feature contributions do not provide a causal explanation. However, they do provide an explanation of the classification of a surgery into a cluster by quantifying the amount each feature contributed to the predicted cluster probability.

Chapter 8

Using cluster properties in scheduling

The scheduling characteristics associated with clusters can be used to schedule patients. The classification discussed in Chapter 7 predicts to which cluster, identified in chapter 6, a surgery that is to be scheduled belongs. Each cluster has specific characteristics that can be leveraged during scheduling. As this section discusses how we can use the scheduling characteristics provided by the predicted surgery clusters to schedule surgeries, it corresponds to research sub-goal *SG 4* discussed in Section 3.5. Figure 8.1 displays the general scheduling approach with the steps corresponding to the scheduling discussed in this chapter highlighted.

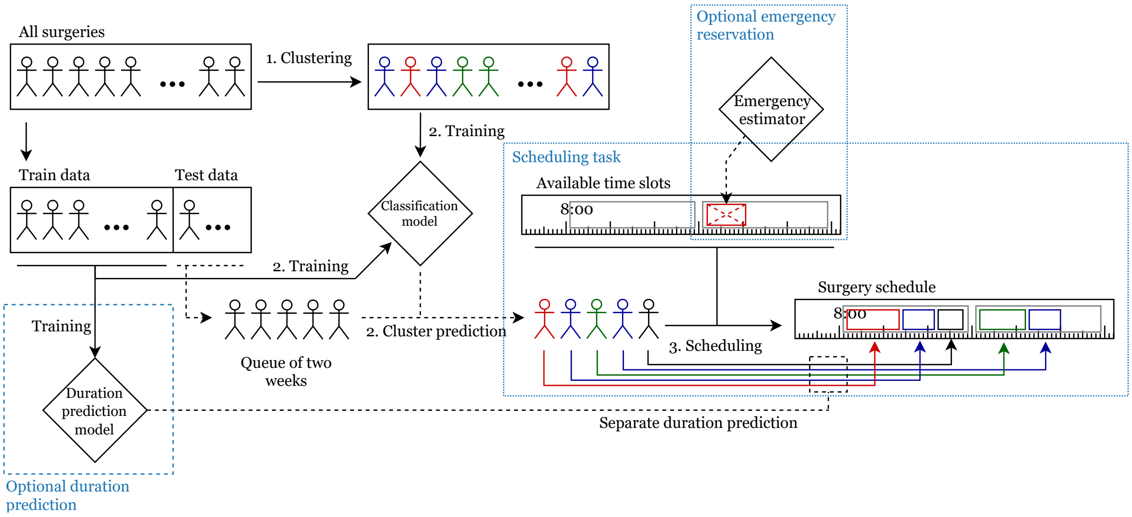


Figure 8.1: The general scheduling method with the highlighted scheduling task.

We propose to employ heuristic scheduling strategies to schedule surgeries based on their expected duration estimated from their predicted clusters. First, Section 8.1 discusses these basic heuristic scheduling strategies identified from literature. Next, we apply these heuristics to the surgery scheduling problem for both case hospitals in Section 8.2. By simulating the schedules resulting from these heuristic strategies, we can compare their performance. Section 8.3 explains this simulation of the schedules obtained with the scheduling strategies. This simulation allows us to compute simulated schedules with the performance measures identified in Chapter 5. These performance measures allow us to determine which scheduling strategy is most suitable for a specific hospital case and allow us to contrast its performance to the current hospital scheduling performance. We perform this first evaluating task, identifying the most suitable for each hospital case in Section 8.4. Section 8.5 discusses the next evaluation task, determining whether or not the proposed scheduling method results in schedules with better performance than the schedules developed by the hospital. Due to the limited scope of this research problem, the cluster based scheduling method proposed in this thesis can not be readily applied in all clinical settings. Section 8.6 discusses the problems introduced by the uncertain nature of emergency surgery demand hindering the adoption of the proposed scheduling method. Next, Section 8.7 adapts the method to more accurately predict the surgery durations used in scheduling.

8.1 Heuristic scheduling strategies

In order to show the surgery clusters identified in Chapter 6 can be used to successfully schedule surgeries, a scheduling method leveraging these cluster properties is required. In this research, we use simple heuristic scheduling strategies to define the best-ordered sequence of surgeries based on the surgery duration or duration variability. Since different hospital cases have different challenges, a set of strategies that is suitable in several clinical situations is required. We review literature to identify the set of suitable strategies that is discussed in this section.

To find a set of scheduling heuristics, suitable for various hospital cases, we review classical operation management literature. In this field, the operating room scheduling problem is a wellstudied topic that is typically considered a variation of the job shop scheduling problem. In the context of operating room scheduling, a surgery could be considered a job, the processing time corresponds to the surgery duration and the machines are the available operating rooms. The effectiveness of heuristics for this problem, in which jobs of different processing time have to be scheduled on a set of machines as efficiently as possible, has been studied extensively [45] [46]. In the study of operating room scheduling as a job shop scheduling problem, the optimization of the schedule typically is a challenge of balancing patient waiting times and idle-time and overtime [47]. Next to these performance measures, this research investigates the effect of scheduling heuristics on all performance measures discussed in Section 5.2.2. The heuristics selected for this research are suitable in different clinical situations [47]:

1. *Shortest processing time first* (*SPF*): Schedule patients in ascending order of expected surgery duration. This allows the most surgeries to take place before the postponement threshold is reached but always postpones the longer surgeries. Furthermore, if delays occur more often in larger surgeries, this strategy avoids the accumulation of delays in the earlier surgeries.

2. *Longest processing time first* (*LPF*): Schedule patients in descending order of expected surgery duration. With this method, larger surgeries are less likely to be postponed. However, since the shorter surgeries are scheduled at the end of the day, more surgeries are likely to exceed the postponement threshold. Moreover, if delays are more frequent in smaller surgeries, LPF scheduling avoids the accumulation of delays in earlier surgeries.

3. *Smallest variance first* (*SVF*): Schedule patients in ascending order of surgery duration uncertainty. Since surgeries with high duration variance are more likely to result in delays, this method avoids the accumulation of delays early in the schedule. Consequently, patient waiting time and overtime are reduced but idle time is increase.

4. *Largest variance first* (*LVF*): Schedule patients in descending order of surgery duration uncertainty. This heuristic schedules the surgeries which are most likely to cause delays early. This is more likely to result in an accumulation of delays, which increases the patient waiting time and expected overtime but reduces idle time.

Various adaptations of these heuristics are proposed, in particular, the *Bailey-Welch* (*BW*) approach is a frequently used job scheduling rule. The Bailey-Welch rule assigns two surgeries to the start of the time slot and performs these surgeries before proceeding with the rest of the schedule [48]. Since just one surgery can be performed at a time, this results in a schedule in which one surgery is already available for surgery while the previous surgery is being performed. This evidently increases the patient waiting time, but also ensures a new surgery can start as soon as the previous surgery is performed, decreasing idle time. Other heuristic methods are useful when considering downstream resources as well [46]. However, as the proposed scheduling method does not schedule the patients outside of their time slot, the downstream capacity is out of scope.

From these heuristics found in literature, we identify eight different scheduling strategies. We use each basic heuristic individually and in combination with the Bailey-Welch rule. This results in the strategies: *SPF*, *LPF*, *SVF*, *LVF*, *SPF-BW*, *LPF-BW*, *SVF-BW*, and *SPF-BW*. Since these strategies are suitable in different clinical situations, it is interesting to investigate which strategy is most suitable for each of the case hospitals.

8.2 Applying heuristic strategies

To show these heuristic strategies improve the surgery schedules currently developed by the hospitals, we need to construct the surgery schedules resulting from these heuristics. The performance of these schedules can then be compared to the schedules as they happened in the hospital. However, in the general operating room scheduling problem, these scheduling strategies can not be trivially applied. The surgery processing time (duration) is uncertain and needs to be estimated for the surgeries that are being scheduled. Similarly, using the variance of surgery duration in scheduling, assumes that the distribution of the duration of that type of surgery is available. This section discusses how the previously identified heuristics are applied to the orthopedic and cardiothoracic surgery scheduling case studies.

The duration and duration variance of a surgery type is provided by the clusters to which surgeries are predicted to belong. As discussed in section 6.3, clusters have distinct scheduling characteristics, including the surgery duration. The surgery processing time that is used by some heuristics is the average duration of the surgeries in the cluster into which the surgery is classified. Likewise, the variance used in the scheduling heuristics is the variance of the expected cluster of the surgery that is being scheduled. Recall that the scheduling method schedules the surgeries within their original time slots. In order to compute a schedule using one of the scheduling heuristics, we first predict the clusters to which the surgeries of a time slot belong. Next, we use a scheduling heuristic to determine the new ordering of surgeries in that time slot. Finally, we map this reordered sequence of surgeries to a schedule by assigning a start time to every surgery in the time slot. Figure 8.2 illustrates this heuristic scheduling.

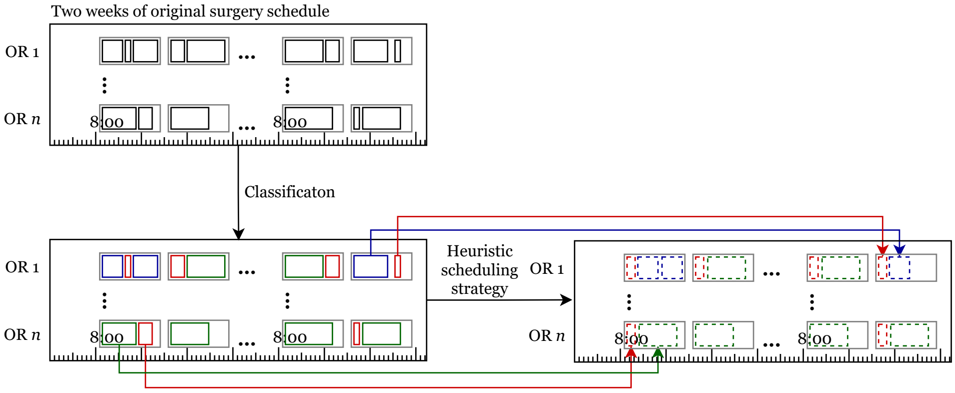

Two weeks of original surgery schedule
OR 1
OR *n*
8:00
8:00
Classificaton
OR 1
OR *n*
8:00
8:00
Heuristic scheduling strategy
OR 1
OR *n*
8:00
8:00

Figure 8.2: The heuristic scheduling approach scheduling the surgeries that occurred in two weeks of a surgery schedule.

To apply one of the heuristic strategies discussed in the previous section, we use an additional scheduling parameter: surgery buffer time. This is the time in between surgeries, reserved by hospitals to prepare for the next surgery. The schedule is obtained by assigning the start time of the time slot to the start time of the first surgery of the time slot. For all following surgeries, we determine the start time by adding the surgery buffer time to the end time of the previous surgery. Consequently, we define the start time τ_i^S for every surgery t_i, which is the o_i'th planned surgery in the reordered time slot $TS = \{(t_1, o_1), \ldots, (t_n, o_n)\}$, as:

$$\tau_i^S = \begin{cases} TS^S & \text{if } o_i = 1 \\ \tau_{i-1}^S + \hat{\delta}_i + \beta & \text{if } 1 < o_i \leq n \end{cases} . \tag{8.1}$$

Here, $\hat{\delta}_i$ is the expected surgery duration, TS^S denotes the time slot start time and we formalize the surgery buffer time as β. We determine the position of a surgery in the reordered time slot $o_i \in 1, \ldots, n$ by using the scheduling heuristic. Furthermore, TS is ordered such that $o_i \leq o_{i+1}$. Note that Bailey-Welch scheduling results in two surgeries with $o_i = 1$.

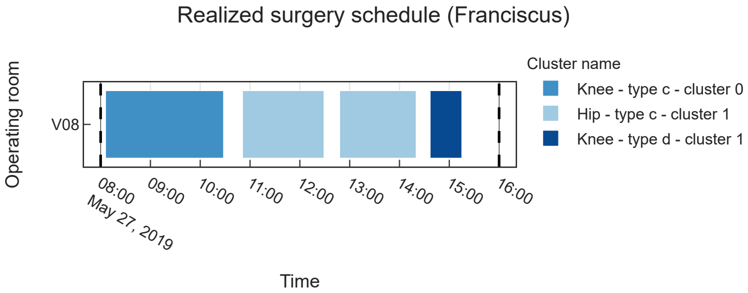
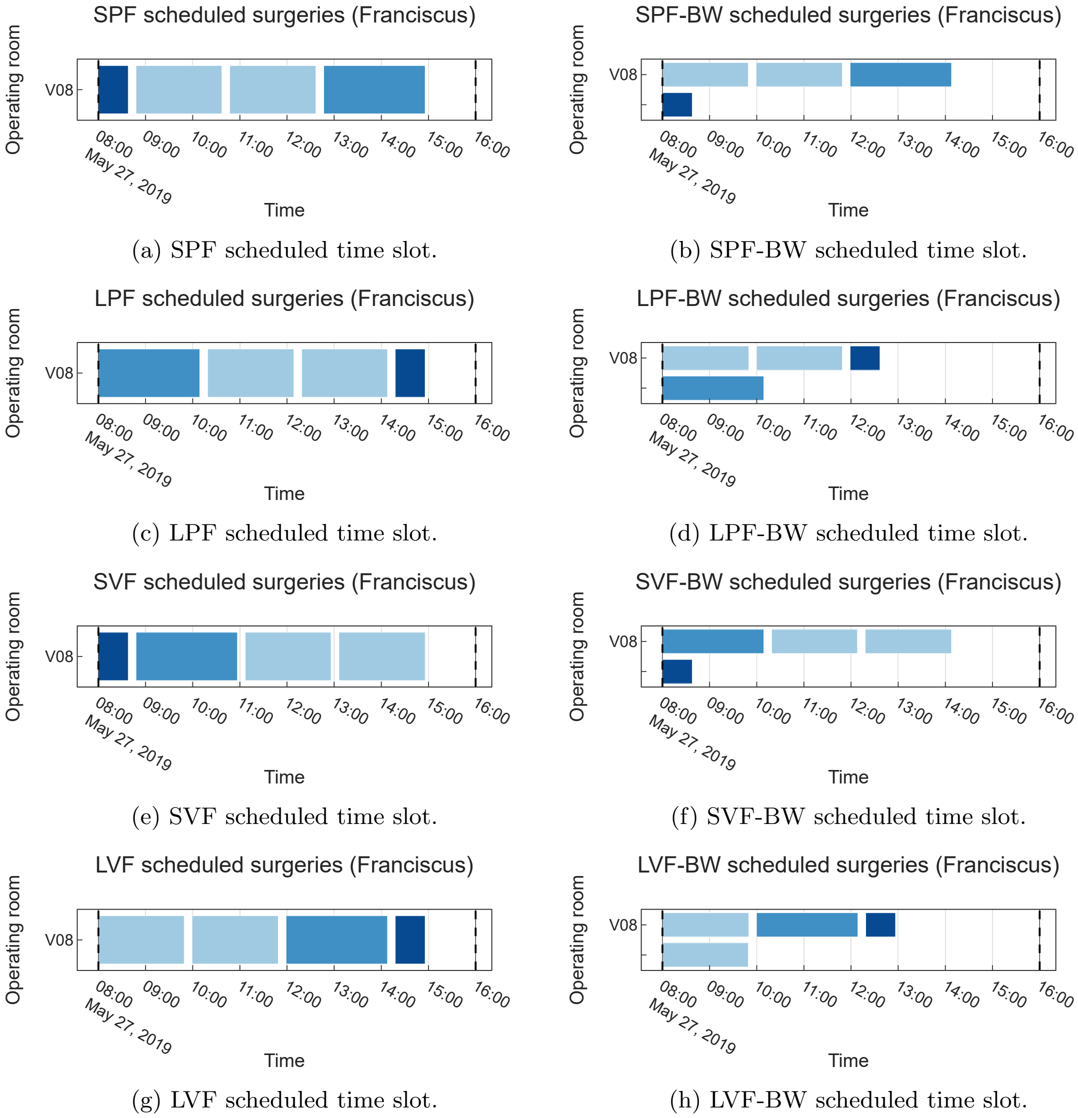


Figure 8.3: The orthopedic surgery schedule of on May 27th in operating room V08. The time slot is depicted by the dashed start- and end-lines

Figures 8.3 and 8.4 show the 4 surgeries of 3 different types performed on May 27th, 2019 in operating room V08 at Franciscus Gasthuis & Vlietland. The original schedule, shown in Figure 8.3, provides a running example that showcases the result of each of the scheduling strategies identified in section 8.1. Note that the first surgery does not exactly start at $08:00$, this shows that for some reason this first surgery was delayed. In contrast to this hospital schedule, the schedules displayed in Figure 8.4 are a result of scheduling with the heuristic scheduling strategies:

- The *SPF* strategies are displayed in Figures 8.4a and 8.4b. Note that the clusters 'Knee - type d - cluster 2', 'Knee - type c - cluster 0' and 'Hip - type c - cluster 0' have increasing expected durations, explaining this surgery order in the figures.
- The *LPF* strategies result in the schedules displayed in Figures 8.4c and 8.4d. Understandably, the ordering of surgeries of these schedules is the reversed order observed in the *SPF* strategies.
- The *SVF* strategies result in the schedules shown in Figures 8.4e and 8.4f. Coincidentally, the ordering of the displayed clusters is the same in terms of duration and duration variability. This results in the same ordering for *SVF* and *LVF* strategies compared to *SPF* and *LPF* strategies.
- The *LVF* strategies are displayed in Figures 8.4g and 8.4h. Again, due to the ordering of the displayed clusters being the same in terms of duration and duration variability, the ordering of these surgeries is the same as the ordering of the *LPF* surgeries.

- The *Bailey-Welch* strategies are shown in Figures 8.4b, 8.4f, 8.4d and 8.4h. Note that the *Bailey-Welch* scheduling results in two surgeries starting at the start of the time slot, displayed as two parallel bars starting at the start of the time slot. Each *Bailey-Welch* strategy starts with the two first surgeries of the corresponding non-BW strategy.



(a) SPF scheduled time slot.

(b) SPF-BW scheduled time slot.

(c) LPF scheduled time slot.

(d) LPF-BW scheduled time slot.

(e) SVF scheduled time slot.

(f) SVF-BW scheduled time slot.

(g) LVF scheduled time slot.

(h) LVF-BW scheduled time slot.

Figure 8.4: The heuristically scheduled time slot of orthopedic surgery on May 27th in operating room V08 in Franciscus Gasthuis & Vlietland. We show each investigated scheduling strategy in one of the subfigures and the time slot is depicted by the dashed start- and end-lines.

As we schedule the original time slot in Figure 8.3 according to the scheduling strategies, the application of the scheduling heuristics discussed in this section is successful. However, this method still has some limitations. The scheduling method assumes the same scheduling buffer time is used between every surgery. In reality, this time is different between every surgery and scheduling with variable buffer sizes might improve scheduling performance. For example, scheduling more buffer after surgeries with large duration variability could help deal with this increased uncertainty. Regardless, the scheduling heuristics are applied effectively, so the performance of these schedules can be compared in the upcoming chapter.

8.3 Simulating schedules for evaluation

The schedules resulting from the heuristic strategies need to be simulated before they can be compared to the schedules developed by the hospital itself. Sections 8.1 and 8.2 identify and apply some relevant scheduling strategies to the operating room scheduling problem. Ideally, we compare the schedules obtained with these heuristics with the current hospital scheduling method. However, no data of the planned schedules is available for cardiothoracic surgery at MUMC+. Conversely, the realized schedules are available in both hospitals. So, to evaluate the performance of our scheduling method, we compare the schedules resulting from the scheduling strategies to the schedules as they happened in the hospital. However, it is not fair to compare the performance of a realized schedule to a planned schedule directly. Factors like emergencies and suddenly unavailable patients impact the surgery schedule. Furthermore, each surgery also has uncertain scheduling characteristics, that become known during its realization. In fact, we cannot even compare two planned schedules trivially. As planned surgery durations are uncertain, these values deviate from their actual realization. Evaluating a poor scheduling method directly can lead to misleading results. For example, a method that consistently underestimates surgery durations would have an easier challenge scheduling these surgeries efficiently. In order to be able to compare planned schedules fairly, we simulate the schedule that would result from a planned schedule. This section discusses the simulation of planned schedules, used to obtain the simulated realizations that can be compared fairly.

The schedule realization simulation substitutes expected surgery times with the actual duration of the corresponding surgeries. Furthermore, it uses a simulation parameter, preoperative patient availability that may vary per hospital case. This is used to allow surgeries to start earlier than their scheduled start time, by a maximum of the number of minutes defined as the preoperative patient availability. Three scenarios depicted in figure 8.5 illustrate the simulation used to create comparable schedules. In scenario (a) the expected duration of the first surgery in the available time slot is too short. The simulation delays the start time for the subsequent surgeries. Scenario (b) and (c) however, show that when the expected duration is too long, the subsequent surgeries occur earlier in the simulated schedule. In that case, the subsequent surgery starts as soon as the previous surgery ends when the patient is already available (scenario (b)) or as soon as the patient becomes available (scenario (c)). The simulated schedules created for the planned schedules approximate how these schedules are expected to perform in reality and can be used to compare different methods of planning.

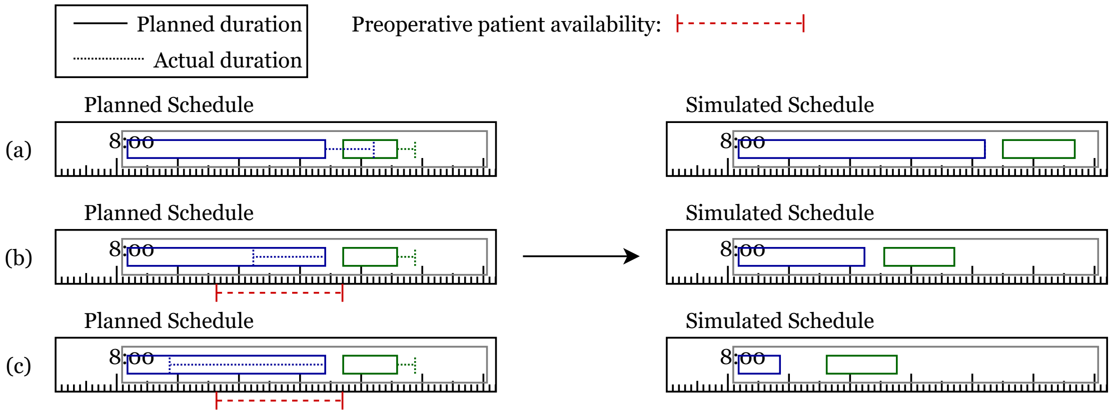


Figure 8.5: Three scenarios that illustrate the simulation method used to compare planned schedules.

We apply this simulation to all planned schedules before evaluating their performance. Figure 8.6 shows the simulated time slot resulting from simulating the realization of the SPF scheduled time slot displayed in Figure 8.4a. The simulation method substituted the actual duration of the surgery and postponed the subsequent surgeries. We observe some interesting simulation results. The surgery belonging to the 'Knee - type c - cluster 0' cluster has a much larger duration in the simulation than in the expected schedules resulting from the heuristic strategies. This is due to the fact that 'Knee - type c - cluster 0' cluster was incorrectly expected to have a shorter duration than the two hip surgeries. Note that this also explains the larger duration for that cluster in the original schedule presented in Figure 8.3.

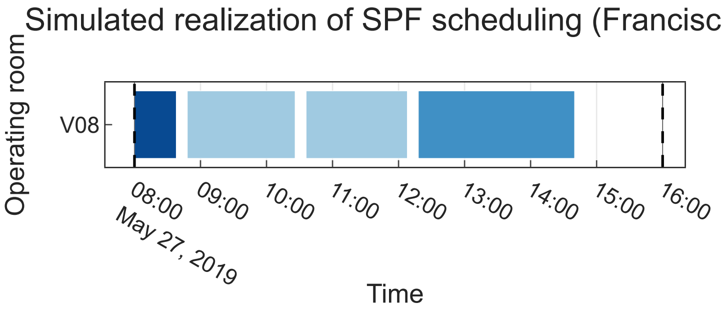


Figure 8.6: The simulated realized orthopedic surgery schedule, resulting from SPF scheduling, of the time slot on May 27th in operating room V08. We depict the available time slot by the two dashed start- and end-lines.

The simulated schedules contain the actual surgery durations as they happened in the hospital and delays are propagated along all subsequent surgeries. This allows the fair comparison of simulated planned schedules and provides an approximation that can be compared to realized schedules. Although this method tackles the uncertainty of surgery durations to which planned schedules are subjected, additional uncertain factors can not be simulated. The realized hospital schedule also incorporated emergencies and other unexpected complications, such as late patient arrivals. In Figure 8.3, the first surgery does not start exactly at 08 : 00 whereas this is the case for the simulation in Figure 8.6. This indicates a late start of the first surgery in the original realized schedule and is an example of a factor of uncertainty that can not be stimulated. Nevertheless, the uncertain surgery durations are incorporated, resulting in a practical approximation of the realized surgery schedule that can be used to compare the schedule to actually realized hospital schedules.

8.4 Comparing scheduling strategies

Comparing the average performance measures of schedules obtained with different scheduling strategies, allows us to determine which strategy is most suitable for a particular case hospital. By computing the performance measures of the schedules resulting from the schedules in the test data, we obtain a different distribution of performance measures for every scheduling strategy. This section explains the first scheduling evaluation task and discusses the method we use to compare the performance measures of scheduling strategies in order to identify the most suitable strategy per hospital case.

To compare the performance of all scheduling strategies, we compare the average performance measures of the scheduling strategies per hospital case. To evaluate specific aspects of the OR schedule, We use the performance measures explained in Section 5.2.2. The overall quality of a surgery schedule is dependant on the case hospital as the importance of a performance aspect varies per case hospital. For example, overtime quantifies the usage of unavailable resources (OR and staff outside of time slots) and patient waiting time evaluates an aspect of the patient experience. For some hospital cases, like cardiothoracic surgery scheduling at MUMC+, the patient waiting time is less important, and efficient resource usage is considered more relevant. To compare scheduling performance for a case hospital we compare the performance measures considered important to that specific hospital.

To address the performance of a scheduling strategy we compute the performance measures of the schedules resulting from scheduling with that strategy. We do so, by scheduling every consecutive two weeks in the test data, simulating the realization of these scheduled plannings, and computing the performance measures defined in section 5.2.2. Note that every week of surgeries in the test data has two adjacent weeks with which a bi-weekly schedule can be constructed: one week before and one week after the current week of the operating room schedule. By evaluating both combinations, we evaluate each week (apart from the initial and final week) twice. Figure 8.7 shows the combination of the test data weeks in order to create the set of schedules that is evaluated.

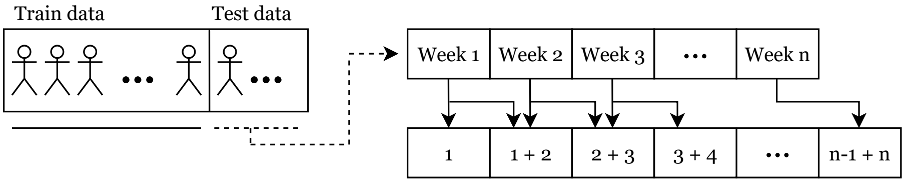


Figure 8.7: The bi-weekly data used to evaluate the heuristic scheduling method.

We compute the average performance measures of all bi-weekly schedules to compare the different scheduling strategies per hospital case. Section 9.3 evaluates these average performance measures in more detail. These performance measures can be compared naively. However, for some performance measures, a higher value is considered good while for others we prefer smaller values. To easily compare the strategy performances and discuss them with hospital stakeholders, we symbolize the performances of the strategies for every measure. Normalizing the performance measures obtained by all strategies allows us to symbolically denote the effect of a scheduling strategy on a performance measure.

By normalizing the performance measures, using the *min-max scaling* results in normalized value $n(x)$ of the value x whose values are ranged in the unit interval:

$$n(x) = \frac{x - x_{min}}{x_{max} - x_{min}}, \tag{8.2}$$

where x_max and x_{min} are the maximum and minimum values observed for x, respectively. For the performance measures idle time, patient waiting time, overtime, postponements and factor postponements, we consider a smaller value to be positive. For these values we inverse the performance score by subtracting it from 1:

$$s(x) = 1 - n(x) \tag{8.3}$$

We subsequently map ranges of these performance measures to symbols in order to obtain the scheduling strategy performance summaries. We list the ranges of $n(x)$ and the corresponding performance symbols in table 8.1.

Table 8.1: The symbolic mapping, used to provide a summary of the performance measures of multiple scheduling strategies.

$n(x)$	idle time, patient waiting time, overtime, postponements, factor postponements	undertime, utilization
[0, 0.2]	++	- -
(0.2, 0.4]	+	-
(0.4, 0.6]	±	±
(0.6, 0.8]	-	+
(0.8, 1]	- -	++

The performance measures of the scheduling strategies when used to schedule orthopedic and cardiothoracic surgeries are compared in section 9.3. That section also provides the symbolized strategy performance summaries and explains we find *shortest processing time first (SPF)* scheduling to be most suitable for orthopedic surgery scheduling. For cardiothoracic surgery, however, we find using *longest processing time first, Bailey-Welch (LPF-BW)* or *largest variance first, Bailey-Welch (LVF-BW)* scheduling results in the most suitable cardiothoracic surgery schedules.

8.5 Evaluating heuristic scheduling strategies

To identify if the proposed scheduling method can be valuable in practice, we evaluate its performance by comparing it to the hospital baseline. The previous section discusses how to compare the performance of the investigated heuristic scheduling strategies. Im this section we discuss the second evaluation task. We explain how to evaluate a single scheduling strategy for a case hospital by comparing it to the hospital schedules. Again, we use the simulation method approximating the realization of a planned schedule. This simulated realization of a planned schedule facilitates the comparison of planned schedules and realized schedules. To understand the performance of a single scheduling strategy for a case hospital, we compute the performance of a simulated realization of a planned schedule and compare this to the performance of a schedule developed by the hospital.

Similarl to the scheduling performance comparison discussed in section 8.4, we use the performance measures explained in Section 5.2.2. Also, we again evaluate the same bi-weekly schedules obtained by combining the each week in the test data with its preceding and subsequent week.

We compare the distribution of performance measures of these bi-weekly schedules. To perform fair comparisons, we ideally compare the simulated realization of the planning obtained from the scheduling strategy to the simulated realization of the planning made by the hospital. Comparing planned schedules comparison is fairer since they take the same uncertainties into account. The actual realization of a schedule is subject to additional unforeseen challenges, impairing its schedule performance measures. However, the planned schedule is unavailable for cardiothoracic surgery scheduling at MUMC+. In this case, we can only compare the performance of the simulated realization of the schedules resulting from the method proposed in this research with the performance of the actual realization of the hospital schedules.

To compare the performance of all evaluated schedules, we investigate the distributions of the performance measures for these schedules. In the previous section, we show how we determine the scheduling heuristics *SPF* and *LPF-BW* to be the most suitable for the orthopedic and cardiothoracic surgery scheduling, respectively. To compare the performance measures of these strategies with their corresponding hospital performance measures, we plot the distribution of these performance measures in boxplots. We discuss this evaluation in detail, in Section 9.3.

8.6 Uncertain emergency demand

Emergency surgeries provide a scheduling challenge that is overlooked by the general method proposed in this research. Recall that emergency surgeries have to happen within 24 hours and ideally as soon as possible. By considering a scheduling horizon of 2 weeks, this research is not focused on managing the emergency demand efficiently. In fact, the scheduling method schedules surgeries that actually happened within a single time slot, assuming these surgeries were scheduled and not an emergency. As the emergencies are typically performed in dedicated operating rooms at Franciscus Gasthuis & Vlietland, the emergency demand rarely impacts the regular surgery schedule. In the case of cardiothoracic surgery, however, the emergencies are scheduled within the regular surgery schedule, delaying or postponing scheduled surgeries. To be able to apply the scheduling method proposed in this thesis to hospital cases where emergencies interact with the regular surgery schedule, time can be reserved for emergencies during scheduling. This allows

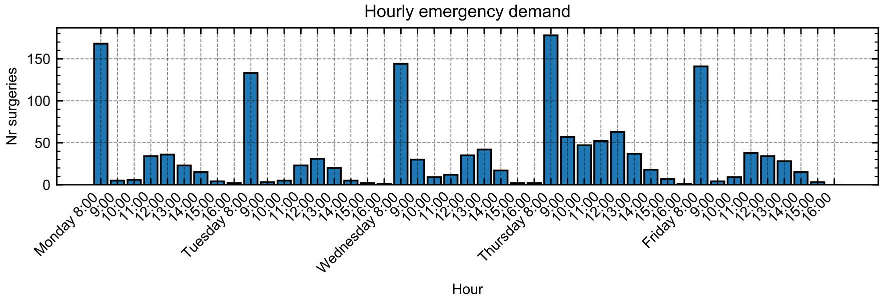


Figure 8.8: The number of emergencies per hour that were performed by the cardiothoracic surgery department at MUMC+ in 2018 and 2019.

emergencies to be performed immediately (in the reserved schedule space) or a scheduled surgery to be delayed within the current scheduling. This section provides an optional method adaptation that can be used to deal with emergencies. It explains how to determine the number of emergencies to consider when reserving time during scheduling. Furthermore, this section studies the moment at which emergencies are typically performed. Figure 8.1 shows the general method overview with this optional adaptation included. Since the orthopedic surgery scheduling case does not include emergencies, we perform this analysis only for cardiothoracic surgery at MUMC+. To evaluate the same scheduling method for both hospital cases we do not employ this optional method in the evaluation tasks discussed in Sections 8.4 and 8.5.

To find out how to reserve space for emergency surgeries, we compute the number of emergency surgeries performed per week and study the times at which these surgeries typically occur. First, we compute the empirical distribution of the number of emergencies per week. Assuming the number of emergencies in a week is an i.i.d. random variable whose distribution is the empirical distribution, allows us to compute some useful descriptive statistics. The mean and median number of emergencies provide an expected number of surgeries for every week. Additionally, the sample proportion of an event provides an estimate of the probability of that event being observed. In this case, the proportion of weeks with at most m number of emergencies provides an estimate of the probability that during a particular week w the number of emergencies E_w is at most m. We compute this proportion as follows:

$$P(E_w \leq m) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}\{E_i \leq n\}, \tag{8.4}$$

where E_i is an observation in the empirical distribution based on the observations $\{E_1, E_2, \ldots, E_n\}$. Next, we study the number of emergencies per working hour in the week, in order to find out when reserving time for emergencies would be most effective. To do so, we sum the number of emergencies per hour of the week.

Table 8.2: Emergency frequency descriptive statistics for cardiothoracic surgery scheduling at MUMC+. The statistics describing the distribution of the weekly of cardiothoracic emergencies.

Average	Median	σ	$P(E_w = 0)$	$P(E_w \leq 1)$	$P(E_w \leq 2)$	$P(E_w \leq 3)$
0.904762	1	0.94588	0.419048	0.752381	0.92381	1

Table 8.2 lists the descriptive statistics used to determine how much time needs to be reserved for emergencies. Note that the probability the number of emergencies in a particular week is smaller or equal to 1 is estimated to be 0.75, while the probability the number of emergencies is at most

2 is estimated to be 0.92. Figure 8.8 displays the number of emergencies per hour of the week. Here, we observe that emergencies start most frequently from 8:00 to 9:00 and hardly from 9:00 to 11:00. Note that the frequent emergency start time of 8:00 indicates batch behavior which is intuitively explained by the start of the surgery time slots at 8:00. As most operating room time slots start at 8:00 an emergency either starts at that time or waits until an OR becomes available. Cardiothoracic surgeries have an average duration of approximately 4 hours, explaining the gap of emergencies from 9:00 to 11:00.

By reserving time in the schedule for emergencies, a hospital is able to use a scheduling method that does not address emergencies specifically. The probability the number of emergencies is at most m, corresponds to the probability of reserving time for m surgeries is sufficient to ensure all regularly scheduled surgeries can still occur. This allows us to determine for some number of reserved surgeries m, the percentage of weeks this m is sufficient:

- We expect reserving time for 1 surgery is sufficient for 75% of the weeks of cardiothoracic surgery at MUMC+.
- Reserving time for 2 surgeries results in an expected 92% of weeks in which an emergency arrives for which no time was reserved in the OR schedule.
- If we accept 5% of the weeks to have an emergency impact the regular schedule for cardiothoracic surgery at MUMC+, we argue reserving time for 3 surgeries is sufficient.

Note that emergencies do not happen at a predetermined time, so the emergency demand will still impact the regular surgery schedule. To treat emergencies as soon as possible, scheduled surgeries need to be delayed. However, by ensuring there is space reserved in the surgery schedule for emergencies, the delayed surgeries can still be performed in the same week. Delaying surgeries is not desirable for some hospitals, particularly for those looking to optimize patient service (e.g. by minimizing patient waiting time). At the cardiothoracic surgery department at MUMC+ however, being able to treat all emergencies and scheduled surgeries with additional patient waiting time is preferred over rescheduling scheduled surgeries to a new week. Furthermore, patients are admitted to the hospital before the start of the surgery time slot, and increased patient waiting time while admitted before surgery is not considered problematic. Hence, reserving time for emergency surgeries provides a method of dealing with emergencies for MUMC+ while allowing the application of a scheduling method for the elective (non-emergency) surgeries.

8.7 Scheduling duration predictions

By improving the estimation of the duration used by the scheduling strategies, we can improve the method proposed in this thesis. Section 8.2 explains that the expected duration of the surgeries in heuristically scheduled time slots is determined by the expected duration of the surgeries in the cluster to which the surgery is predicted to belong. Chapter 6 shows that each identified surgery cluster has a different variability in duration. To estimate the duration of a surgery from a cluster we used the average duration of that surgery cluster. As it assigns the same expected duration to all surgeries in a cluster, this is a crude approximation for clusters with a large duration variance. Individually predicting the surgery durations allows us to improve the estimated duration of surgeries within the heuristically scheduled time slots. This section provides an optional method adaptation that allows us to predict the duration separately. It describes how we adapt the scheduling method to include individually predicted surgery duration and its effect on the scheduling performance measures. Figure 8.1 shows the general method overview with this optional adaptation included. Since this is an optional adaptation, we do not employ this optional method in the evaluation tasks discussed in Sections 8.4 and 8.5. However, we do evaluate its performance by comparing it to the performance of the general scheduling method.

To predict the individual surgery durations, we use random forest regression. Random forest regression is a supervised learning method that learns a model mapping a set of features to a numerical outcome variable. In the context of duration prediction, we use random forest regression to build a random ensemble of decision trees that learn a relation ship between the dependant features used in the classification of the surgery into a cluster and the duration of that surgery. Unsurprisingly, random forest regression is closely related to random forest classification. Section 2.3.1 provides a more in-depth explanation of random forest regression. Apart from the fact that random forest regression allows us to use a similar method as for classification, we use it since it serves as a good general predictor that is easy to configure. We use the same independent variables used for regression that is also used for classification: the surgery characteristics available a priori. However, in contrast to the classification setting, the outcome is not a cluster but the duration of that surgery.

To improve the scheduling method with a more accurate duration estimate, we reserve the predicted duration of each surgery instead of the average duration of a surgery from the predicted cluster. The heuristic scheduling methods studied in this thesis use the surgery duration twice. First, the order of surgeries in the surgery schedule is determined based on either the duration (*SPF*, *LPF*) or duration variance (*SCF*, *LVF*). Next, the scheduled start times are determined by consecutively reserving the expected duration of each surgery. To test the effect of supplementing the scheduling method with individual predictions, we maintain the order of surgeries determined by the surgery clusters but replace the estimated duration with the predicted duration during the start time assignment. The results of this scheduling with this optional method, separately predicting the surgery durations are discussed in Section 9.3.

Chapter 9

Evaluation

The previous Chapters 5 to 8 described how we addressed the research sub-problems discussed in Section 3.5. These chapters described experiments performed to study the scheduling method proposed in this thesis. This chapter evaluates how well the overall method proposed in this thesis allows us to leverage surgery clusters in operational operating room scheduling. As this chapter assesses how well the sub-goals identified in the introduction are fulfilled, it corresponds to the evaluation phase of the *crisp-DM* methodology [16]. We develop and evaluate the surgery scheduling method that plans surgeries based on their predicted cluster characteristics. We do so, by performing the following steps for both the orthopedic and cardiothoracic surgery scheduling cases:

1. First, we develop a clustering model, able to discriminate groups of surgeries based on scheduling characteristics and constraints. Central to the development is the silhouette method, determining the number of clusters per initial cluster. Section 9.1 evaluates the use of this silhouette method and discusses whether or not the clusters found using this method are meaningful.

2. Next, we train a classification model to predict the surgery clusters for surgeries that are being scheduled. In Section 9.2, we evaluate how well this classification model is able to provide a scheduling method with relevant scheduling characteristics. We first discuss the training of this model and evaluate the classification performance of the classification model resulting from this training. Next, we assess the estimation of scheduling characteristics from the predicted clusters resulting from this classification model. Finally, we discuss the results of the proposed classification explanation method.

3. Finally, we use scheduling heuristics to schedule surgeries based on their predicted cluster characteristics. Section 9.3 evaluates this heuristic surgery scheduling. We identify the most suitable strategy per case hospital and compare the performance of that strategy to the performance of the original hospital scheduling. Furthermore, we compare the performance of the optional adapted method, separately predicting the surgery durations, the the performance of the general method.

The aforementioned sections assess the different models and determine their applicability within the overall scheduling method. Finally, Section 9.4 provides a discussion of the results of the experiments described in the preceding sections and discusses whether the developed scheduling method is acceptable.

9.1 Clustering

To group surgeries based on scheduling characteristics and constraints, we develop the clustering method discussed in Chapter 6. To determine a suitable number of subclusters per initial cluster, we use the silhouette method. This method and the sizes of the clusters obtained with it are explained in section 6.5. This section presents the silhouette scores obtained with the silhouette method and discusses the resulting clusters 6.5.

Objective

The objective of this evaluation step is to understand whether the clustering method results in meaningful clusters. As discussed in Section 6.6, clusters are considered meaningful when the scheduling characteristics associated with different clusters are distinct. First, we try to determine if the silhouette method results in a reasonable number of clusters. Next, we aim to assess the difference between the distribution of scheduling characteristics of the resulting clusters.

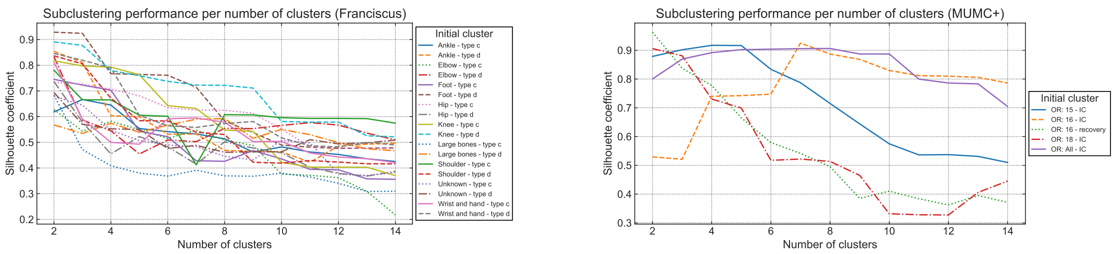
Setup

First, we cluster the entire dataset of orthopedic and cardiothoracic surgeries, discussed in Section 3.4, in the initial clusters described in Section 6.2. Within these initial clusters, we measure the silhouette coefficient after performing agglomerative clustering to identify an increasing number of sub-clusters. The agglomerative clustering is used to cluster the surgeries within an initial cluster-based on the features *surgery duration*, *postoperative length of stay*, and *department after surgery*. Finally, we investigate the distribution of these scheduling characteristics of the resulting clusters.

Execution

As discussed in Chapter 6, we cluster the initial clusters into subclusters using agglomerative clustering with the average Gower distance as the linkage criterion. For the clustering of orthopedic surgeries, we measure the silhouette score for 2 to 9 subclusters per initial cluster. For the case of cardiothoracic surgery scheduling, however, we measure the silhouette score for 2 to 14 subclusters per initial cluster. To obtain the cluster distributions presented in the next section we use the empirical distribution of the numerical scheduling characteristics to fit a kernel density estimation. For the categorical *department after surgery* characteristic, we count the occurrences of each value per subcluster.

Results



(a) Silhouette scoress when clustering orthopedic surgeries.

(b) Silhouette scoress when clustering cardiothoracic surgeries.

Figure 9.1: The silhouette score when subclustering each initial cluster into a variable number of subclusters.

Figure 9.1a shows the silhouette coefficient of subclustering the initial orthopedic surgery clusters into 2 to 9 sub-clusters. Figure 9.1b displays this silhouette coefficient for the subclustering of cardiothoracic surgeries into 2 to 14 subclusters. In general, the silhouette coefficient does not seem to be increasing to a new maximum after 4 and 10 clusters, for orthopedic and cardiothoracic sub-clustering, respectively. The number of clusters resulting in the highest silhouette scores is listed in Table 6.4 for every initial cluster. The sizes of these sub-clusters per initial cluster are presented in Figure 6.6 in chapter 6.

To evaluate the scheduling characteristics for every subcluster, we plot the distributions of the *surgery duration* and *postoperative length of stay.* Likewise, we plot the number of surgeries admitted to each *department after surgery* for every subcluster. Figure 9.2 displays the *surgery duration* and *postoperative length of stay* for the initial knee surgery clusters. Additionally, it provides the *department after surgery* for all orthopedic surgery clusters. Similarly, Figure 9.3 shows the scheduling characteristic distributions for cardiothoracic surgeries. Figures 9.2a, 9.2b, 9.3a, and 9.3bshow the distributions of the continuous scheduling characteristics duration and postoperative length of stay for the initial 'Knee' and 'OR: All' clusters. Figures 9.2 and 9.3 display the distribution of the postoperative department per initial cluster for the orthopedic and cardiothoracic surgery scheduling cases respectively. We provide distributions of the other sub-clusters per initial cluster in Appendix D.

Visually inspecting Figures 9.2, 9.3, D.1, and D.2 show that the distributions of subclusters identified in each initial clusters are different. For example, in Figure 9.2 we clearly observe a longer duration with more variability for the knee surgery clusters *Type d - cluster 0* compared to *Type d - cluster 1.* The knee surgery clusters *Type c - cluster 0* and *Type c - cluster 1* however, have a similar duration distribution location and variability but the former has a shorter postoperative length of stay. Similarly, the clusters observed in Figure 9.3 either have a distinct duration or length of stay distribution. In general, each resulting initial cluster has subclusters with different distributions of scheduling characteristics, for some notable cases this is however not immediately clear:

1. Similarly to the knee surgery clusters, some clusters are distributed similarly in terms of duration but have distinct *LOS* distributions. This is the case for the following subclusters:
   - The ankle surgery clusters displayed in Figures D.1a and D.1b: *Type d - cluster 0*, *Type c - cluster 1* and *Type c - cluster 2.* Cluster *Type d - cluster 0* has the shortest *LOS.* Furthermore, the *LOS* of *Type c - cluster 2* is shorter than *Type c - cluster 1* as well.
   - The foot surgery clusters displayed in Figures D.1e and D.1f: *Type c - cluster 1*, *Type d - cluster 0*, and *Type d cluster 1.* Cluster *Type d cluster 1* has the shortest *LOS* and the *LOS* of *Type c - cluster 1* is shorter than that of *Type d - cluster 0.*
   - The hip surgery clusters displayed in Figures D.1g and D.1h: *Type c - cluster 0* and *Type c - cluster 1.* The *LOS* of *Type c - cluster 1* is shorter than that of *Type c - cluster 0.*
   - The surgeries exclusively possible in operating room 15 displayed in Figures D.2c and D.2d: *IC - cluster 0* and *IC - cluster 1.* Note that the average length of stay is similar, but the variability of these *LOS* distributions is different. The *LOS* distribution of *IC - cluster 1* is very narrow and thus contains little variability, whereas the *IC - cluster 0 LOS* is broad (contains a lot of variability).
2. Since we display the subcluster distributions of initial clusters with the same subspecialism and possible OR in the same figure, we can observe similar distributions that do not belong to the same initial cluster. For example:
   - The shoulder surgery clusters displayed containing clinical (type c) and daily (type d) admissions, displayed in Figures D.1k and D.1l, are distributed similarly. This

means they should be scheduled similarly, but have different scheduling constraints. In this case, these patients require capacity in different postoperative departments: the orthopedic department for surgeries of 'type - c' and the daily admission department for surgeries of 'type d'.

3. The clustering is not based solely on duration and *LOS*. Some clusters have similar duration and LOS but have different postoperative departments. For example:

   - The surgeries exclusively possible in operating room 16, displayed in Figures D.2e and D.2f: *IC - cluster 0* and *IC - cluster 2*. These clusters are distributed similarly in terms of duration and length of stay. Nevertheless, Figure 9.3c shows surgeries from these clusters move to different departments after surgery. Again, these surgeries can thus be scheduled similarly when solely optimizing the efficient usage of the operating room. However, when managing the postoperative flow of patients as well, these surgeries would need to be scheduled differently.

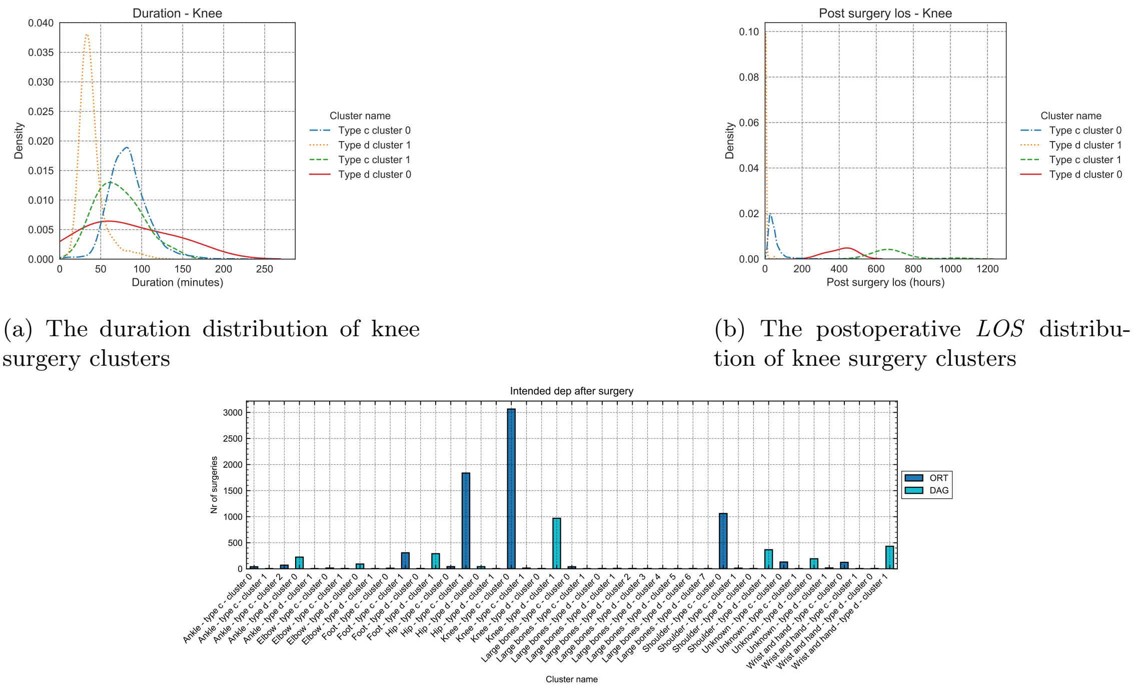
A few other interesting distributions stand out. In particular, some clusters distributions contain a lot of variability while others distributions are very narrow (contain little variance) are contain notably little variability. Examples of these include:

1. Clusters with very little variability: *Elbow - type d - cluster 1* displayed in Figures D.1c and D.1d, *Hip - type d - cluster 1* displayed in Figures D.1g and D.1h and *OR: All - IC - cluster 7* displayed in Figures D.2a and D.2b.

2. Clusters with large variability: *Wrist and Hand - type d - cluster 1* displayed in Figure D.1p and *Unknown - type c - cluster 1* displayed in Figure D.1n

Section 6.5 discusses the clusters identified using the agglomerative clustering and the silhouette method and presents a summary of the resulting clusters. This summary, describing the cluster sizes, is provided in Figure 6.6. Note that the clusters with notably large or small variability, discussed above, correspond to clusters containing a small number of surgeries.

We provide a more thorough discussion of these results in Section 9.4. The small clusters indicate overfitting and are not considered to be meaningful. Therefore, we only consider surgeries with at least 5 surgeries in the rest of the method. Regardless of the cluster sizes, the distribution of the found clusters is indeed distinct. We thus consider these clusters meaningful and use them in the rest of this method.

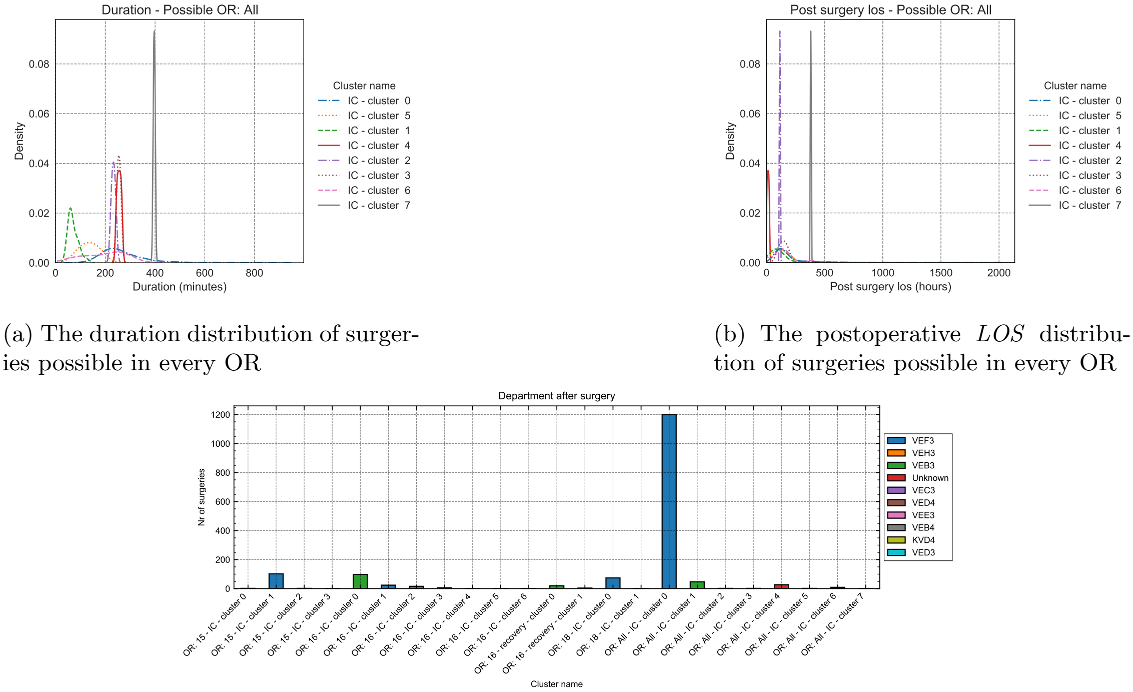


(a) The duration distribution of knee surgery clusters

(b) The postoperative *LOS* distribution of knee surgery clusters

(c) The postoperative department distribution for orthopedic surgery clusters

Figure 9.2: Distribution of orthopedic surgery cluster scheduling characteristics



(a) The duration distribution of surgeries possible in every OR

(b) The postoperative *LOS* distribution of surgeries possible in every OR

(c) The postoperative department distribution for cardiothoracic surgery clusters

Figure 9.3: Distribution of cardiothoracic surgery cluster scheduling characteristics

9.2 Classification

In order to provide a scheduling method with scheduling characteristics for surgeries that are being scheduled, we developed a classification method. Chapter 7 explains this classification method, predicting the clusters used to distinguish surgeries with distinct scheduling characteristics. The proposed classification model is trained using a rolling cross-validation and Bayesian hyperparameter tuning method. This section evaluates that training and discusses the resulting classification model. We assess the performance of this classification model by itself and as a tool to estimate scheduling characteristics based on the clustering discussed in the previous section. Finally, we evaluate the method proposed to explain the classification of surgeries into clusters.

Objective

The objective of this classification evaluation is to determine whether the proposed classification method is able to accurately predict surgery clusters and can be used during scheduling. To evaluate the training of our classification model we aim to perform three steps:

- We aim to verify if the classification accuracy of the model when predicting validation data is similar to the accuracy when predicting test data.
- We study the concept drift of the surgery process. To do so, we measure the accuracy of the model during each cross-validation step using the configuration found using the Bayesian hyperparameter tuning.
- Finally, we assess whether or not more tuning iterations could result in a more accurate classifier. To study this, we measure the accuracy of the classifier after each Bayesian optimization iteration.

To determine if the random forest classification model is successful in predicting the surgery clusters, we evaluate the performance of this classifier. To do so, we determine its *accuracy* and *initial cluster accuracy* on unseen test data. However, as the accuracy does not provide insight in which clusters or initial clusters are predicted correctly, these measure is not sufficient. Hence, to measure the confusion of specific clusters, we compute two classifier confusion matrices: one evaluating the prediction of the clusters used in scheduling and one of the initial cluster prediction.

To leverage the scheduling characteristics approximated from predicted clusters during scheduling, this approximation needs to be accurate. Hence, we aim to evaluate the estimation performance of the combined clustering and classification method. More specifically, we determine how accurate the estimation of categorical characteristics is and how much the estimated continuous characteristics deviate from their actual characteristics. We measure the accuracy of the combined method to predict the department after surgery and compute the *mean absolute error (MAE)* to quantify the difference between the estimated and actual surgery duration and length of stay.

Finally, to show we can explain the classification of a surgery into a cluster we aim to compute the feature contributions for every feature used in that classification. To determine these feature contributions we compute *SHAP* feature contribution values for surgeries that are predicted to a particular cluster.

Setup

To train the random forest classifier proposed in Chapter 7 we split the surgery case hospital datasets in a test and training set. Section 7.1 explains this cross-validation and Bayesian hyperparameter tuning approach in more detail. First, we cluster the surgeries in the aforementioned datasets into the clusters discussed in section 6.6. Next, these cluster labels are used as dependent variables to train the random forest classification model. Section 3.4 discusses the available data that is used to predict the cluster for every surgery. Appendix B provides a summary of the data per hospital. Note that the features summarized in this appendix include the features used in clustering. These clustering features are not used during classification.

The training test set is split l times in m training and validation datasets, where l is the number of iterations in the Bayesian hyperparameter tuning method discussed in Section 7.3. Each iteration, we test a hyperparameter configuration by training a random forest model on the training data and computing its prediction accuracy on the validation data. The configuration resulting in the best average prediction accuracy in every training and validation split is used to train the model on the complete training dataset. To compute the confusion matrices, used to evaluate the resulting classifier, we count for all surgeries in the test set which cluster is predicted and to what cluster it actually belongs.

The surgeries in the test data are not used to train the classification model but do have realized scheduling characteristics. Every surgery has a realized *department after surgery*, *duration*, and *length of stay (LOS)*. To investigate the estimation performance of the surgery characteristics from the predicted surgery clusters, we compute the estimation performance for each of these characteristics.

Finally, to test the explanation of surgery classifications using *SHAP* feature contributions, we compute the *SHAP* values for the predictions made by our trained classifier. To evaluate the explanation of these predictions using *SHAP* feature contributions we compute these feature contributions for a predicted clusters.

Execution

As discussed in Section 7.1, we perform $l = 100$ Bayesian optimization iterations and split the complete training dataset $m = 9$ times in a training and validation set using the rolling time series cross-validation. Section 7.3 explains the specific configurations searched over with Bayesian optimization. The specific explored parameters are:

1. *Tree depth*: [3, 20].
2. *Maximum number of features*: $\{n, \sqrt{n}, \log_2 n\}$, with n the total number of available features.
3. *Minimum number of samples per leaf*: [1, 10].
4. *Number of decision trees*: [100, 500].

We use the classifier resulting in the highest validation accuracy as the model to be used in the scheduling method. Table 7.2 lists this configuration in section 7.3.

To determine the effectiveness of the estimation of scheduling characteristics from the predicted clusters, we first obtain the estimated scheduling characteristics. Next, we determine the estimation error of these surgery characteristics in the following manner:

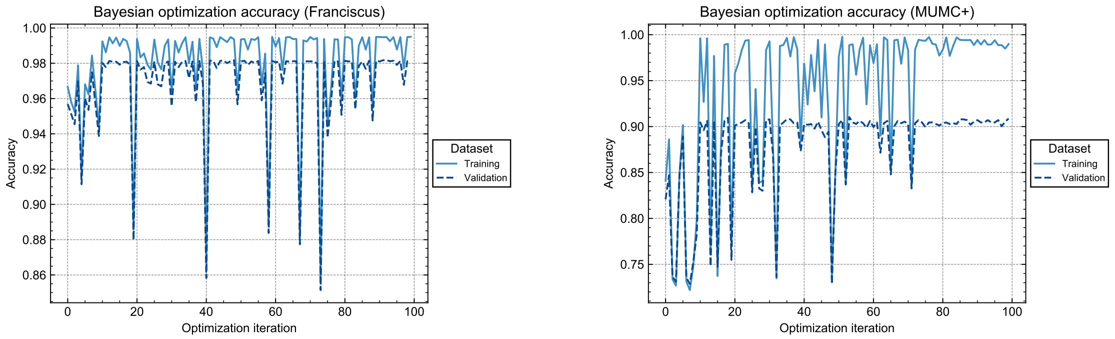
1. For the continuous variables, *duration*, and *length of stay* we compute the average of the observations in the cluster to which the surgery is predicted to belong. To obtain the estimation performance we compute the *MAE*.
2. For the categorical feature, *department after surgery* we use the mode of the predicted cluster to estimate that variable for a surgery that is being scheduled. The estimation performance is obtained by calculating the estimation *accuracy*, the factor of correctly predicted post-operative departments.

Since the postoperative department accuracy does not allow us to evaluate which department after surgery is confused with which department, we additionally plot the confusion matrix.

Finally, to show the explanation of a predicted cluster using *SHAP* feature contributions, we classify a knee surgery performed on September 14'th 2019. We used the classification model used in the experiments discussed in the previous sections to perform this classification. Next, we compute the *SHAP* feature contributions and show them in a force plot. The interpretation of this force plot is explained in Section 7.4.

Results

Figure 9.4 shows the progression of the random forest classification accuracy during each iteration of the Bayesian optimization. The accuracy presented in this figure is the average accuracy of all training and validation datasets. For both hospital cases, the prediction accuracy seems to converge to a local optimum after approximately 10 iterations. For orthopedic surgeries, the accuracy on the training and validation data converge to approximately 0.99 and 0.98, respectively. For cardiothoracic surgery the training accuracy optimum is approximately 0.99, whereas this optimum for the validation dataset is approximately 0.9. We use these best configurations, resulting in the highest validation accuracy, to train the random forest classifier that is used during scheduling. Table 7.2 in Section 7.3 lists these configurations. We train the model resulting from these hyperparameter settings on the complete training data and evaluate its performance in the next section.



(a) Bayesian hyperparameter optimization orthopedic surgery classifier

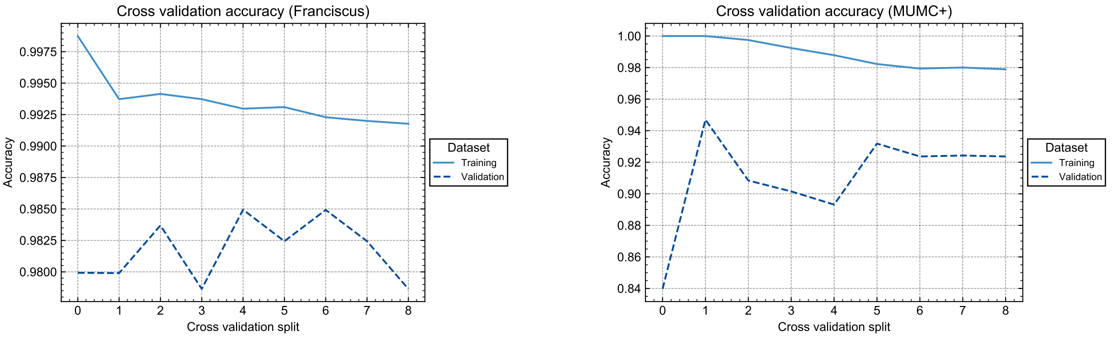
(b) Bayesian hyperparameter optimization cardiothoracic surgery classifier

Figure 9.4: The accuracy during the Bayesian hyperparameter optimization of the random forest classifier used to predict surgery clusters before scheduling.

Figure 9.5 displays the cross-validation accuracy of the random forest classification model. It depicts the validation and training accuracy for each cross-validation split testing the best configuration found with Bayesian optimization. For both hospital cases we observe the training accuracy slowly decreases with each subsequent cross-validation split. This training accuracy decreases from approximately 1 to 0.99 and 0.98 for orthopedic and cardiothoracic surgery, respectively. For orthopedic surgeries, the validation accuracy seems consistent and approximately 0.98. For cardiothoracic surgeries, however, the accuracy is smaller in the first split but remains consistent and approximately 0.92 for the next splits.

Section 7.2 discusses the performance of this classifier and lists a summary of the performance in Table 7.1.

The higher accuracy for orthopedic surgeries, when compared to the accuracy of cardiothoracic surgery classification stands out. Orthopedic and cardiothoracic surgeries are classified with an accuracy of 0.96 and 0.9, respectively. This shows the classifiers are able to correctly predict the



(a) Training accuracy orthopedic surgery classifier

(b) Training accuracy cardiothoracic surgery classifier

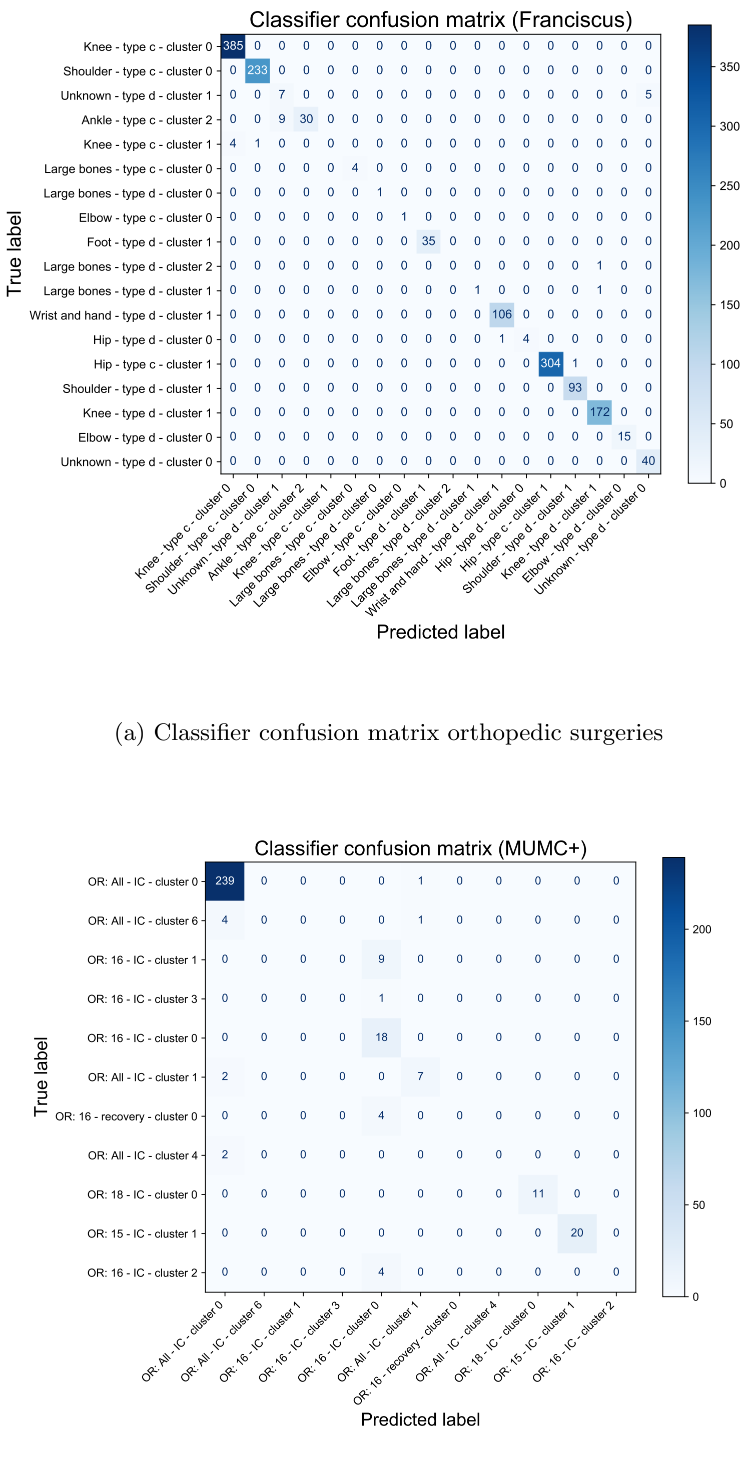
Figure 9.5: The accuracy each cross-validation during training of the random forest classifier used to predict surgery clusters before scheduling.

surgery cluster for at least 90% of the surgeries in the test data. Unfortunately, within the 4% and 10% of surgeries that are misclassified there are some surgeries that are predicted to belong to the incorrect initial cluster. The initial cluster accuracy of the classifier predicting orthopedic surgeries is 0.99 and the classifier predicting cardiothoracic surgeries has an initial cluster accuracy of 0.98.

Figure 9.6 shows the confusion matrix denoting the number of surgeries that belong to and are predicted to be in every combination of surgery clusters. This allows us to identify which clusters are confused by the classification model. For both hospital cases, we observe most of the hospital cases are predicted to belong to the correct cluster. Some clusters are however confused. For orthopedic surgeries, for example, *Knee - type c - cluster 1* surgeries are predicted to belong to *Knee - type c - cluster 0*, and *Ankle - type c - cluster 2* is confused with *Unknown - type d - cluster 1*. Similarly, cardiothoracic surgeries belonging to *OR: 16 - IC - cluster 1* are predicted to belong to *OR: 16 - IC - cluster 0*, and *OR: 16 - recovery - cluster 0* surgeries are predicted to belong to *OR: 16 - IC - cluster 0*.

Confusing clusters with the same initial is not necessarily a problem but confusing surgeries belonging to different initial clusters can result in infeasible schedules. Figure 9.7 displays the initial cluster confusion for both hospital cases. This shows which initial clusters are confused by the classification model. Understandably, the confused clusters that belong to different initial clusters in Figure 9.6 are shown again in Figure 9.7. Among other confusions, we observe the clusters *Ankle - type c - cluster 2* and *Unknown - type d - cluster 1* confused 9 times in the general confusion matrix, resulting in 9 confusions of the initial clusters *Ankle - type c* and *Unknown - type d*. Additionally, we find that the initial cluster for 4 cardiothoracic surgeries are confused. These are predicted to belong to the initial cluster *OR: 16 - IC* while they actually belong to the initial cluster *OR: 16 - recovery*.

Table 9.1 lists the general estimation effectiveness measures of the combined clustering en classification method. It provides the *MAE*for the estimated duration and LOS and the postoperative department prediction accuracy for both hospital cases. The estimation method results in smaller *duration* and *length of stay MAE* and higher *department after surgery* accuracy for orthopedic surgeries than for cardiothoracic surgeries. Figure 9.8 shows the confusion matrices of the prediction of the department after surgery. We observe in Figure 9.8a that for orthopedic surgery scheduling, 29 admissions to the orthopedic department are incorrectly predicted to be admissions to the daily admission department. Furthermore, Figure 9.4b shows that nearly all surgeries are predicted to go to the intensive care unit *VEF3*. Moreover, 29 surgeries are correctly predicted to go to the cardiology ward *VEB3* immediately after surgery, the rest of the departments are



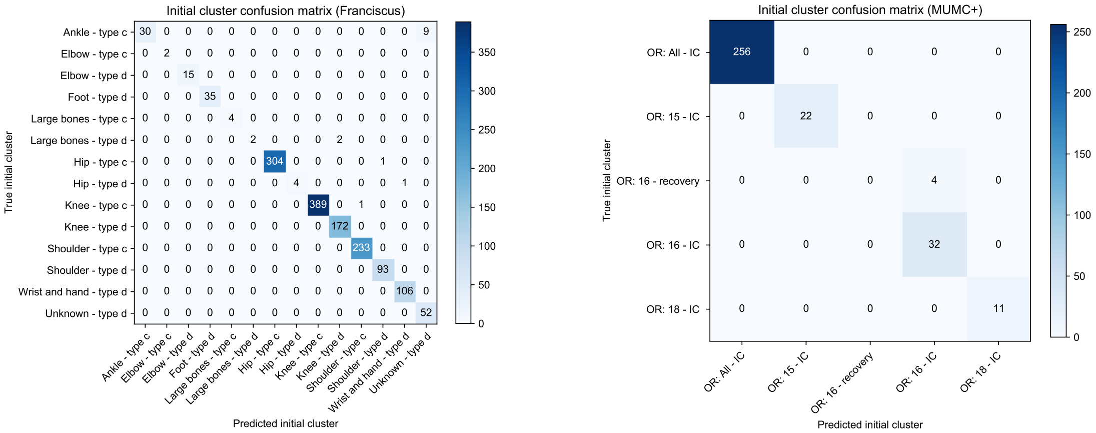
True label \ Predicted label	Knee - type c - cluster 0	Shoulder - type c - cluster 0	Unknown - type d - cluster 1	Ankle - type c - cluster 2	Knee - type c - cluster 1	Large bones - type c - cluster 0	Large bones - type d - cluster 0	Elbow - type c - cluster 0	Foot - type d - cluster 1	Large bones - type d - cluster 2	Large bones - type d - cluster 1	Wrist and hand - type d - cluster 1	Hip - type d - cluster 0	Hip - type c - cluster 1	Shoulder - type d - cluster 1	Knee - type d - cluster 1	Elbow - type d - cluster 0	Unknown - type d - cluster 0
Knee - type c - cluster 0	385	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Shoulder - type c - cluster 0	0	233	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Unknown - type d - cluster 1	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5
Ankle - type c - cluster 2	0	0	9	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Knee - type c - cluster 1	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Large bones - type c - cluster 0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0
Large bones - type d - cluster 0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Elbow - type c - cluster 0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Foot - type d - cluster 1	0	0	0	0	0	0	0	0	35	0	0	0	0	0	0	0	0	0
Large bones - type d - cluster 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Large bones - type d - cluster 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0
Wrist and hand - type d - cluster 1	0	0	0	0	0	0	0	0	0	0	0	106	0	0	0	0	0	0
Hip - type d - cluster 0	0	0	0	0	0	0	0	0	0	0	0	1	4	0	0	0	0	0
Hip - type c - cluster 1	0	0	0	0	0	0	0	0	0	0	0	0	0	304	1	0	0	0
Shoulder - type d - cluster 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	93	0	0	0
Knee - type d - cluster 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	172	0	0
Elbow - type d - cluster 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	0
Unknown - type d - cluster 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	40

(a) Classifier confusion matrix orthopedic surgeries

True label \ Predicted label	OR: All - IC - cluster 0	OR: All - IC - cluster 6	OR: 16 - IC - cluster 1	OR: 16 - IC - cluster 3	OR: 16 - IC - cluster 0	OR: All - IC - cluster 1	OR: 16 - recovery - cluster 0	OR: All - IC - cluster 4	OR: 18 - IC - cluster 0	OR: 15 - IC - cluster 1	OR: 16 - IC - cluster 2
OR: All - IC - cluster 0	239	0	0	0	0	1	0	0	0	0	0
OR: All - IC - cluster 6	4	0	0	0	0	1	0	0	0	0	0
OR: 16 - IC - cluster 1	0	0	0	0	9	0	0	0	0	0	0
OR: 16 - IC - cluster 3	0	0	0	0	1	0	0	0	0	0	0
OR: 16 - IC - cluster 0	0	0	0	0	18	0	0	0	0	0	0
OR: All - IC - cluster 1	2	0	0	0	0	7	0	0	0	0	0
OR: 16 - recovery - cluster 0	0	0	0	0	4	0	0	0	0	0	0
OR: All - IC - cluster 4	2	0	0	0	0	0	0	0	0	0	0
OR: 18 - IC - cluster 0	0	0	0	0	0	0	0	0	11	0	0
OR: 15 - IC - cluster 1	0	0	0	0	0	0	0	0	0	20	0
OR: 16 - IC - cluster 2	0	0	0	0	4	0	0	0	0	0	0

(b) Classifier confusion matrix cardiothoracic surgeries

Figure 9.6: The classifier confusion matrix denoting for every actual and predicted cluster how many surgeries are classified as such.



(a) Initial cluster confusion matrix orthopedic surgery

(b) Initial cluster confusion matrix cardiothoracic surgery

Figure 9.7: The initial cluster confusion matrix denoting for every actual and predicted initial cluster how many surgeries are classified as such.

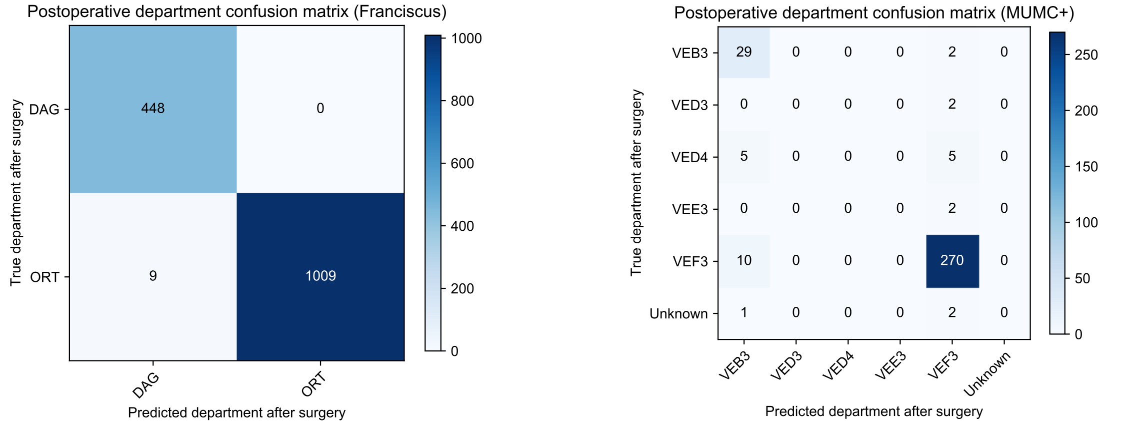
incorrectly predicted to go to *VEF3*. Furthermore, it stands out that all cardiothoracic surgeries are expected to go to either department *VEB3* or *VEF3*. The small number of surgeries which moves to a different department is incorrectly predicted.

Table 9.1: The joint clustering and classification performance of method estimating surgery characteristics before scheduling. The mean absolute error (MAE) of surgery duration is denoted in minutes whereas the length of stay is measured in hours.

Hospital case	Duration (MAE)	Post surgery length of stay (MAE)	Department after surgery (accuracy)
Franciscus	20.91	28.40	0.98
MUMC+	59.08	101.38	0.91

Figure 9.9 displays two force plots visualizing the *SHAP* feature contributions of a surgery belonging to the cluster *Knee - type c - cluster 0* for two predictions. The incorrect cluster *Knee - type d - cluster 1* probability 0.00 is explained by a large positive contribution of the feature *subspecialism: Knee* which is counteracted by large negative contributions of the features *type: c*, *type: d*, and *department: VDAG*. These features "push" the cluster probability from the base probability 0.1014 to the predicted class probability of 0.00. Similarly, the cluster probability 0.95 of the correct cluster *Knee - type c - cluster 0* is explained by large contributions of the features: *subspecialism: Knee*, *type: c*, *type: d*, and *operator: SOD005*. All visible features have a positive contribution, increasing the base value of the predicted class probability from 0.3499 to 0.95.

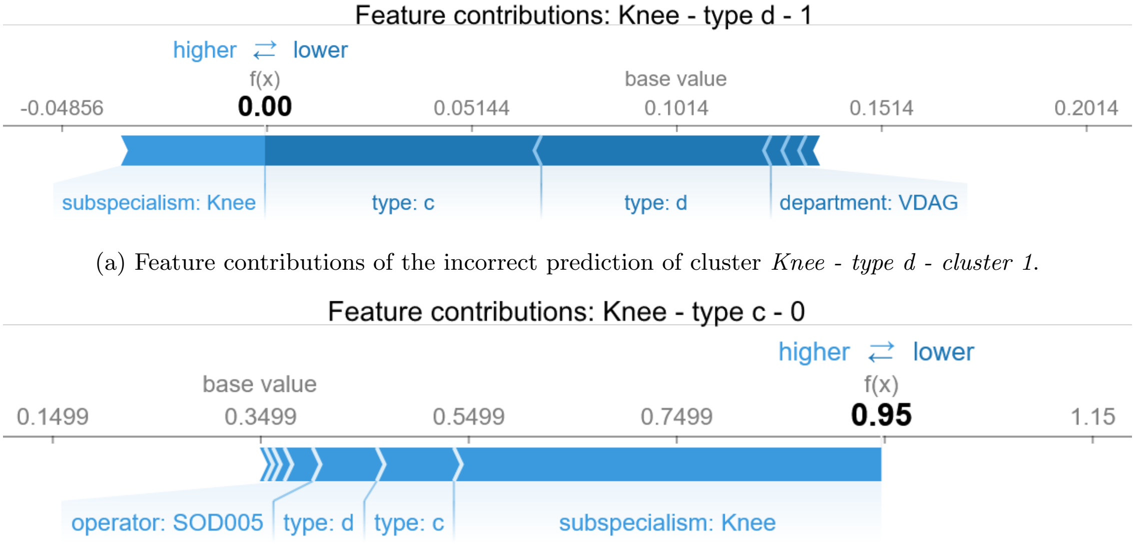
As discussed further in section 9.4, this classification method does perform perfectly. The model consistently incorrectly classifies several (small) clusters and even confuses some initial clusters. However, these problems currently do not result in infeasible schedules for the hospital cases studied in this thesis. Hence, we continue to use this classification model in the scheduling method.



(a) Estimated department after orthopedic surgery confusion matrix

(b) Estimated department after cardiothoracic surgery confusion matrix

Figure 9.8: The estimated department after surgery confusion matrix denoting for every actual and estimated department the number of patients that are estimated to go to that department after surgery.



(a) Feature contributions of the incorrect prediction of cluster *Knee - type d - cluster 1*.

(b) Fature contributions of the correct prediction of cluster *Knee - type c - cluster 0*.

Figure 9.9: *SHAP* force plots visualizing the feature contributions for two predictions made for a surgery belonging to the cluster *Knee - type c - cluster 0*.

9.3 Scheduling

We developed the clustering en classification method evaluated in the previous sections to be used during operating room scheduling. Chapter 8 explains how to use the estimated surgery duration, obtained from the clustering and classification models, to schedule surgery time slots. To schedule surgeries based on their estimated surgery duration, we employ 8 heuristic scheduling strategies. This section discusses the scheduling performance of these strategies for the orthopedic and cardiothoracic surgery scheduling cases. Furthermore, we discuss the performance of the optional method adaptation, predicting the surgery duration separately. Specifically, this section addresses three evaluation tasks:

1. We compare the heuristic strategies in order to determine the one that is most suitable for a particular hospital case.
2. Next, we compare the best heuristic strategy for each hospital to the original hospital scheduling.
3. Finally, we compare the schedules resulting from the optional adaptation of the scheduling method, separately predicting the surgery duration, to the performance of the general method.

Objective

To determine which scheduling strategy from Section 8.1 is most suitable for each hospital case, we evaluate the performance of each strategy for both hospital cases. Chapter 5 discusses the performance of a surgery schedule and identifies 7 concrete performance measures which can be used to compare surgery schedules. To identify the most suitable scheduling strategy, we measure and compare the average performance measures per strategy.

Next, we aim to decide whether or not the cluster-based scheduling method proposed in this thesis improves upon the scheduling performed by the investigated case hospitals. To do so, we compare the performance measures of the schedules resulting from the most suitable scheduling heuristic to those of schedules developed by the case hospitals. Depending on the available data, we compare the performance of the schedules created with our scheduling method to the performance of the planned and realized hospital schedules.

Finally, we aim to determine whether or not separately predicting the surgery duration improves the general cluster-based scheduling method. To do so, we compare the performance measures resulting from scheduling using the most suitable scheduling strategy with the general method to those resulting from the adapted scheduling method (using the same strategy). Additionally, we evaluate the effectiveness of the separate prediction of the surgery duration. To determine this effectiveness, we measure and compare the MAE of the duration predictions to the MAE of the duration estimations from the surgery clusters.

Setup

We test the scheduling strategy performance by scheduling all biweekly schedules in the test data using each heuristic scheduling strategy and comparing the performance measures of these schedules. First, we estimate, for all surgeries in the test data, the duration from the predicted clusters using the clustering and classification method evaluated in the previous sections. Next we use the scheduling strategies discussed in Section 8.1 to schedule the surgeries in the test data. We do so by identifying the biweekly schedules in the test data and scheduling all surgeries in each schedule using the scheduling heuristic. The realization of the biweekly surgery schedules created this way are subsequently simulated to be able to fairly compare their performance.

For the orthopedic surgery test data, both the planned and realized hospital schedule is available. For cardiothoracic surgery, however, this is limited to the realized schedule. Similar to the hospitals created by our scheduling method, we simulate the realizations of the planned hospital schedules available for the cardiothoracic scheduling case. We compare the performance measures distributions of the simulated schedules resulting from the scheduling method to the realized schedules performed by the hospital. If possible, like for cardiothoracic surgery, we also compare these to the performance measure distributions of the simulated planned hospital schedules.

Finally, we predict for each surgery in this test data to which cluster it belongs and for the adapted method we separately predict the surgery duration. To compare the performances of these methods, we compute the duration prediction *MAE* for the cluster-based estimation and separate prediction. Next, we schedule the surgeries according to the general method explained in chapter 8 and the adapted method discussed in section 8.7. Again, the resulting schedules are evaluated by computing their average performance measures.

Execution

To obtain an overview of the scheduling performance and to be able to identify a suitable strategy for each hospital case we compare the performance measures of all 8 strategies: *SPF*, *SVF*, *LPF*, *LVF*, *SPF-BW*, *SVF-BW*, *LPF-BW*, and *LVF-BW*. We compare the following performance measures idle time, patient waiting time, overtime, undertime, utilization, postponements en factor postponements. We simulate the resulting schedules using the simulation discussed in section 8.3. This simulation uses several parameters we identified with the hospital stakeholders to mimic the hospital situation. Table 9.2 lists these parameters used to schedule and simulate and evaluate the orthopedic and cardiothoracic surgery schedules.

Table 9.2: The scheduling parameter configuration used to mimic the hospital situations at orthopedic surgery scheduling at Franciscus Gasthuis & Vlietland and cardiothoracic surgery scheduling at MUMC+

Hospital case	Surgery buffer time (min)	Preoperative patient availability (min)	Acceptable overtime (min)
Franciscus	10	30	30
MUMC+	15	120	90

Computing the performance measures of the simulated schedules, we obtain the performance per scheduling strategy. However, as for some performance measures positive values are best while for others negative values are best, directly presenting these results in summaries which are hard to interpret. Hence, to provide an easily interpretable summary of the heuristic strategy performances, we symbolize the surgery scheduling performance measures using the method discussed in section 8.4. We discuss these results with hospital stakeholders and identify the most suitable scheduling strategy.

To compare the distribution of scheduling performance measures resulting from our scheduling method to those of the hospital, we use this most suitable scheduling strategy. For orthopedic and cardiothoracic surgery scheduling we use the *SPF* and *LPF-BW* scheduling strategies respectively. To compare the performance measure distributions of the simulated schedules created with these strategies to the performance measure distributions of the hospital schedules, we illustrate these distributions in boxplots.

Finally, we use the most suitable scheduling strategies, to compare the general scheduling method and the method employing separately predicted surgery durations. Furthermore, the random forest regression used to predict the surgery duration is trained using the same cross-validation and hyperparameter tuning method used for the random forest classification model. Table 9.3

lists the hyperparameters used for the random forest regression model.

Table 9.3: The best random forest regression hyperparameters for the prediction of surgery durations, found using bayesian optimization.

Hospital case	Max depth	Max features	Min samples leaf	Number estimators
Franciscus	13	sqrt	9	447
MUMC+	11	sqrt	3	500

Results

Tables 9.4 and 9.4 list the scheduling performance of the scheduling strategies for orthopedic and cardiothoracic surgery scheduling, respectively. For orthopedic surgery, the *Bailey-Welch* strategies result in smaller idle time, overtime, and number of postponements while increasing the undertime and utilization. This comes at the cost of a larger patient waiting time. Furthermore, the strategies *SPF* and *SVF* that do not employ *Baily-Welch* result in higher idle time, overtime but also more undertime and utilization compared to the *LPF* and *LVF* strategies. The strategy for which this holds the most is *SPF*. For cardiothoracic surgery, a similar pattern appears. In this case, the strategies basing the ordering of surgeries on the processing time and variability result in the same performance measures. Again, the *Bailey-Welch* strategies result in the smallest idle time, overtime, and number of postponements while increasing the undertime, utilization, and patient waiting time. Interestingly, the postponements and factor postponements are smallest for the strategies *LPF-BW* and *LVF-BW*.

Table 9.4: The average scheduling strategy performance measures for orthopedic surgery scheduling. The performance measures Idle, Patient waiting, Under- and Over-time are denoted in minutes. The utilization and factor postponements are fractions and postponements is a number of surgeries.

Franciscus							
Scheduling strategy	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor postponements
SPF	2729.35	1154.88	72.744	3004.66	0.709	1.536	0.016
SVF	2736.59	1085.91	79.981	2986.91	0.708	1.607	0.016
LPF	2777.39	1047.41	116.96	2901.62	0.705	1.607	0.015
LVF	2773.18	1109.47	112.749	2894.33	0.705	1.643	0.015
SPF BW	2709.75	4953.19	53.964	3184.07	0.71	1.071	0.011
SVF BW	2709.75	4947.85	53.964	3173.74	0.71	1.071	0.011
LPF BW	2708.3	6316.07	56.475	3191.03	0.711	0.536	0.006
LVF BW	2711.36	6131.18	59.54	3185.25	0.711	0.571	0.006

Table 9.5: The average scheduling strategy performance measures for cardiothoracic surgery scheduling. The performance measures Idle, Patient waiting, Under- and Over-time are denoted in minutes. The utilization and factor postponements are fractions and postponements is a number of surgeries.

MUMC+							
Scheduling strategy	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor postponements
SPF	1829.22	193.585	409.869	2437.93	0.787	2.263	0.064
SVF	1829.22	193.585	409.869	2437.93	0.787	2.263	0.064
LPF	1857.57	180.478	438.183	2416.34	0.784	2.316	0.065
LVF	1857.57	180.478	438.183	2416.34	0.784	2.316	0.065
SPF BW	1816.83	2340.09	397.474	2445.82	0.788	2.263	0.064
SVF BW	1816.83	2340.09	397.474	2445.82	0.788	2.263	0.064
LPF BW	1816.83	2421.99	397.474	2445.82	0.788	1.895	0.053
LVF BW	1816.83	2421.99	397.474	2445.82	0.788	1.895	0.053

The Tables 9.6 and 9.7 list the symbolized summary of the performance comparison. In these tables, the best value is marked with a ++. Note that these tables directly follow from Tables 9.4 and 9.4.

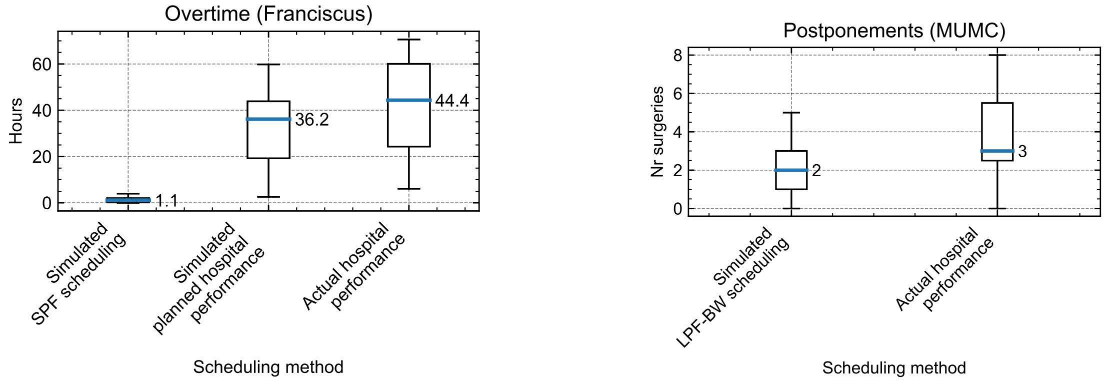
Table 9.6: The scheduling strategy performance comparison of orthopedic surgery scheduling performance measures. The average performance measures are provided in Table 9.4

Franciscus							
Scheduling strategy	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor postponements
SPF	+	++	+	-	+	- -	- -
SVF	±	++	±	-	±	- -	- -
LPF	- -	++	- -	- -	- -	- -	- -
LVF	- -	++	- -	- -	- -	- -	- -
SPF BW	++	-	++	++	++	±	±
SVF BW	++	-	++	++	++	±	±
LPF BW	++	- -	++	++	++	++	++
LVF BW	++	- -	++	++	++	++	++

Figure 9.10 shows two performance distribution boxplots, illustrating the performance of the *SPF* scheduling of orthopedic surgeries and *LPF-BW* scheduling of cardiothoracic surgeries. For the orthopedic surgery scheduling, this figure displays the overtime, whereas the number of postponements is shown for cardiothoracic surgery scheduling. A complete exposition of the performance measure distribution boxplots for the *SPF* and *LPF-BW* scheduling of orthopedic and cardiothoracic surgeries are provided in Appendix F.

Table 9.7: The scheduling strategy performance comparison of cardiothoracic surgery scheduling performance measures. The average performance measures are provided in Table 9.5

MUMC+							
Scheduling strategy	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor postponements
SPF	+	++	+	+	+	- -	- -
SVF	+	++	+	+	+	- -	- -
LPF	- -	++	- -	- -	- -	- -	- -
LVF	- -	++	- -	- -	- -	- -	- -
SPF BW	++	- -	++	++	++	- -	- -
SVF BW	++	- -	++	++	++	- -	- -
LPF BW	++	- -	++	++	++	++	++
LVF BW	++	- -	++	++	++	++	++



(a) Overtime of SPF orthopedic surgery scheduling

(b) Postponements of LPF-BW cardiothoracic surgery scheduling

Figure 9.10: The distribution of utilization performance of heuristic and hospital surgery schedules.

In Figure 9.10 the reduction of overtime in the simulated SPF schedule stands out compared to the hospital schedule performances. Furthermore, the number of postponements is smaller in the surgery schedules developed with our cluster-based method employing the *LPF-BW* strategy, compared to the actual hospital schedules. Specifically, the *SPF* scheduling improves the simulated overtime for orthopedic scheduling planning. The simulated realizations of the schedules, resulting from scheduling with the method proposed in this thesis using the SPF strategy, have a median of 1.1 hours of overtime. In contrast, the median overtime of the schedules obtained by simulating the realization of the planned hospital schedules is 36.2 hours. In fact, *SPF* scheduling outperforms hospital surgery scheduling for the case of orthopedic surgery scheduling in all performance measures. In Appendix F we observe the median *utilization* and *undertime* of the simulated method schedules increased with 0.34 and 36 hours, whereas the median *idle time*, *overtime*, on-site *patient waiting time*, and number of *postponements* decreased with 57.1 hours, 35.1 hours, 263.4 hours, and 28 surgeries per two weeks compared to the simulated hospital schedules.

The planned performance of the hospital can not be compared for MUMC, since this data is unavailable. Nevertheless, the number of postponements expected by simulating the schedules created with the *LPF-BW* strategy with a median of 2 is an improvement from the actual schedule realized in the hospital in which 3 surgeries are marked as theoretical postponements. We observe a similar improvement in all performance measures for cardiothoracic surgery in Appendix F.

The median *utilization* and *undertime* of the schedules resulting from the simulated scheduling method increased by 0.14 and 6.3 hours, whereas the median *idle time*, *overtime*, and number of *postponements* decreased by 24.1 hours, 20 hours and 1 surgery per two weeks compared to the realized hospital schedules.

Table 9.8 lists the mean absolute error for the duration estimation methods used in the different scheduling methods compared in this section. Additionally, it presents the hospital surgery duration prediction for orthopedic surgeries. This is a feature used in theclassification of surgeries into clusters in the general method and in random forest regression to separately predict the surgery duration. Furthermore, this feature itself can be used as an estimate of the surgery duration. For both hospital cases, the random forest prediction results in the duration estimates with the smallest *MAE*. For cardiothoracic surgery, however, the difference in *MAE* between the cluster average estimation method and the random forest prediction method is 2,6 minutes. In contrast to cardiothoracic surgery, the random forest predictions for orthopedic surgery durations are on average 7.3 minutes more accurate compared to the cluster average. However, the difference between the hospital prediction *MAE* and the RF prediction *MAE* is minimal, only 0.2 minutes.

Table 9.8: The duration estimation performance of the cluster based estimation, random forest prediction and if available the hospital prediction. The mean absolute error (MAE) of surgery durations is denoted in minutes.

	Franciscus		**MUMC+**	
Estimation method	Training (MAE)	Test (MAE)	Training (MAE)	Test (MAE)
Cluster average	20	20.2	62.6	57.2
Random forest prediction	10.4	12.9	51.5	54.6
Hospital prediction	12.7	13.1	-	-

Finally, Tables 9.9 and 9.10 list the average performance measures obtained with the general and adapted heuristic scheduling methods. Table 9.9 lists the performance of scheduling with SPF and predicted surgery durations for orthopedic surgeries. Table 9.10 lists the performance of scheduling with *LPF-BW* and predicted surgery durations for cardiothoracic surgeries. Note that the performance of the cluster estimate duration estimation method corresponds to the chosen strategy performance presented in the results of Section 9.3.

Table 9.9: The average performance measures obtained using SPF scheduling with cluster estimated durations and individually predicted duration estimates for orhtopedic surgery scheduling. The performance measures Idle, Patient waiting, Under- and Over-time are denoted in minutes. The utilization and factor postponements are fractions and postponements is a number of surgeries.

Franciscus							
Duration estimation method	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor postponements
Cluster estimate	2729.35	1154.88	72.744	3004.66	0.709	1.536	0.016
RF prediction	2708.54	592.591	52.536	3165.34	0.711	0.571	0.009

Table 9.10: The average performance measures obtained using LPF-BW scheduling with cluster estimated durations and individually predicted duration estimates for cardiothoracic surgery scheduling. The performance measures Idle, Patient waiting, Under- and Over-time are denoted in minutes. The utilization and factor postponements are fractions and postponements is a number of surgeries.

MUMC+							
Duration estimation method	Idle time	Patient waiting time	Overtime	Undertime	Utilization	Postponements	Factor post-ponements
Cluster estimate	1827.23	2335.59	407.88	2442.56	0.787	1.947	0.053
RF prediction	1816.83	2421.99	397.474	2445.82	0.788	1.895	0.053

For orthopedic surgery, using the random forest predicted duration instead of the cluster estimate during scheduling results in a smaller average idle time, patient waiting time, overtime, number of postponements and factor of postponements, and a larger undertime and utilization. When using the duration prediction, we reduce the average patient waiting time by approximately 500 minutes, the average overtime by 20 minutes, and avoid almost 1 theoretical postponement per schedule. Similar differences are found for cardiothoracic surgery scheduling, with the increase in patient waiting time being a notable exception. However, the difference in performance is smaller for the case of cardiothoracic surgery scheduling than for that of orthopedic surgery scheduling.

9.4 Discussion

The previous sections described the experiments performed in this thesis and presented their results. This section interprets these results, in order to evaluate the validity of the arguments provided by the previous research chapter.

9.4.1 Silhouette method

In Figure 9.1 we observe that by clustering the initial clusters into more than 8 clusters, the silhouette score starts to decay for all initial clusters. The silhouette score represents the similarity of the surgeries to the other surgeries in their own cluster versus the similarity to the surgeries in other clusters. Increasing the number of clusters increases the similarity of surgeries within each cluster. When there are still dissimilar subgroups in a clustering, increasing the number of clusters should not increase the similarity of surgeries to surgeries of other clusters. However, when increasing the number of clusters results in additional similar clusters, the similarity of surgeries to surgeries of other clusters increases. This means the silhouette score reaches a maximum when further increasing the number of clusters results in similar clusters. This maximum is reached for all initial clusters before or at 8 subclusters. Hence, we argue that the most suitable number of clusters is equal to, or smaller than 8 for the studied initial surgery clusters.

It stands out that clustering the small initial cluster *Large bones - type d* (with only 32 surgeries) into the relatively high number of 8 clusters results in the best silhouette coefficient. Subclustering a small initial cluster of 32 surgeries in 8 clusters results in many small clusters, explaining the small cluster sizes of the *Large bones - type d* clusters observed in Figure 6.3. Since we approximate surgery characteristics from the surgery clusters, small clusters indicate a sensitivity to outliers and overfitting. As discussed in 6.6, we attempt to mitigate this sensitivity and obtain meaningful clusters by filtering the identified clusters on their size.

9.4.2 Clustering distributions

The distributions of scheduling characteristics belonging to the clusters identified with our clustering method, allow us to evaluate this clustering. When two clusters have different scheduling feature distributions, we consider them distinct as they could be scheduled differently. As discussed in Section 9.1, the clusters resulting from the agglomerative clustering method indeed have distinct scheduling characteristics. Either their duration is different, as is the case for clusters *Knee - type c - cluster 0* and *Knee - type c - cluster 1*, or a different characteristic differs. For example, the cardiothoracic surgery clusters *OR: 16 - IC - cluster 0* and *OR: 16 - IC - cluster 2* have similar duration and length of stay distributions, but differ in terms of postoperative department.

Section 9.1 also discusses some remarkable distributions with either large or small distributions. These distributions are based on a small number of surgeries, which has implications when used in scheduling. Using a distribution based on a small number of surgeries (especially one with a strange distribution) can result in misleading results. By using the small clusters, we would base our scheduling estimates on a small number of surgeries, posing problems for generalization. As discussed in Section 6.6 we prevent this by omitting the smallest clusters (with a size ≥ 5) in our method. In our analysis, we thus only use larger clusters with meaningful (distinct) scheduling characteristics.

9.4.3 Classifier training

The results observed in Figures 9.4 and 9.5 provide some interesting insight in the training of the classifier used to predict the clusters for surgeries that are being scheduled. Firstly, one of the reasons we used the rolling time series cross-validation explained in section 7.1, is that it allows us to monitor whether or not the classifier needs to account for concept drift. If the surgical process would have changed in the period from which the training data originated, we would expect a sudden drop in validation accuracy for one of the train/test splits. As this drop is not observed in Figure 9.5 no concept drift is expected to implicate the random forest classification model training. Furthermore, the fact that the validation accuracies shown in Figure 9.4 converges quickly to an optimum indicates a suitable set of hyperparameters is found and used to train a classifier which is used during scheduling. Finally, the accuracy of this classifier on validation data during training is approximately 0.98 for orthopedic surgeries and 0.92 for cardiothoracic surgeries. Table 7.1 lists the performance of these classifiers on unseen test data. As the accuracy on the test data is smaller, 0.96 for orthopedic surgeries and 0.90 for cardiothoracic surgeries, the classifier is still biased to the data used in training. However, since the difference in accuracy is just 0.2 and the accuracy is at least 0.9, we argue the classifier generalizes to unseen data sufficiently to be used in OR scheduling.

9.4.4 Classification performance

Table 7.1 lists an accuracy of 0.96 and 0.90 for orthopedic and cardiothoracic classification, respectively. This suggests the classifier is predicting the clusters decently. However, due to class imbalance, simply predicting the most common cluster could already produce decent results as well. As discussed in Section 6.5, the surgeries are distributed unevenly among the clusters. In the case of orthopedic surgery, simply predicting the cluster ‘Knee - type c - 0’ results in an accuracy of 0.34. For cardiothoracic surgery, predicting the cluster ‘OR: All - IC - cluster 0’ produces a classification with an accuracy of 0.74. Note that this naive baseline is outperformed by the random forest classifier trained for both hospital cases. Additionally, Table 7.1 shows the initial cluster accuracy is not perfect. Investigating the confusion matrices allows us to understand which clusters are confused by the classification model.

Note that the classifier for orthopedic surgeries is more accurate than the cardiothoracic surgery classifier. Similarly, the random forest regression model predicting duration for orthopedic surgeries outperforms the regression model predicting cardiothoracic surgery duration. These regression models are trained with the same data and presented in Section 9.3. This indicates that for orthopedic surgeries there is a more evident relationship between the surgery features available a priori and the surgery scheduling characteristics. This is to be expected as for orthopedic surgery scheduling an accurate surgery duration prediction feature is available, *planned duration*. A similar feature is unavailable for MUMC+, since this hospital does not record their planned schedules. Table 9.9 compares this feature to the predicted surgery durations.

The confusion matrices shown in Figure 9.6, show which clusters are correctly and incorrectly predicted. As is to be expected with the reported accuracies, most surgeries are predicted to belong to the correct cluster. For cardiothoracic surgeries, however, we observe a lot of surgeries that are predicted incorrectly. The confusion matrix in Figure 9.6b shows that some small clusters are consistently misclassified. These surgeries are mostly predicted to belong to a larger cluster within their initial cluster. As the classifier seems to favor the larger clusters, it is unsuccessful in predicting each cluster accurately.

Unfortunately, both classification models do not perfectly predict the initial clusters. As is apparent from the initial cluster confusion matrices shown in Figure 9.7. For orthopedic surgery scheduling, the classifier confuses, among others, the initial clusters *Ankle - type c* and *Unkown - type d*. For cardiothoracic surgeries, the classifier even misclassifies all surgeries belonging to the initial cluster *OR: 16 - recovery* to subclusters of the initial cluster *OR: 16 - IC*. Surgeries that belong to different initial clusters can not always be scheduled similarly and might require different resources. This is problematic when this method would be applied in a clinical setting. Surgeries that are expected to belong to the wrong initial cluster during scheduling, might be scheduled in such a way that the hospital scheduling constraints are not met. By creating separate classification models for every initial cluster, this problem can be circumvented. Section 10.2.2 discusses how this method could be adapted to prevent the creation of infeasible schedules and why that was impossible for this research.

The number of surgeries with incorrectly predicted initial clusters is limited to either 1% or 2% of all surgeries. Fortunately, the problems with schedules feasibility arising from these incorrectly predicted initial clusters are not severe within the scheduling scope of this thesis. We schedule surgeries within the time slots in which they were originally performed. Hence, the surgeon and operating room apparently were available for these time slots. Also, some confusions are more problematic than others. For cardiothoracic surgery scheduling we observe surgeries that do not require ICU care after surgery are predicted to move to the ICU after surgery. In practice this would not result in infeasible schedules as a patient who does not require a bed in the ICU but does have capacity reserved can always move to the regular ward with sufficient capacity. So, despite the fact that these classification models would not suffice in a clinical application, the schedules resulting from this classification model in this research are still valid. Hence, to show the surgery clusters can be leveraged during scheduling, this random forest classification method can be used in this research. However, applying a method like this in practice requires initial cluster predictions to be perfectly accurate.

9.4.5 Joint clustering and classification performance

Section 9.2 evaluates the combined performance of the clustering and classification method. Again, the combined clustering and classification method is more accurate for orthopedic surgery scheduling compared to cardiothoracic surgery scheduling. This further indicates that for orthopedic surgeries there is a clearer relationship between the scheduling characteristics, categorized with the clustering method and predicted with the classification method, and the surgery and patient data available before scheduling. Furthermore, the surgery duration and length of stay of cardiothoracic surgeries are bigger than those of orthopedic surgeries. A duration *MAE* of 59 minutes is unacceptable for orthopedic surgeries with an average duration of 71 minutes. However, since cardiothoracic surgeries take on average 242.7 minutes ($\sigma = 132.8$), this *MAE* is still challenging but not problematic for cardiothoracic surgeries. Similarly, the *MAE* for the *length of stay* is larger than for *duration* due to the fact that the postoperative length of stay is much larger than the duration of surgeries.

The 9 orthopedic surgery for which the incorrect department after surgery is predicted stand out. This corresponds to the 9 incorrectly predicted initial clusters in Figure 9.7a. These 9 surgeries of the initial cluster *Ankle - type c*, which are supposed to go to the orthopedics department (*ORT*), are predicted to be surgeries belonging to cluster *Unknown - type d*, which are admitted to the daily admission department (*DAG*). For cardiothoracic surgery scheduling, we observe the same incorrect prediction of infrequent departments. As discussed in the previous subsection, the classification model is sensitive to class imbalance.

9.4.6 Cluster explanations

The *SHAP* feature contributions illustrated in force plots allow us to explain the opaque classification of surgeries into the clusters used in scheduling. Figure 9.9a displays the feature contributions of the incorrect prediction of a surgery belonging to the cluster *Knee - type c - cluster 0* to the cluster *Knee - type d - cluster 1*. This surgery is not predicted to belong to the latter cluster, reflected by the predicted class probability of 0.0. The one-hot encoded feature *subspecialism: Knee* increases the probability of this cluster being correct, which is understandable as the surgery does belong to a cluster of knee surgeries. The other one-hot encoded features offset this increase in predicted class probability, illustrated by their negative feature contributions 'pushing' back the class probability. These features are associated with the surgery type and department to which the patient is admitted, which are different for surgeries belonging to the predicted and correct clusters, decreasing the predicted class probability. The force plot displayed in Figure 9.9b displays the feature contributions for the correct classification of surgeries to the cluster *Knee - type c - cluster 0*. As the predicted cluster probability for this cluster is higher than the predicted class probabilities of other clusters, this surgery is predicted to belong to the correct cluster. When evaluating the classification of this surgery, this force plot allows us to discover which features contributed to this classification.

It should be noted that the *SHAP* feature contributions do not provide a causal explanation. However, they do provide an explanation of the classification of a surgery into a cluster by quantifying the amount each feature contributed to the predicted cluster probability.

9.4.7 Strategy scheduling performance comparison

As discussed in chapter 3, the schedulers at Franciscus face the challenge of managing overtime and patient waiting time while postponements are not a frequent issue. At MUMC+ however, the scheduling experts indicate managing last-minute postponements is one of their primary challenges. Furthermore, in contrast to the orthopedic surgery scheduling case, the patient waiting time is not an issue. In fact, the cardiothoracic department admits all patients that are to receive surgery to the hospital before the start of the time slot. Hence, orthopedic surgery prioritizes the performance measures: idle time, patient waiting time, overtime, undertime, and utilization. In cardiothoracic surgery scheduling, however, the scheduling experts emphasize the importance of idle time, overtime, undertime, utilization, postponements, and factor postponements.

The performance summary in tables 9.4 and 9.5 allows us to identify the most suitable heuristic for both hospital cases. For orthopedic surgery, the hospital stakeholders consider the performance measures idle time, patient waiting time, overtime, undertime, and utilization to be the most relevant. Note that strategies with the *Bailey-Welch* rule result in more patient waiting time. Since the hospital stakeholders consider the patient waiting time to be important for the quality of care, we prefer the scheduling strategies that do not employ the *Bailey-Welch* rule for orthopedic surgery. Furthermore, in order to balance the idle time, patient waiting time, and overtime, we prefer the SPF scheduling strategy over the other heuristics for this hospital case. In contrast to orthopedic surgery scheduling, cardiothoracic surgery stakeholders consider the patient waiting time (after being admitted to the hospital) to be irrelevant but struggle with the number of postponements. The *Bailey-Welch* strategies generally outperform the other strategies in terms of performance measures outside of patient waiting time. As discussed more thoroughly in section 9.3, the expected duration and variance of the duration of surgery clusters are correlated in the clustering used to schedule cardiothoracic surgeries. This results in the strategies with the same ordering (*SPF-SVF* and *LPF-LVF*) having identical performance. Since the *LPF-BW* and *LVF-BW* result in fewer simulated postponements than the *SPF-BW* and *SVF-BW* strategies, we prefer these strategies for cardiothoracic surgery scheduling.

Tables 9.6 and 9.6 provide an easy overview to present to hospital stakeholders. Since the highest value can be the best value for some performance measures (undertime and utilization) but the worst value for the other performance measures, mapping the values to good (++) and bad (–) helps to avoid confusion.

Tables 9.6 and 9.7 provide some interesting insight in the hospital cases and scheduling strategies. Understandably, the scheduling strategies employing the *Bailey-Welch* rule result in poor patient waiting time and improved idle time. As *Bailey-Welch* schedules two patients at the start of the day, the patients that is treated secondly and all subsequent patients have to wait until the previous surgery is finished but are immediately available at that moment. Note that the scheduling strategies scheduling processing time and variance in the same order result in similar results for orthopedic surgery. This can again be explained with the correlation between surgery cluster duration and the variance of this duration.

9.4.8 Hospital performance comparison

Comparing the distribution of scheduling performances allows us to evaluate the scheduling methods. The boxplots in Figure 9.10 and Appendix F show that the *SPF* and *LPF-BW* strategies outperform the other hospital strategies for orthopedic and cardiothoracic surgery scheduling, respectively. Specifically in the performance measures considered important in both hospital cases, the simulated performance has improved. Unfortunately, the planned cardiothoracic surgery scheduling performance of the hospital is unavailable can not be compared. However, for orthopedic surgery scheduling, the simulated planned schedule performance is closer to the actual hospital performance than the simulated SPF strategy performance. When the hospital manages to perform the simulated *LPF-BW* schedules exactly, it would improve their surgery scheduling. This is however unlikely due to the uncertainty involved with OR scheduling that results in the observed difference between the simulated planning and realized planning. However, if for cardiothoracic surgery scheduling this difference is similar to the case of orthopedic surgery scheduling, using *LPF-BW* will improve the scheduling.

In addition, the distribution of all other performance measures for *SPF* and *LPF-BW* scheduling (provided in Appendix F) show an improvement of both the planned and realized hospital schedules. Hence, we argue that the cluster-based scheduling method described in this thesis improves the schedules developed by the case hospitals.

9.4.9 Duration prediction evaluation

Tables 9.9 and 9.10 provide insight into the effect of separately predicting the surgery durations in the scheduling method proposed by this thesis. By predicting the duration instead of estimating it from the clusters, all performance measures observed for *SPF* scheduling of orthopedic surgeries are improved. In particular, the improvement of the average patient waiting time, overtime, and postponements stands out. For cardiothoracic surgery scheduling, however, predicting the duration has a much smaller effect.

By more accurately estimating the surgery durations, we reduce the number of times surgeries occupy the OR after surgery reserved for other surgeries. This reduction of surgeries with overtime explains the smaller patient waiting time and results in less accumulation of surgery delay. By reducing the number of surgery delay, we allow the final surgery to start earlier, explaining the reduction of overtime and the number of postponed surgeries. It should be noted, that the predictive accuracy of the random forest model predicting the surgery duration is higher for orthopedic surgery, explaining the small effect on scheduling performance. In fact, the random forest predicting the surgery duration for cardiothoracic surgeries is barely more accurate than the cluster average estimation. Table 9.8 presents the MAE of the random forest regression models compared to the cluster average estimate and hospital prediction MAE. Note that the random prediction model MAE is minimally smaller than the hospital prediction model indicating our duration prediction barely improves upon the prediction made by Franciscus Gasthuis & Vlietland.

As a result of the disappointing improvement of the duration estimation by the random forest prediction, we do not draw a conclusion from the minor difference in performance measures observed for the cardiothoracic surgery scheduling case. Nevertheless, the improvement of the performance measures observed for orthopedic surgery scheduling shows us that improving the duration estimation allows us to improve the heuristic scheduling approach studied in this thesis.

Chapter 10

Conclusion

Chapter 9 discussed and evaluated the findings obtained in this thesis. To explain the contributions made by this work to the field of hospital operating room scheduling, this section positions and interprets these results. Section 10.1 outlines the contributions this research provides to the state of the art. Subsequently, Section 10.2 discusses the limitations and shortcomings of the research presented in this thesis and explores the possible future work needed to overcome these limitations. Finally, Section 10.3 summarizes the lessons learned to improve the operating room scheduling for the orthopedic and cardiothoracic surgery scheduling case studies.

10.1 Contributions

This thesis contributes a general method that can be used to leverage surgery clusters during operational surgery clustering. Even though both the clustering and classification methods provide imperfect predicted surgery clusters, we have shown surgery clusters can be used to improve hospital operating room scheduling. Additionally, this scheduling method is successfully extended with optional methodologies including a way to deal with unplannable care demand for cardiothoracic surgery scheduling at MUMC+, a separate surgery feature prediction approach and classification explanation method. This work serves as an extensible framework for future research and shows a method with which surgery clusters can be leveraged in operating room scheduling. We proposed a system of components that allows us to improve hospital operating room scheduling by estimating surgery scheduling characteristics from predicted surgery clusters. However, every component of this model can (and should) be substituted when a more appropriate model is available.

Section 1.1 introduced the general research goal addressed in this thesis. We set out to show surgery clustering can provide surgery scheduling characteristics that can be used to improve hospital operating room scheduling. To this end, we aimed to develop 3 methods:

1. A surgery clustering method able to distinguish clusters with meaningful scheduling differences. The clustering method we developed, which is discussed in Chapter 6, is able to find clusters with distinct surgery characteristics but also results in clusters with very uneven sizes. The agglomerative clustering suffices for a proof of concept but does not produce a reliable clustering methodology to be used in clinical operating room scheduling.

2. A classification method able to predict the surgery clusters for surgeries that are to be scheduled. This classification model is discussed in Chapter 7. Similar to the clustering method, the classification model is not perfect but for the proof of concept, this classification model suffices. This classification is accurate enough so that no infeasible schedules arise within this scheduling scope. However, before a classification model is can be used in practice,

it needs to predict every cluster reliably.

3. A scheduling method able to leverage the surgery scheduling characteristics provided by the predicted surgery schedules. To identify a suitable strategy per hospital case, we have developed a scheduling method and a set of performance measures with which we can compare heuristic scheduling strategies. Furthermore, we have shown the resulting, most suitable, scheduling strategies outperform the current hospital scheduling. Specifically, despite the unbalanced clustering and lacking classification methods, the developed heuristic scheduling method is able to leverage the predicted cluster scheduling characteristics and outperform the original hospital schedules.

10.2 Limitations

Before concluding which contributions this research brings, this section discusses where it falls short and how future research could continue this work. First, we explain the limitations of the problem scope in Section 10.2.1. Next, Section 10.2.2 discusses the limitations of the employed research method. Section 10.2.3 subsequently highlights the importance of domain knowledge for operating room scheduling. Finally, Section 10.2.4 discusses the limitations of the investigated hospital cases.

10.2.1 Limitations problem scope

We chose the scope of the research problem, to provide a proof of concept, not to build a clinically adoptable tool or best practice. To make use of the limited available data, but still leverage surgery clusters in operating room scheduling, we limit the scheduling problem to schedule the surgeries within the time slots in which they were originally performed. This allows us to ignore some of constraints, posed on the scheduling problem by resources that have to be available. For example, the availability of a qualified surgeon for every surgery can be inferred from the fact that this surgeon was available during the original surgery time slot. However, this scope limits the possible scheduling solutions. When scheduling biweekly schedules, allowing each surgery to be performed everywhere when the required resources are available, would result in more degrees of freedom and possibly better schedules. To do so, however, would require the scheduling method to manage the constraints posed by these required resources. These constraints are difficult to define for many hospital cases. This thesis shows, by maintaining the same clusters in each time slot but reordering them, that even when some constraints are unknown we can already improve the operational surgery schedule.

In reality, even scheduling two weeks is a simplification. Operating room scheduling typically is a dynamic problem as uncertain demand or unplannable care requires surgeries to be rescheduled. For MUMC in particular, the schedule is frequently revised and online scheduling decisions need to be made. Our method is currently not able to take the unplannable care, like emergencies, into account directly. However, by improving the surgery schedule we also reduce the probability that when unplannable care arrives this results in adverse scheduling outcomes like postponements.

Developing a method that implicitly manages such uncertain demand is an important next step. In the future, extending the scheduling scope used in this thesis by developing a method that can schedule surgeries outside of their original schedules or in an online fashion is necessary to work towards a mature clinical tool. For this, the heuristic scheduling method used in this thesis is insufficient, and more advance scheduling methodologies need to be explored. Mathematical programming is a promising approach that might be used to efficiently schedule surgeries based on cluster characteristics, taking the complex constraints which arise when scheduling outside of the original time slots into account [49]. Moving towards the dynamic online setting would require an approach that is capable of leveraging an expected surgery demand while keeping track of the current schedule state. Combining the cluster-based estimation of surgery characteristics with such methods, like for example Markov decision processes [14], provides another interesting topic of further study.

One final drawback of the scheduling scope considered in this thesis is the fact that it allows us to leverage just one of the scheduling characteristics used to define our surgery clusters. The only scheduling characteristic used by our heuristic scheduling method is the *duration*. Both the *postoperative length of stay* and *department after surgery* can be used to manage the patient flow to postoperative wards. However, the capacity of these wards is usually managed per day as the length of stay typically exceeds 24 hours. Since our method schedules the surgeries on the day they were originally scheduled, the required postoperative ward capacity is minimally altered. So again, extending the scheduling scope in future research and using a more advanced scheduling method, allows us to leverage all cluster features found in Section 9.1. Furthermore, this can also be seen as a separate step in the scheduling optimization process. We could manage the postoperative patient flow by optimizing the allocation of surgeries to time slots in a separate optimization step. After this step, the operational scheduling proposed in this thesis can still be used to efficiently schedule the allocated time slots.

10.2.2 Limitations research approach

The scheduling method proposed in this thesis shows we can leverage surgery clusters in operational operating room scheduling. However, this approach comes with several limitations, that are discussed in this Section. First, we discuss the limitations associated with the used performance measures and clustering method. Next, we explain the problems related to not including the clustering in the cross-validation method. Furthermore, we discuss the confusion of initial clusters, potentially resulting in infeasible schedules. Next, we discuss the drawbacks associated with the imbalanced classification task, simulation method, heuristic scheduling approach and biweekly queue selection. Finally, we explain the drawbacks of using *SHAP* to explain cluster predictions.

Limitations performance measures

To develop the performance measures used to evaluate our scheduling method, we used a Delphi study. Chapter 5 discussed how we obtained these performance measures from the scheduling goals identified with hospital stakeholders. Section 9.3 presents the scheduling performance measures relevant to our problem scope obtained by scheduling with the scheduling method proposed in this thesis. However, the Delphi study marked some performance factors as relevant which could not be used to evaluate our scheduling. These performance factors, like surgical staff stress, are important to consider when extending the method scope. Moreover, the performance measures we use to evaluate schedules are all aggregated measures, i.e. either sums or averages of individual scheduled surgery features. This provides a clear overview of the general scheduling performance but does not allow for the evaluation of the quality of the schedule for individual surgeries and patients. In future research, these aggregate features can be extended with individual surgery performance measures like the *maximum patient waiting time* to be able to optimize individual patient scheduling performance as well.

Limitations clustering method

Section 9.1 discusses the clustering of surgeries into clusters with distinct scheduling characteristics to be used in scheduling. As discussed in section 9.4, these clusters do indeed represent groups of surgeries with distinct *duration*, *postoperative length of stay*, and *department after surgery*. However, the clusters found using our method are not ideal. Primarily, our clustering approach results in clusters containing a small number of surgeries. We observe that our agglomerative clustering method clusters the surgeries belonging to small initial clusters into more subclusters than the surgeries belonging to large initial clusters. This results in notably small subclusters and indicates a sensitivity to outliers and overfitting. To improve the clustering method used in this research, future research could take a few approaches:

1. A different heterogeneous distance measure could be used. We used the Gower distance, explained in section 2.1.1. However, this distance measure favors categorical features as it assigns the maximal distance that can be obtained with continuous features to observations with different categorical features. When surgeries have infrequent categorical features, like uncommon postoperative departments, this results in small clusters. Future research could tackle this problem by using a heterogeneous distance measure that takes the cardinality of categorical feature values into account, like the *Heterogeneous Value Distance Measure (HVDM)* [39].

2. A different linkage criterion could be investigated. We used average linkage, instead of complete linkage to decrease the sensitivity to outliers but still observe small clusters. Ward's criterion, utilizing Ward's minimum variance method, considers both within- and between-cluster distance [50]. Ward's minimum variance method, however, can only be used with the Euclidean distance. It can however be generalized to different distance measures [50]. Using a similar generalization for the heterogeneous distance measure in future research could result in a linkage criterion less sensitive to outliers than the average linkage.

3. A more robust clustering evaluation method could be used. In section 9.1 we manually evaluated clusters by visually determining whether or not cluster distributions are distinct. This approach is subjective and ideally, one would use a statistical test to test the hypothesis the empirical distributions of two clusters are drawn from the same distribution. The *Kolmogorov-Smirnov test* is a nonparametric test testing this hypothesis [51]. By using a statistical approach, future research could more objectively determine if the identified surgery clusters indeed differ.

Including clustering in cross-validation

To evaluate the classification method, we used a rolling time series cross-validation approach, discussed in Section 7.1. This allows us to train a model that generalizes to unseen surgeries. The combined clustering and classification method used to estimate the scheduling characteristics is discussed in Section 9.2 uses the same training and test data split. However, due to the unsupervised nature of the clustering problem, we do not use this split during clustering itself. Also, the classification model requires a ground truth, the correct cluster labels, to predict. This implicates the bias of our estimation method as the scheduling characteristic estimations for surgeries in the test set are based on the clusters containing these surgeries themselves. The estimation used in scheduling should not be based on surgery information that is unknown a priori, especially not the actual scheduling characteristic that is being estimated. By introducing another test set, one that is not used in clustering, we circumvent this problem and allow for an unbiased evaluation of the estimation and scheduling method. We should omit this additional test from clustering and use it to evaluate the estimation and scheduling method. To evaluate the classification model by itself, we should use the other test set which does contain ground truth cluster labels. Since this method adaptation evaluates the scheduling on the data which has no cluster labels available, the classification performance on this test data is unknown. The available data for this research was limited in size and reserving an additional test set limits the data that can be used during training even further. Furthermore, we do not use the smallest clusters (for which the estimates would be most biased) to train our classification model. Also, the clusters used for the estimation contain more surgeries with similar scheduling characteristics. Hence, the effect of the realized scheduling characteristic on the expected characteristic is expected to be limited. However, in future method designs, we recommend incorporating this additional test set when sufficient data is available.

Initial cluster confusion

As shown in Section 9.2 the initial cluster prediction evaluation results show that the initial clusters are not perfectly predicted. As explained in the discussion of these results, this could result in infeasible operating room schedules. Assuming initial clusters are not known a priori allows us to predict them incorrectly. For orthopedic surgery, however, this is not the case, the surgery subspecialism and admission type are available before scheduling. The initial cluster features for cardiothoracic surgery are less convenient. The operating room in which the surgery can possibly take place is available before scheduling, but whether or not the patient requires intensive care after surgery is not. Before a surgery takes place, surgery schedulers know which patients are expected to require ICU care, but due to complications, additional ICU demand may arise during the surgical process. Since this IC feature was not known a priori we developed a single classification model, which predicts the final surgery cluster and, in doing so, also predicts the corresponding initial cluster. Should the initial clusters be available a priori, we would design a separate classification model for each initial cluster. As discussed in Section 9.4, the misclassified initial clusters do not result in invalid schedules. However, to prevent this from happening in future cluster-based scheduling cases, we recommend incorporating separate classification models per initial cluster. For cases in which the initial cluster feature, like the ICU capacity requirement, are not available a priori, we suggest using an expected component in the initial clustering and including the actual feature in the clustering as a scheduling characteristic.

Imbalanced classification labels

In addition to the misclassified initial clusters, the classification model also confuses surgeries within the initial clusters. Section 9.2 presents the classification results and as discussed in Section 9.4, the small clusters are predicted poorly. The fact that the clustering results small clusters, results in a class imbalance during training of the classification model. To improve the small cluster prediction performance, future research could use methods that balance the training data. For example, the *synthetic minority oversampling technique (SMOTE)* provides artificial samples for underrepresented cluster labels [52]. Another approach would be not to focus on accuracy but a different prediction performance measure, like *balanced accuracy* during training [53]. The Bayesian hyperparameter tuning currently optimizes accuracy, in future research the effect of optimizing measures like *balanced accuracy* could be investigated.

Limitations schedule simulation

To evaluate the performance of planned schedules, we use a simulation method that allows us to compare simulated realizations of planned surgery schedules and actual realization. This simulation method, discussed in Section 8.3, approximates a schedule realization by greedily substituting actual surgery durations. This way, it accounts for the uncertain duration of surgery schedules. However, we fail to simulate some additional uncertain scheduling effects. This results in the discrepancy between the simulated hospital performance and the actual hospital performance observed in the results presented in Section 9.3. Within the scope of this research, the chosen simulation provides a fair comparison between the simulated schedule performance and planned hospital schedule performance. However, some additional uncertainties could be incorporated into the simulation method with additional data. For example, when patient arrival times are available, the effect of patient lateness could be modeled. Explaining the difference and adjusting the simulation accordingly in future research, would allow us to better compare the simulated schedules obtained from the proposed scheduling method to the actual hospital schedule.

Limitations heuristic scheduling

The scheduling method proposed in this thesis uses basic heuristics to obtain an improved hospital scheduling performance. The schedule performance discussed in Section 9.3 shows the selected heuristics outperform the operating room scheduling developed by the hospital. We select the most suitable heuristic per hospital case manually in Section 9.4 by comparing the relevant performance measures. Since we consider just 8 heuristics, this scheduling method is rather limited. As discussed in the previous subsection, more advanced optimization techniques are required to schedule outside of surgery time slots. In fact, we can use these techniques to improve scheduling within the scope used in this research as well. As discussed in Section 1.2 mathematical optimization techniques like *mixed integer linear programming* and *constraints programming* have been successfully applied to operating room scheduling problems [49]. By designing an objective function from the relevant schedule performance measures and formal set of constraints for every hospital case, future research could use mathematical optimization to optimize the scheduling of surgeries within as well as outside their original time slot.

Biweekly schedule evaluation

To compare the performance of the scheduling heuristics in Sections 9.2 and 9.3 we average the performance of biweekly schedules in the test data. This evaluation method is explained in section 8.5. To increase the number of biweekly schedules that we could compare, we evaluated each week in the test data twice: once with the preceding week and once with the following week. This indeed increased the number of schedules whose average is used to obtain the results in Section 9.3. However, each schedule performance measure is either a sum or average of that performance measure of the individual time slots in the biweekly schedules. As the reported schedule performance is an average of these aggregated performance values, using each week twice does make the reported schedule performances more accurate. Additionally, we use the first and last week of the test data available for evaluation only once as these do not have a preceding or following schedule, halving their contribution to the average performance measures. Since just 2 scheduled weeks are underrepresented in the final performance measures, this effect is minimal. Nevertheless, we recommend evaluating each surgery schedule week once in future research.

Limitations SHAP explanations

Section 7.4 explained the usage of *SHAP* feature contribution values to explain surgery classifications into clusters. In Section 9.4 we note these explanations quantify the amount each feature contributes to a predicted cluster probability, but do not provide a causal explanation. Additionally, the computational complexity of *SHAP* makes it difficult to quickly predict explanations. When explainability is prioritized, future research can explore more transparent classification models. For example, single decision trees provide direct and unambiguous explanations in the form of the tree structure [21]. These models do however likely lack the predictive performance or generalization of the random forest classification used in this thesis.

10.2.3 Domain knowledge dependency

In order to successfully leverage surgery clusters during operating room scheduling, this research relied heavily on domain knowledge. Both the method design and interpretation of the results required input of hospital stakeholders. This section discusses where domain knowledge proved crucial during this research and why it will remain important in future operating room scheduling settings.

When we implement the cluster-based operating room scheduling method proposed in this thesis to a new hospital case, a few things need to be configured. First, the scheduling challenges of that surgery scheduling case need to be identified with hospital stakeholders. Within these scheduling constraints, we distinguish strict scheduling constraints and flexible challenges that impact the scheduling performance but do not determine schedule feasibility. Next, we translate these strict constraints into initial clusters exploring the available data with hospital stakeholders. The flexible challenges can be taken into account by the scheduling method and may require an adaptation of this method. To be able to schedule and simulate these surgeries, we require the *surgery buffer time*, *preoperative patient availability*, and *acceptable overtime* scheduling parameters, characterizing the case scheduling situation. By leveraging domain knowledge this way, we are able configure a new hospital case with which we can investigate the cluster-based scheduling method.

Since we need to translate the constraints and available data for a new hospital case into initial clusters, our method has to be configured using domain knowledge. This means that developing a cluster-based scheduling application requires hospital participation. However, due to the differences between hospital departments and OR scheduling situations, a universal method is unable to cope with all specific scheduling constraints.

10.2.4 Case hospital limitations

The research discussed in this thesis used patient, surgery, and scheduling data provided by the Franciscus Gasthuis & Vlietland and MUMC+ hospitals. However, both hospital cases face different scheduling challenges and have different data available.

Cardiothoracic surgery scheduling is complicated by the large and uncertain nature of their surgeries. Additionally, this department faces unplannable care such as emergencies. Managing this unplannable care requires scheduling surgeries in other time slots than their original time slots. This larger scope was impossible to investigate for MUMC+ due to missing information. For MUMC+, we identified the additional hard scheduling constraints possible surgeon availability. However, no data is available regarding which surgeons were available when. So rescheduling surgeries outside their original time (in which a different surgeon might not be available is not possible). Furthermore, for the evaluation of our scheduling method, we compared the performance of our method to both the hospital planning and the realization of this planning. This surgery planning is unavailable for MUMC+, resulting in the missing simulated planned surgery schedule realizations in the results of Section 9.3.

Similar to the orthopedic surgery scheduling case, the surgeon availability data is missing for orthopedic surgery scheduling. However, we know which type of surgeries are at least possible in each time slot. When in a time slot at least one surgery of a particular subspecialism is performed, this time slot is available for surgeries of this subspecialism. Hence, future research could use the data available for orthopedic surgery scheduling in Franciscus Gasthuis & Vlietland to investigate a scheduling method allowing for surgeries to be scheduled outside of their original time slot.

10.3 Case hospital recommendations

In the previous sections, we have concluded the operating room scheduling research by discussing its limitations and contributions. We used orthopedic surgery scheduling and cardiothoracic surgery scheduling at Franciscus Gasthuis & Vlietland and MUMC+ as hospital case studies, respectively. While we did not develop a method that can be directly applied at either of these hospitals, schedulers can use some relevant results to improve their surgery scheduling. This section summarizes the lessons learned for the case hospitals.

10.3.1 Orthopedic surgery scheduling, Franciscus Gasthuis & Vlietland

Hospital stakeholders generally already consider orthopedic surgery at Franciscus Gasthuis & Vlietland to be efficient. The heuristic scheduling method discussed in section 9.3 however outperforms the original orthopedic surgery schedules in terms of the performance measures considered relevant in orthopedic surgery scheduling. By scheduling using the *SPF* strategy, we improve utilization, idle time, overtime, undertime, and patient waiting time simultaneously. Our scheduling method used the estimated surgery duration based on the surgery clusters. These predicted clusters are currently not available to hospital schedulers. However, this hospital case has a duration estimate available: *planned duration*. We used this estimate as a classification and regression feature and as discussed in Section 9.3 it also provides a duration estimate. By scheduling each time slot using *SPF* when all surgeries that will be performed in that time slot are known, the orthopedic surgery scheduling department is already expected to improve their scheduling performance.

10.3.2 Cardiothoracic surgery scheduling, MUMC+

For the case of cardiothoracic surgery scheduling, recommending a heuristic scheduling strategy is unlikely to solve all challenges faced by this department. Currently, the cardiothoracic department does not collect data on the planned surgery schedules but only stores schedule realizations. This makes it impossible to compare the scheduled surgeries resulting from our scheduling method with the hospital planning. Hence, we recommend the cardiothoracic surgery scheduling department to start recording the surgery planning. This does not only allow us to evaluate an experimental scheduling method but also is a first step to improving the scheduling process at that department. By collecting planned schedule information, the cardiothoracic surgery department is able to evaluate and improve its scheduling. Furthermore, it extends the patient and surgery information already collected with planned surgery characteristics. The feature *planned duration* proved valuable in orthopedic surgery scheduling and might help with developing scheduling tools for cardiothoracic surgery scheduling as well.

The scheduling experiment discussed in section 9.3 shows that the scheduling strategies *BW-LPF* and *BW-LVF* result in the smallest number of surgeries postponed last-minute. As the number of postponed surgeries is an important challenge for cardiothoracic surgery scheduling, we recommend performing the surgeries in descending order of duration and variability of duration and schedule two surgeries at the start of each time slot.

Moreover, Section 8.6 discussed a method of dealing with uncertain emergency demand. Currently, emergencies sometimes result in the last-minute postponement of scheduled surgeries. By reserving time for surgeries at a specific moment during the week and delaying all surgeries that would be postponed by emergencies to that moment, the number of surgeries that needs to be postponed is reduced. In fact, reserving time for 3 emergencies per week during the 2 studied years would have resulted in less than 5% of the weeks containing an emergency postponement. Hence, we recommend the cardiothoracic surgery department at MUMC+ to create a buffer for emergencies by reserving time at a fixed moment each week.

Acknowledgements

We would like to thank Franciscus Gasthuis & Vlietland and Maastricht UMC+ for providing us with the operating room scheduling data and close collaboration. In particular, we extend our gratitude to Bart Scheenstra, Nol Visschers, Renée van Houten, Mirjam Kerklaan and Marloes Kerklaan for their feedback during this research.

The clustering, classification, and regression methods used in this thesis are built upon the general open-source implementation of agglomerative clustering and random forest learning in the *scikit-learn* python package [54]. Similarly, data visualizations illustrating the results in this thesis rely upon the python visualization libraries *matplotlib* [55] and *seaborn* [56]. Finally, *SHAP* feature contributions are estimated using the implementation provided by Lundberg et al. (2017) [25].

Bibliography

[1] "Dutch health expenditure 10th highest in Europe." Statistics Netherlands (CBS), https://www.cbs.nl/en-gb/news/2020/47/dutch-health-expenditure-10th-highest-in-europe , accessed on 2021-07-09. 1

[2] D. A. Etzioni, J. H. Liu, M. A. Maggard, and C. Y. Ko, "The Aging Population and Its Impact on the Surgery Workforce," *Annals of Surgery*, vol. 238, pp. 170–177, Aug. 2003. 1

[3] B. Cardoen, E. Demeulemeester, and J. Beliën, "Operating room planning and scheduling: A literature review," *European Journal of Operational Research*, vol. 201, pp. 921–932, Mar. 2010. 1, 3

[4] C. T. Strömblad, R. G. Baxter-King, A. Meisami, *et al.*, "Effect of a Predictive Model on Planned Surgical Duration Accuracy, Patient Wait Time, and Use of Presurgical Resources: A Randomized Clinical Trial," *JAMA Surgery*, vol. 156, pp. 315–321, Apr. 2021. 1, 3

[5] F. Guerriero and R. Guido, "Operational research in the management of the operating theatre: a survey," *Health Care Management Science*, vol. 14, pp. 89–114, Mar. 2011. 2

[6] M. A. Bartek, R. C. Saxena, S. Solomon, *et al.*, "Improving Operating Room Efficiency: Machine Learning Approach to Predict Case-Time Duration," *Journal of the American College of Surgeons*, vol. 229, pp. 346–354.e3, Oct. 2019. 2, 3

[7] A. J. Thomas Schneider, J. Theresia van Essen, M. Carlier, and E. W. Hans, "Scheduling surgery groups considering multiple downstream resources," *European Journal of Operational Research*, vol. 282, pp. 741–752, Apr. 2020. 2

[8] M. Samudra, C. Van Riet, E. Demeulemeester, *et al.*, "Scheduling operating rooms: achievements, challenges and pitfalls," *Journal of Scheduling*, vol. 19, pp. 493–525, Oct. 2016. 3, 4, 38

[9] M. W. Mulholland, P. Abrahamse, and V. Bahl, "Linear Programming to Optimize Performance in a Department of Surgery," *Journal of the American College of Surgeons*, vol. 200, pp. 861–868, June 2005. 3

[10] B. Cardoen, E. Demeulemeester, and J. Beliën, "Optimizing a multiple objective surgical case sequencing problem," *International Journal of Production Economics*, vol. 119, pp. 354–366, June 2009. 3

[11] J. M. H. Vissers, I. J. B. F. Adan, and J. A. Bekkers, "Patient mix optimization in tactical cardiothoracic surgery planning: a case study," *IMA Journal of Management Mathematics*, vol. 16, pp. 281–304, July 2005. Publisher: Oxford Academic. 3

[12] A. van den Broek d'Obrenan, A. Ridder, D. Roubos, and L. Stougie, "Minimizing bed occupancy variance by scheduling patients under uncertainty," *European Journal of Operational Research*, vol. 286, pp. 336–349, Oct. 2020. 3

[13] B. Roland, C. di Martinelly, and F. Riane, “Operating Theatre Optimization : A Resource-Constrained Based Solving Approach,” in *2006 International Conference on Service Systems and Service Management*, vol. 1, pp. 443–448, Oct. 2006. ISSN: 2161-1904. 3

[14] J. Zhang, M. Dridi, and A. E. Moudni, “Scheduling Elective Surgeries with Markov Decision Process and Approximate Dynamic Programming,” *IFAC*, vol. 52, pp. 1831–1836, Jan. 2019. 3, 97

[15] J. Lai, C.-C. Huang, S.-C. Liu, *et al.*, “Improving and Interpreting Surgical Case Duration Prediction with Machine Learning Methodology,” *medRxiv*, p. 2020.06.10.20127910, Dec. 2020. Publisher: Cold Spring Harbor Laboratory Press. 3

[16] R. Wirth and J. Hipp, “CRISP-DM: Towards a Standard Process Model for Data Mining,” *Proceedings of the Fourth International Conference on the Practical Application of Knowledge Discovery and Data Mining*, 2000. 4, 15, 71

[17] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to data mining.* Pearson Education India, 2016. 7, 9, 11

[18] J. C. Gower, “A General Coefficient of Similarity and Some of Its Properties,” *Biometrics*, vol. 27, p. 857, Dec. 1971. 7

[19] L. Rokach and O. Maimon, “Clustering Methods,” in *Data Mining and Knowledge Discovery Handbook* (O. Maimon and L. Rokach, eds.), pp. 321–352, Boston, MA: Springer US, 2005. 8

[20] P. J. Rousseeuw, “Silhouettes: A graphical aid to the interpretation and validation of cluster analysis,” *Journal of Computational and Applied Mathematics*, vol. 20, pp. 53–65, Nov. 1987. 8

[21] M. Krzywinski and N. Altman, “Classification and regression trees,” *Nature Methods*, vol. 14, pp. 757–758, Aug. 2017. Number: 8 Publisher: Nature Publishing Group. 10, 54, 100

[22] N. Altman and M. Krzywinski, “Ensemble methods: bagging and random forests,” *Nature Methods*, vol. 14, pp. 933–934, Oct. 2017. Number: 10 Publisher: Nature Publishing Group. 10, 54

[23] W. Grossmann and S. Rinderle-Ma, *Fundamentals of Business Intelligence.* Springer, Jan. 2015. 10

[24] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” in *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’12, (Red Hook, NY, USA), pp. 2951–2959, Curran Associates Inc., Dec. 2012. 12, 56

[25] S. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*, (Long Beach, CA), Dec. 2017. 12, 13, 57, 103

[26] J. M. van Oostrum, M. Van Houdenhoven, J. L. Hurink, *et al.*, “A master surgical scheduling approach for cyclic scheduling in operating room departments,” *OR Spectrum*, vol. 30, pp. 355–374, Apr. 2008. 15

[27] B. Addis, G. Carello, and E. Tànfani, “A Robust Optimization Approach for the Operating Room Planning Problem with Uncertain Surgery Duration,” in *Proceedings of the International Conference on Health Care Systems Engineering*, Springer Proceedings in Mathematics & Statistics, (Cham), pp. 175–189, 2014. 16

[28] G. J. Skulmoski, F. T. Hartman, and J. Krahn, “The Delphi Method for Graduate Research,” *Journal of Information Technology Education: Research*, vol. 6, pp. 1–21, Jan. 2007. Publisher: Informing Science Institute. 18, 33

[29] W. E. J. Garrett, M. F. Swiontkowski, J. N. Weinstein, *et al.*, “American Board of Orthopaedic Surgery Practice of the Orthopaedic Surgeon: Part-II, Certification Examination Case Mix,” *JBJS*, vol. 88, pp. 660–667, Mar. 2006. 19

[30] J. A. D. Molina and B. H. Heng, “Global Trends in Cardiology and Cardiothoracic Surgery – An Opportunity or a Threat?,” *Annals of the Acadamy of Medicine, Singapore*, 2009. 20

[31] J. L. Argo, C. C. Vick, L. A. Graham, *et al.*, “Elective surgical case cancellation in the Veterans Health Administration system: identifying areas for improvement,” *The American Journal of Surgery*, vol. 198, pp. 600–606, Nov. 2009. 21

[32] B. Ivarsson, S. Larsson, and T. Sjöberg, “Postponed or cancelled heart operations from the patient’s perspective,” *Journal of Nursing Management*, vol. 12, no. 1, pp. 28–36, 2004. 21

[33] S. Bødker and M. Kyng, “Participatory Design that Matters; Facing the Big Issues,” *ACM Transactions on Computer-Human Interaction*, vol. 25, pp. 4:1–4:31, Feb. 2018. 33

[34] C. H. Gyldenkaerne, G. From, T. Mønsted, and J. Simonsen, “PD and The Challenge of AI in Health-Care,” in *Proceedings of the 16th Participatory Design Conference 2020 - Volume 2*, (Manizales Colombia), pp. 26–29, ACM, June 2020. 33

[35] A. Kezar and D. Maxey, “The Delphi technique: an untapped approach of participatory research,” *International Journal of Social Research Methodology*, vol. 19, pp. 143–160, Mar. 2016. 33

[36] T. Khaniyev, E. Kayış, and R. Güllü, “Next-day operating room scheduling with uncertain surgery durations: Exact analysis and heuristics,” *European Journal of Operational Research*, vol. 286, pp. 49–62, Oct. 2020. 36, 37

[37] C. Manning, P. Raghavan, and H. Schuetze, *Introduction to Information Retrieval*. Cambridge UP, 2009. 45

[38] D. Gunopulos, “Cluster and Distance Measure,” in *Encyclopedia of Database Systems* (L. LIU and M. T. ÖZSU, eds.), pp. 374–375, Boston, MA: Springer US, 2009. 45

[39] M. S. Spencer, S. C. B. Prins, and M. S. Beckom, “Heterogeneous Distance Measures and Nearest-Neighbor Classification in an Ecological Setting,” *Missouri Journal of Mathematical Sciences*, vol. 22, pp. 108–123, May 2010. Publisher: University of Central Missouri, Department of Mathematics and Computer Science. 45, 98

[40] K. K. Dobbin and R. M. Simon, “Optimally splitting cases for training and testing high dimensional classifiers,” *BMC Medical Genomics*, vol. 4, p. 31, Dec. 2011. 53

[41] M. Harries and K. Horn, “Detecting Concept Drift in Financial Time Series Prediction using Symbolic Machine Learning,” *Proceedings of the 8th Australian Joint Conference on Artificial Intelligence*, pp. 91–98, July 1996. 53

[42] D. Steinberg, “CART: Classification and regression trees,” Apr. 2009. 54

[43] M. Hoffman, E. Brochu, and N. de Freitas, “Portfolio allocation for Bayesian optimization,” in *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, UAI’11, (Arlington, Virginia, USA), pp. 327–336, AUAI Press, July 2011. 56

[44] S. M. Lundberg, B. Nair, M. S. Vavilala, *et al.*, “Explainable machine-learning predictions for the prevention of hypoxaemia during surgery,” *Nature Biomedical Engineering*, vol. 2, pp. 749–760, Oct. 2018. 57

[45] P. Abreu, C. Soares, and J. Valente, “Selection of Heuristics for the Job-Shop Scheduling Problem Based on the Prediction of Gaps in Machines,” in *Learning and Intelligent Optimization, Third International Conference*, vol. 5851, pp. 134–147, Jan. 2009. 60

[46] J. H. Iser, B. T. Denton, and R. E. King, “Heuristics for balancing Operating Room and post-anesthesia resources under uncertainty,” in *2008 Winter Simulation Conference*, (Miami, FL, USA), pp. 1601–1608, IEEE, Dec. 2008. 60, 61

[47] S. Gul, B. T. Denton, J. W. Fowler, and T. Huschka, “Bi-Criteria Scheduling of Surgical Services for an Outpatient Procedure Center,” *Production and Operations Management*, vol. 20, no. 3, pp. 406–417, 2011. 60

[48] S. Sickinger and R. Kolisch, “The performance of a generalized Bailey–Welch rule for outpatient appointment scheduling under inpatient and emergency demand,” *Health Care Management Science*, vol. 12, p. 408, Feb. 2009. 61

[49] F. Maaroufi, H. Camus, and O. Korbaa, “A mixed integer linear programming approach to schedule the operating room,” in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 003882–003887, Oct. 2016. 96, 100

[50] T. Strauss and M. J. v. Maltitz, “Generalising Ward’s Method for Use with Manhattan Distances,” *PLOS ONE*, vol. 12, p. e0168288, Jan. 2017. Publisher: Public Library of Science. 98

[51] G. W. Corder and D. Foreman, *Nonparametric Statistics: A Step-by-Step Approach.* John Wiley & Sons, 2 ed., 2014. 98

[52] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, “SMOTE: Synthetic Minority Oversampling Technique,” *J. Artif. Intell. Res. (JAIR)*, vol. 16, pp. 321–357, June 2002. 99

[53] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, “The Balanced Accuracy and Its Posterior Distribution,” in *2010 20th International Conference on Pattern Recognition*, pp. 3121–3124, Aug. 2010. ISSN: 1051-4651. 99

[54] F. Pedregosa, G. Varoquaux, A. Gramfort, *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. 103

[55] J. D. Hunter, “Matplotlib: A 2D graphics environment,” *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007. Publisher: IEEE COMPUTER SOC. 103

[56] M. L. Waskom, “seaborn: statistical data visualization,” *Journal of Open Source Software*, vol. 6, no. 60, p. 3021, 2021. Publisher: The Open Journal. 103

List of Figures

List of Tables

Appendix A

Data preprocessing

In order to extract categorical features from free text surgery features, these free text fields were tokenized and checked for relevant tokens. The following tables list the tokens used to categorize the surgeries in categories. Table A.1 contains the tokens used to determine the subspecialisms of surgeries performed by the orthopedics department in Franciscus Gasthuis & Vlietland. Table A.2 list the the tokens employed in order to label a selection of surgery types to the surgeries performed by the cardiothoracic surgery department in MUMC+.

Table A.1: The tokens used to determine subspecialisms from surgery policy free text fields collected for orthopedic surgery at Franciscus Gasthuis & Vlietland.

Subspecialism	**Associated tokens**
Wrist and hand	Pols, Wrist, tfcc, pulley, duim, carpaal tunnel, carpaletunnel, triggerfinger, pink, quervain, scaphoid, volair, TVS, CTR, radiovolair, radio-volair, ulna, trapezium, rayhack, DIP, MCP, mcp, MC1, triangulare, TFC, radiale, neuflex, vinger, brunelli, dupuytren, EPL, lunatum, Hand, hand, PIP, DIP
Knee	knee, knie, menis, kruisband, patell, TKP, journey, HTO, deep dished, hamstring, vkb, bakerse, hoge tibia, hoge tibea, MPFL, gastroc, ukp, hemi KP, cyclops
Shoulder	schouder, shoulder, TSP, cuff, PASTA, aequalis, clavicula, scopische labrum fixatie, tuberculum minus, tuberculum majus, ascend Flex, remplisage, bicepstenodese, neerplastiek, latarjet, bicep, AC res, glenoid, AC-stab, SLAP, SSP, labrumrepair, Latissimus dorsi, latissimus dorsi, tricep
Hip	heup, hip, cup, thp, Avantage arcos, trochanter, BHR, heu p, acromion, taperloc, KHP, DHS, collum
Ankle	enkel, ankle, hiel, achilles, peroneus, perneuspees, OSG, malleolus, haglund, calcaneus, calcaneo, cuboid, achillepsees, gastroc slide
Foot	voet, foot, akin, hallux, hamerteen, morton, motron, tailors, tibiale externum, metatars, teen, tmt, mtp, chevron, mallet, MT 1, MT I, MT 5, MT-5, buniectomie, bunion, TALO-NAVI, cheilectomie
Elbow	elbow, elleboog, ellenboog, olecrani, olecranon, ellboog, radiuskop, epicondylitis
Large bones	tibia, femur, humerus, onderarm, bovenarm, grote beenderen

Table A.2: The tokens used to assign specific surgery types to surgery policy free text fields collected in the MDO form before cardiothoracic surgery at MUMC+.

Surgery type	Associated tokens
TAVI	akkoord TAVI, valve in valve TAVI, inplannen TAVI, TAVI protol MUMC, TAVI Edwards, TAVI Evolut
TAVI-femoral	akkoord TF TAVI, inplannen TF-TAVI
MIDCAB	akkoord poliklinische MIDCAB, akkoord MIDCAB, akkoord LIMA-LAD, poliklinisch MIDCAB, akkoord voor CABG MIDCAB, Akkoord voor LIMA LAD, Akkoord voor MIDCAB, acceptabel voor MIDCAB, voorkeur RATS aldus RIMA-LAD via MIDCAB, MIDCAB LIMA-LAD, MIDCAB LIMA-LAd, MIDCAB LIMA-LDA, MIDCAB: LIMA-LAD, MIDCAB RATS LIMA-LAD, MIDCAB, Lima-LAD, MIDCAB D-LAD mogelijk, MIDCAB LIMA-(D)-LAD, =>MIDCAB, Akkoord bovenstaande: MIDCAB, gekozen voor MIDCAB, Akkoord: MIDCAB, Longfunctie voldoende voor MIDCAB, MIDCAB LAD, MIDCAB: LIMA-(d)-LAD, inplannen MIDCAB, MIDCAB: LIMA-LAD, Inplannen MIDCAB, MIDCAB gewoon door laten gaan, MIDCAB D-LAD, Lima-LAD, MIDCAB: LIMA-LAD, een MIDCAB mogelijk, inplannen voor MIDCAB, MIDCAB: LAD
Minimally invasive mitral or tricuspid valve replacement or repair	patient is suitable for minimally invasive approach, akkoord miniMVP, ip miniMVP, geschikt voor mini, akkoord MVP; ip mini, akkoord mini-MVP, Akkoord: mini-MVP, geaccepteerd voor mini-MVP, akkoord (mini) MVP, Plannen voor mini-MVP, mini-MVR, inplannen, =>miniMVP, Voorstel: mini-MVP, Akkoord: mini-MVRD, inplannen voor miniMVP, Voorstel: mini-MVP, akkoord: mini-MVP, akkoord mini MVP, inplannen voor miniMVR, Akkoord mini-MVR, akkoord miniAVR, dus mini-MVP, akkoord mini-MVP, Voorstel: Mini-MVP

Appendix B

Case hospital statistics

The surgery scheduling cases studied in this research each have different scheduling challenges and goals. These differences can to some extend be attributed to the differences in surgeries performed in both hospital departments. This appendix elaborates the difference in surgery population by providing descriptive statistics of the surgeries performed in both hospitals.

B.1 Orhtopedic scheduling at Franciscus Gasthuis & Vlietland

Table B.1: The surgery statistics describing the population of surgeries investigated for the orthopedic surgery scheduling case at Franciscus Gasthuis & Vlietland.

Characteristic	**(N=9452)**
Age (years)	59.5±17.5
Duration (min)	71.7±33.9
Post operative lenght of stay (hours)	41.1±70.4
BMI (kg/m^2)	28.7±5.26
Sex - no./total no. (%)	
Female	5477/9452 (57.9)
Male	3975/9452 (42.1)
Emergency code - no./total no. (%)	
False	9094/9452 (96.2)
True	358/9452 (3.8)
Surgery type - no./total no. (%)	
Clinical	6781/9452 (71.7)
Daily admission	2671/9452 (28.3)

B.2 Cardiothoracic scheduling at MUMC+

Table B.2: The surgery statistics describing the population of surgeries investigated for the cardiothoracic surgery scheduling case at MUMC+.

Characteristic	**(N=1643)**
Age (years)	67.7±10.0
Sex - no./total no. (%)	
Female	453/1643 (27.6)
Male	1190/1643 (72.4)
BMI (kg/m^2)	27.4±4.3
Duration (min)	242.7±132.8
Post operative lenght of stay (hours)	175.4±176.3
Possible OR - no./total no. (%)	
All	1290/1643 (78.5)
VH-OK16	171/1643 (10.4)
VH-OK15	107/1643 (6.5)
VH-OK18	75/1643 (4.6)
Postoperative IC required - no./total no. (%)	1619/1643 (98.5)
NHR Logistic score	6.9±8.5
NHR Euroscore 2	2.8±3.8
Emergency - no./total no. (%)	45/1643 (2.7)
Surgery type - no./total no. (%)	
Day admission	104/1643 (6.3)
Clinical admission	1539/1643 (93.7)
Referring care centre - no./total no. (%)	
Laurentius Roermond	173/1643 (10.5)
MUMC	470/1643 (28.6)
Vie Curi Venlo	202/1643 (12.3)
Zuyderland Heerlen	483/1643 (29.4)
Zuyderland Sittard	254/1643 (15.5)
Other	61/1643 (3.7)
Kidney function (clearance in ml/min) - no./total no. (%)	
Dialysis	5/1643 (0.3)
Critical (clearance < 50 ml/min)	169/1643 (10.3)
Mediocre (clearance > 51 ml/min and < 85 ml/min)	818/1643 (49.8)
Regular (clearance > 85 ml/min)	651/1643 (39.6)
Left ventricular function (ejection fraction) - no./total no. (%)	
Critical (< 21%)	15/1643 (0.9)
Good (>50%)	1132/1643 (68.9)
Mediocre (31-50%)	445/1643 (27.1)
Poor (21-30%)	51/1643 (3.1)
Urgency - no./total no. (%)	
Elective	1139/1643 (69.3)
Resuscitating to OR	4/1643 (0.2)
Emergency	56/1643 (3.4)
Urgent	444/1643 (27.0)
Thoracic aorta surgery - no./total no. (%)	
True	117/1643 (7.1)
False	1520/1643 (92.5)
Unknown	6/1643 (0.4)
Post myocardial infarct VSR - no./total no. (%)	

Continued on next page

Table B.2 – *Continued from previous page*

Characteristic	**(N=1643)**
False	1619/1643 (98.5)
True	2/1643 (0.1)
Unknown	22/1643 (1.3)
Critical preoperative status - no./total no. (%)	37/1643 (2.3)
CVA/Tia - no./total no. (%)	
True	183/1643 (11.1)
False	1432/1643 (87.2)
Unknown	28/1643 (1.7)
Cardiac health history - no./total no. (%)	
Previous myocardial infarction	
True	484/1643 (29.5)
False	759/1643 (46.2)
Unknown	400/1643 (24.3)
Comorbid disorders - no./total no. (%)	
COPD	145/1643 (8.8)
Extracardiac arterial vascular pathology	173/1643 (10.5)
Kidneyfunction disorder	25/1643 (1.5)
Pulmonary hypertension	
Severe	30/1643 (1.8)
Mediocre	218/1643 (13.3)
False	1395/1643 (84.9)
Poor mobility	
True	3/1643 (0.2)
False	1600/1643 (97.4)
Unknown	40/1643 (2.4)

Appendix C

Delphi study

This appendix summarizes the Delphi study performed to arrive at the collective opinion of a panel of hospital scheduling experts. The interviewed experts are discussed in sections 3.2.2 and 3.2.3. The Delphi study is set up in two surveys. The first survey collects the individual opinions using an open-ended questionnaire. The answers of the respondents of the first survey are collected and summarized for every question. In a second survey, the panel of experts is requested to rate the relevance of their answers to the previous answers. This way experts can adjust their own opinion based on the answer provided by peers. The rated relevances of scheduling goals, challenges and consequences are used to define a group opinion among the operating room scheduling experts. The first survey is discussed in section C.1 and the results of the second survey are provided in section C.2.

C.1 Qualitative survey

The first qualitative study is used to collect the individual opinions of a panel of hospital scheduling experts. In table C.1 the questions presented to the operating room scheduling panel. Note that these questions are open ended to stimulate the experts to formulate their own opinions. Furthermore, several questions highlight aspects of the same topic. For example, both the positive and negative consequences of good and bad scheduling are requested. This ensures the panel reflects on these aspects specifically and does not focus on one aspect in particular. The survey was completed by 5 of the interviewed experts.

Table C.1: The questions asked in the first Delphi study survey.

	Question
1	In what way does good operating room scheduling contribute to the quality of care?
2	What challenges complicate operating room scheduling?
3	What constitutes a good or bad operating room schedule?
4	What are the consequences of poor operating room scheduling for patients?
5	What are the benefits of good operating room scheduling for patients?
6	What are the consequences of poor operating room scheduling for the hospital?
7	What are the benefits of good operating room scheduling for the hospital?

C.2 Collective opinion survey

The second survey used in this Delphi study aggregates the results of the previous survey and approximates the group opinion of the surveyed panel of experts. The answers to the questions of the previous survey of all experts are summarized and returned to each individual expert in this survey. In this survey, the experts are requested to rate the relevance of the answers provided by him- or herself and his or her peers. The following subsections each provide the rated relevance of the answers collected in the first survey for one question. Since all surveyed hospital experts are Dutch, the rated responses are Dutch as well. This survey was completed by 6 of the interviewed experts.

C.2.1 In what way does good operating room scheduling contribute to the quality of care?

The hospital stakeholders were asked to indicate the extent to which they believe consequences of good operating room scheduling contribute to the quality of care. The results are displayed in figure C.1

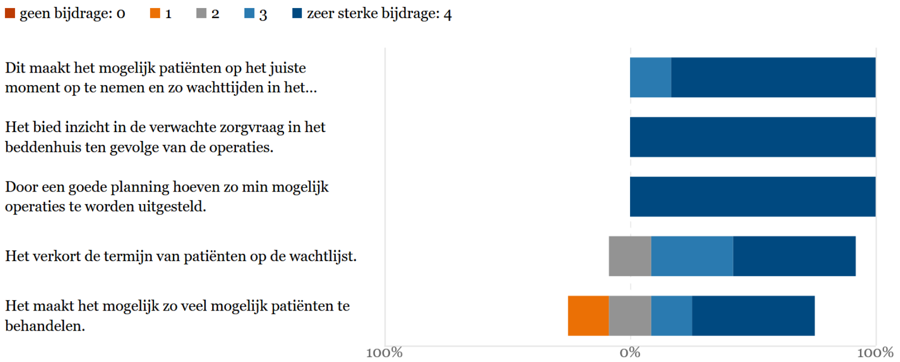


Figure C.1: The rated relevance of answers to the question: "In what way does good operating room scheduling contribute to the quality of care?". The surveyed experts could rate good operating schedules from having no contribution (0) to having a very strong contribution (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.2 What challenges complicate operating room scheduling?

The hospital stakeholders were asked to indicate the relevance of operating room scheduling challenges. Their responses are displayed in figure C.2.

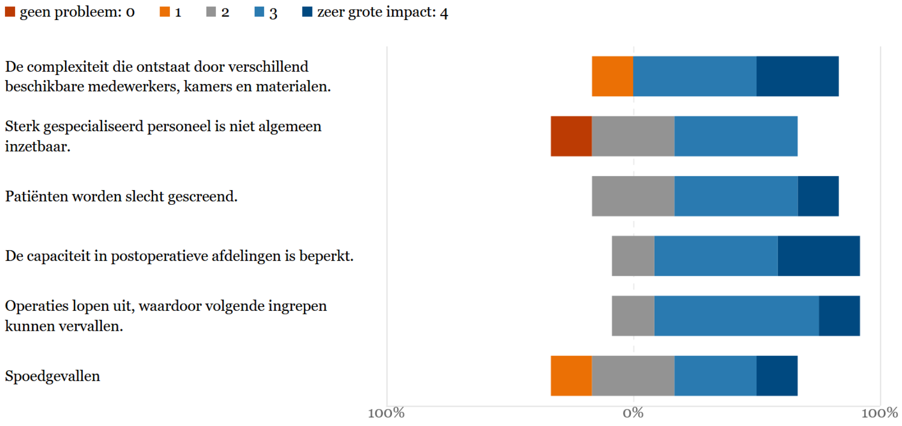


Figure C.2: The rated relevance of answers to the question: "What challenges complicate operating room scheduling?". The surveyed experts could rate the impact of scheduling challenges from being no problem (0) to having a very strong impact (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.3 What constitutes a good or bad operating room schedule?

The hospital stakeholders were asked to indicate the relevance of operating room scheduling performance measures. Their responses are displayed in figure C.3. Note that the answers to this question are used to identify the scheduling objectives in chapter 5. These scheduling objectives are subsequently used to design concrete scheduling performance measures for the method investigated in this research.

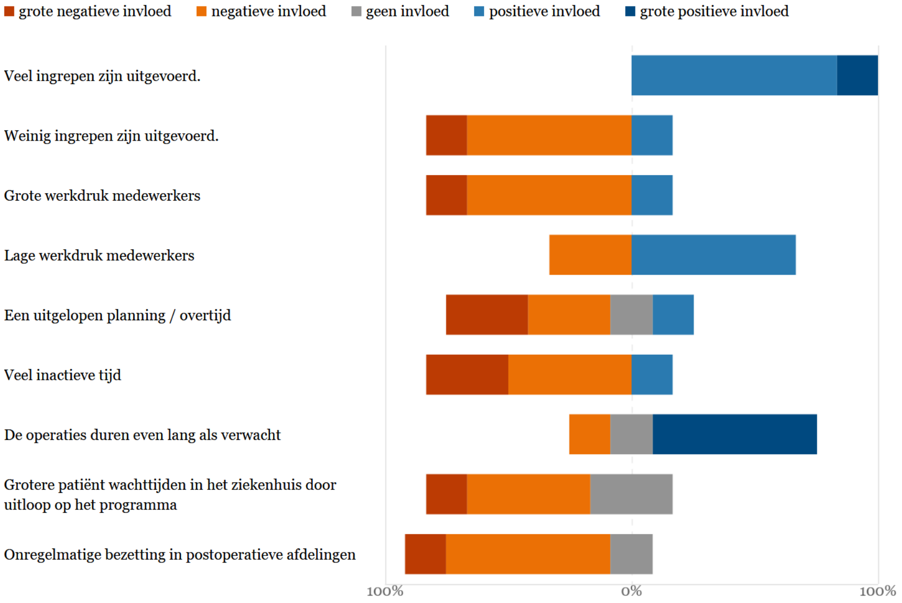


Figure C.3: The rated relevance of answers to the question: "What constitutes a good or bad operating room schedule?". The surveyed experts could rate the relevance of scheduling performance measures from having a large negative impact to having a large positive impact (4) on scheduling goodness in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.4 What are the consequences of poor operating room scheduling for patients?

The hospital stakeholders were asked to indicate the relevance of consequences of poor operating room scheduling for the patient. Their responses are displayed in figure C.4.

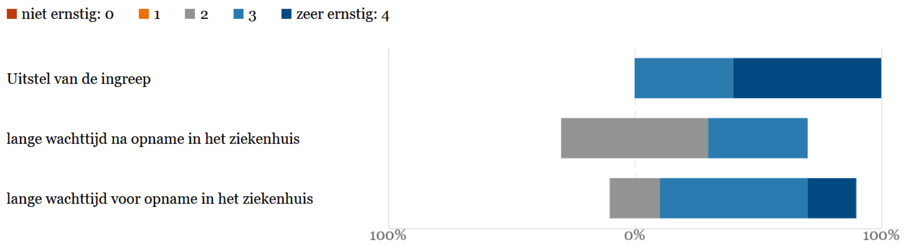


Figure C.4: The rated relevance of answers to the question: "What are the consequences of poor operating room scheduling for patients?". The surveyed experts could rate the relevance of consequences from being not serious (0) to being very severe (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.5 What are the benefits of good operating room scheduling for patients?

The hospital stakeholders were asked to indicate the relevance of benefits of good operating room scheduling for the patient. Their responses are displayed in figure C.5.

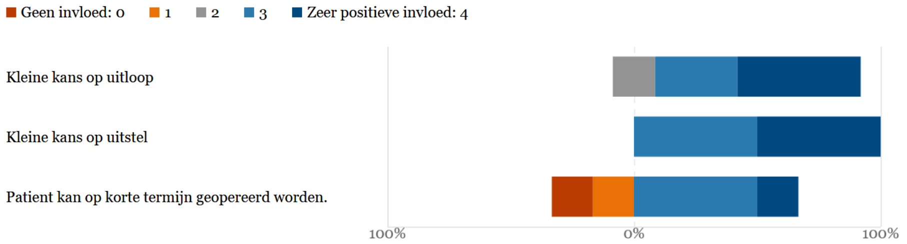


Figure C.5: The rated relevance of answers to the question: "What are the benefits of good operating room scheduling for patients?". The surveyed experts could rate the relevance of benefits from having no impact (0) to having a very positive impact (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.6 What are the consequences of poor operating room scheduling for the hospital?

The hospital stakeholders were asked to indicate the relevance of consequences of poor operating room scheduling for the hospital. Their responses are displayed in figure C.6.

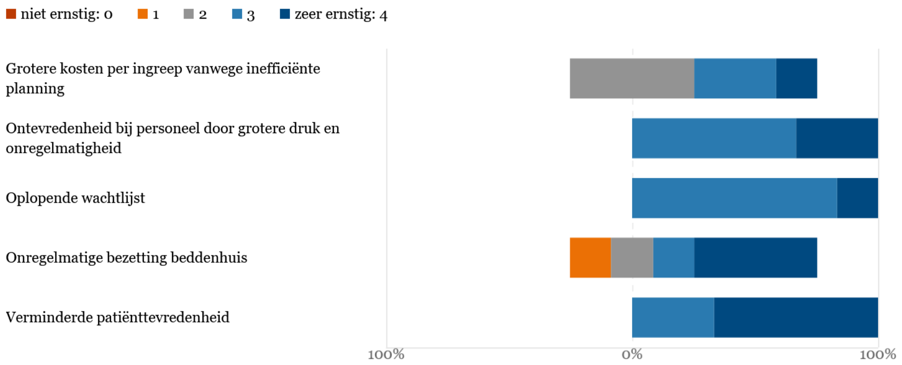


Figure C.6: The rated relevance of answers to the question: "What are the consequences of poor operating room scheduling for the hospital?". The surveyed experts could rate the relevance of consequences from being not serious (0) to being very severe (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

C.2.7 What are the benefits of good operating room scheduling for the hospital?

The hospital stakeholders were asked to indicate the relevance of benefits of good operating room scheduling for the hospital. Their responses are displayed in figure C.7.

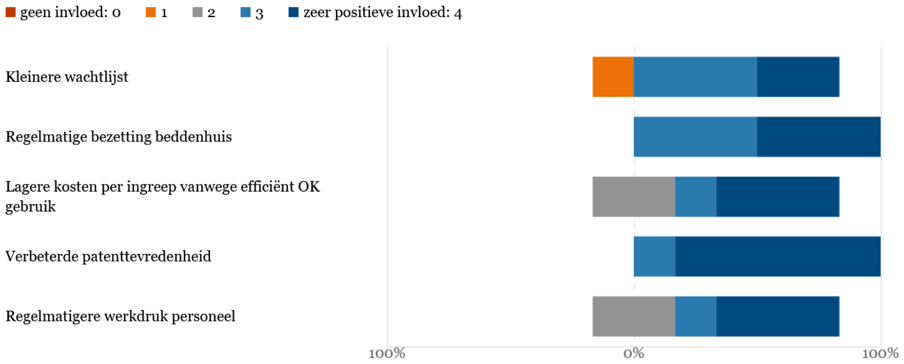


Figure C.7: The rated relevance of answers to the question: "What are the benefits of good operating room scheduling for the hospital?". The surveyed experts could rate the relevance of benefits from having no impact (0) to having a very positive impact (4) in 5 categories. This figure shows the number of times surveyed experts chose each category per answer.

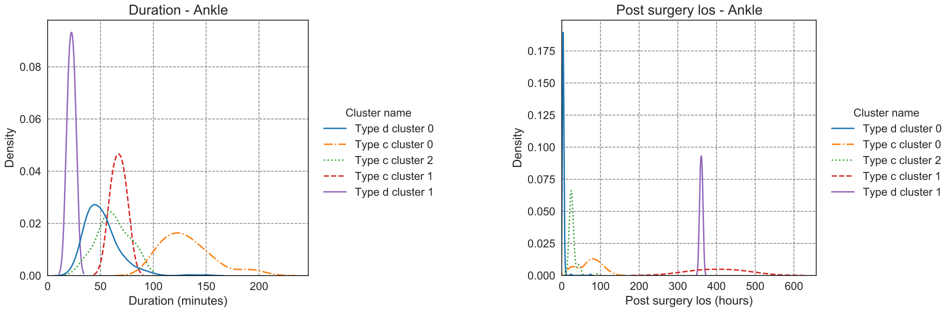
Appendix D

Surgery cluster characteristics

The results of the two step clustering method used in this research are provided more extensively in this appendix. Chapter 6 explains the clustering method in detail and contains an example of the distribution of the surgery duration for an initial cluster for both case hospitals. In section 9.1 in the evaluation, all relevant scheduling characteristic distributions are evaluated for one initial cluster. For the sake of completeness, the distribution of the rest of the characteristic distributions of the other clusters are provided in this appendix. Section D.1 provides the duration and postoperative length of stay of the initial clusters of orthopedic surgeries at Franciscus Gasthuis & Vlietland, whereas section D.2 contains the distribution of these characteristics for cardiothoracic surgeries at MUMC+.

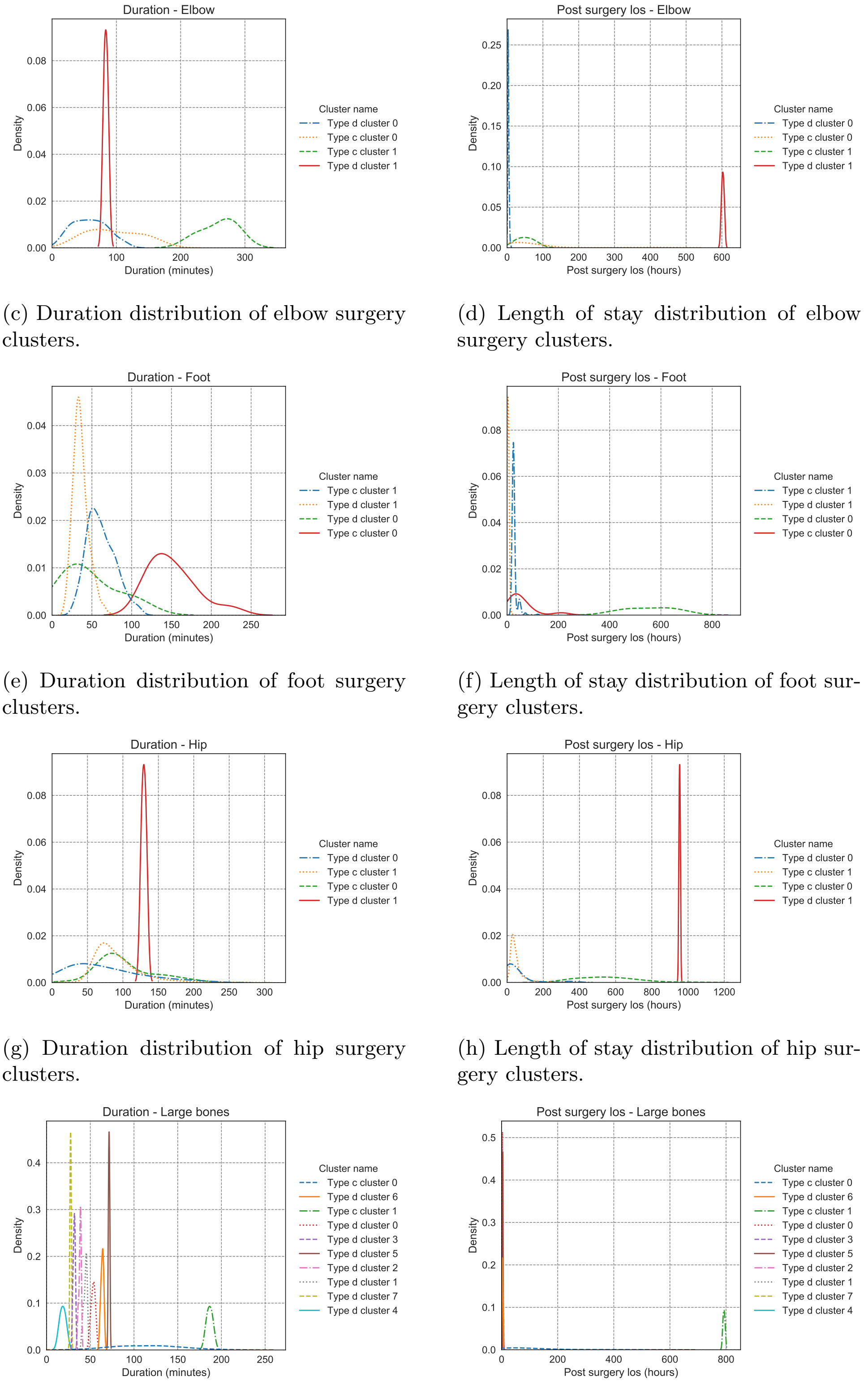
D.1 Cluster characteristics orthopedic surgeries

The distribution of the surgery duration and length of stay of orthopedic surgeries performed at Franciscus Gasthuis & Vlietland are displayed in Figure D.1. Note that the intended department after surgery is another relevant surgery characteristic, the distribution of this feature per cluster is displayed in Figure 9.2c.



(a) Duration distribution of ankle surgery clusters.

(b) Length of stay distribution of ankle surgery clusters.



(c) Duration distribution of elbow surgery clusters.

(d) Length of stay distribution of elbow surgery clusters.

(e) Duration distribution of foot surgery clusters.

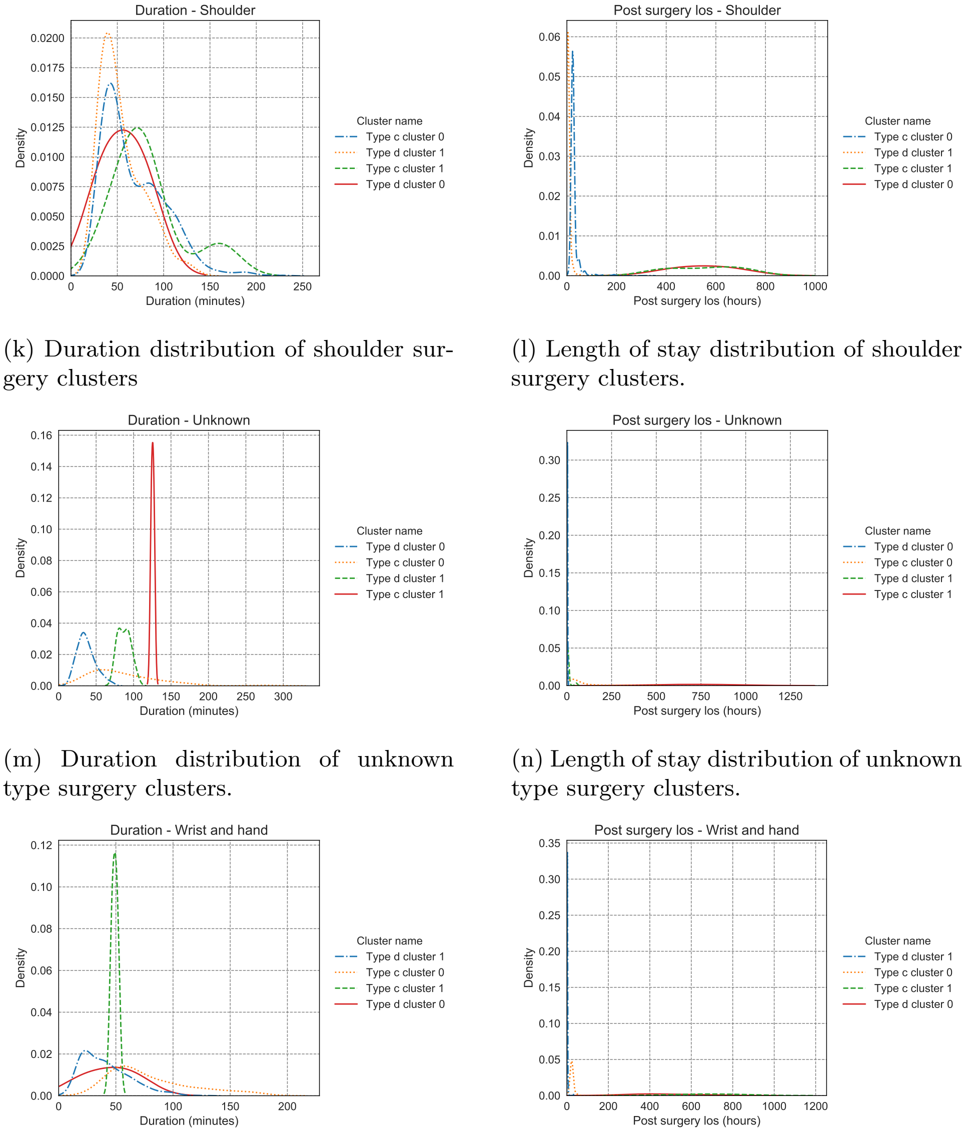
(f) Length of stay distribution of foot surgery clusters.

(g) Duration distribution of hip surgery clusters.

(h) Length of stay distribution of hip surgery clusters.

(i) Duration distribution of large bone surgery clusters.

(j) Length of stay distribution of large bone surgery clusters.



(k) Duration distribution of shoulder surgery clusters

(l) Length of stay distribution of shoulder surgery clusters.

(m) Duration distribution of unknown type surgery clusters.

(n) Length of stay distribution of unknown type surgery clusters.

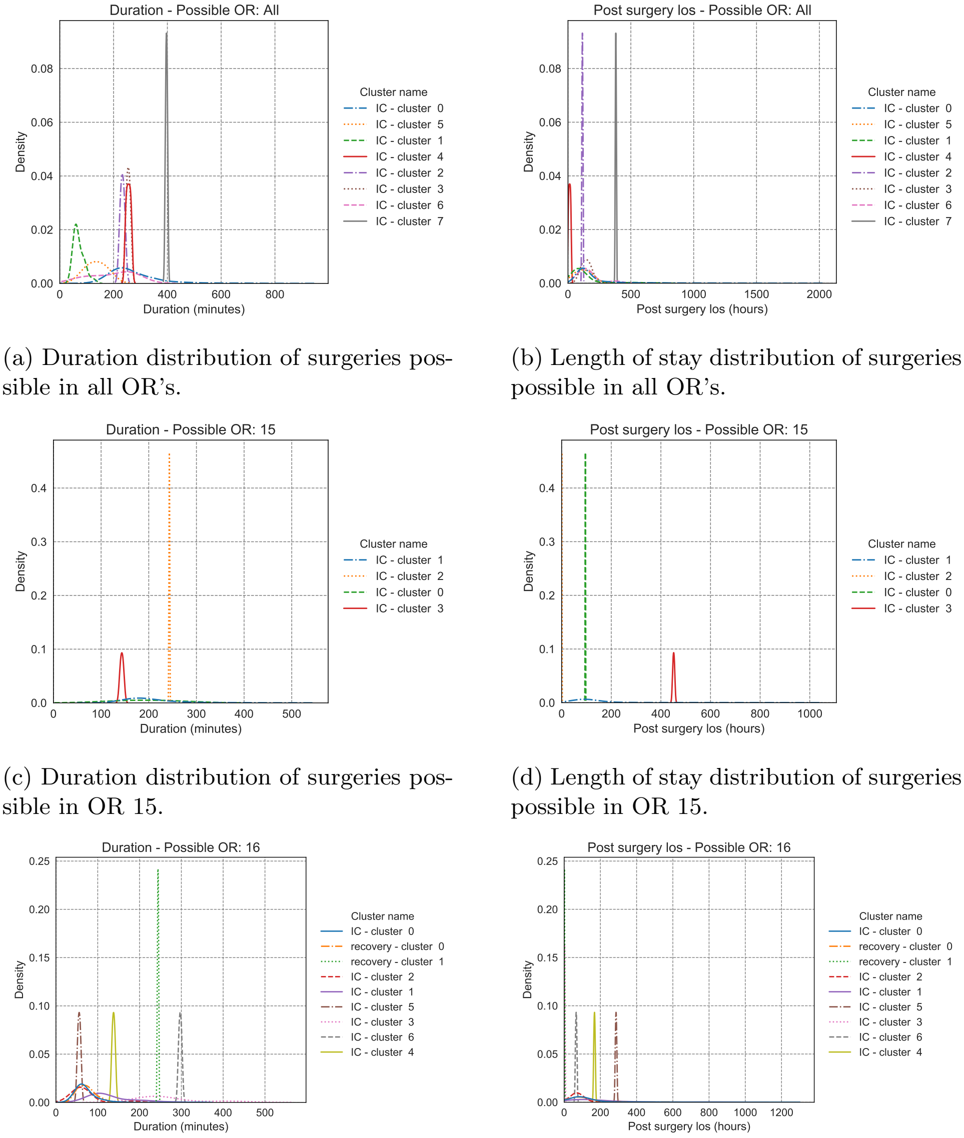
(o) Duration distribution of wrist and hand surgery clusters.

(p) Length of stay distribution of wrist and hand surgery clusters.

Figure D.1: Scheduling characteristics distributions of clusters found in the orthopedic surgeries at Franciscus Gasthuis & Vlietland.

D.2 Cluster characteristics cardiothoracic surgeries

The distribution of the surgery duration and length of stay of the clusters of cardiothoracic surgeries performed at MUMC+ are displayed in Figure D.2. Note that the department after surgery is another relevant surgery characteristic, the distribution of this feature per cluster is displayed in figure 9.3c.



(a) Duration distribution of surgeries possible in all OR's.

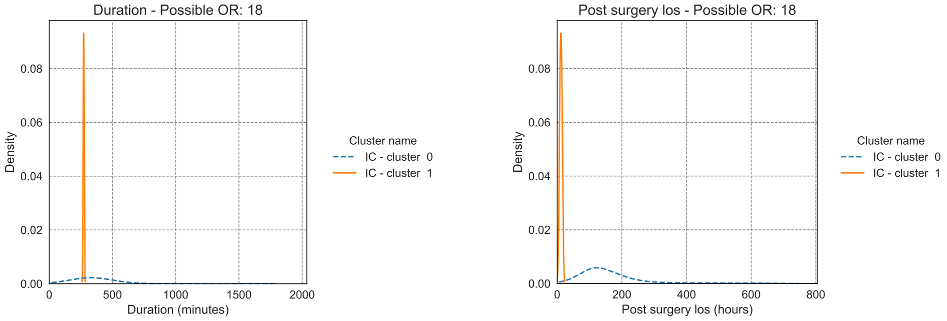
(b) Length of stay distribution of surgeries possible in all OR's.

(c) Duration distribution of surgeries possible in OR 15.

(d) Length of stay distribution of surgeries possible in OR 15.

(e) Duration distribution of surgeries possible in OR 16.

(f) Length of stay distribution of surgeries possible in OR 16.



(g) Duration distribution of surgeries possible in OR 18.

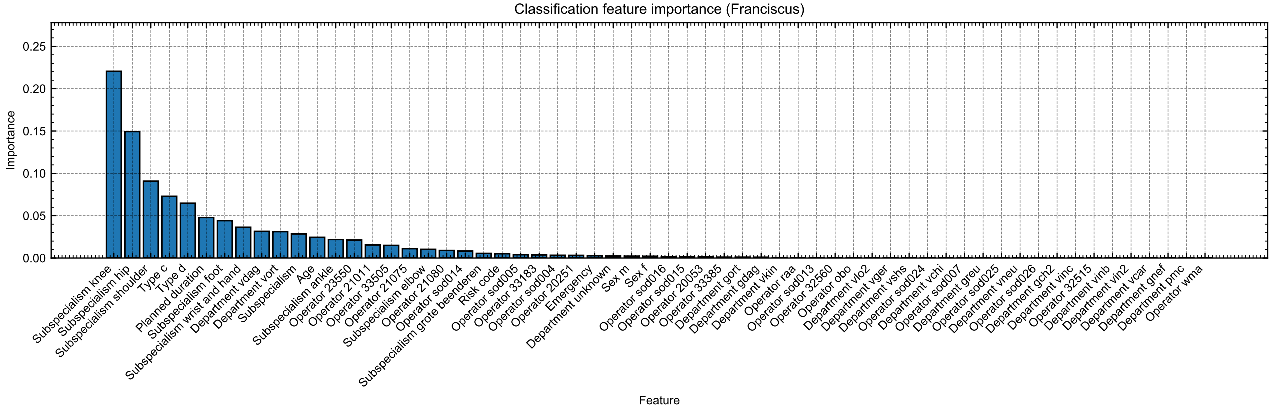
(h) Length of stay distribution of surgeries possible in OR 18.

Figure D.2: Scheduling characteristics distributions of clusters found in the cardiothoracic surgeries at MUMC+.

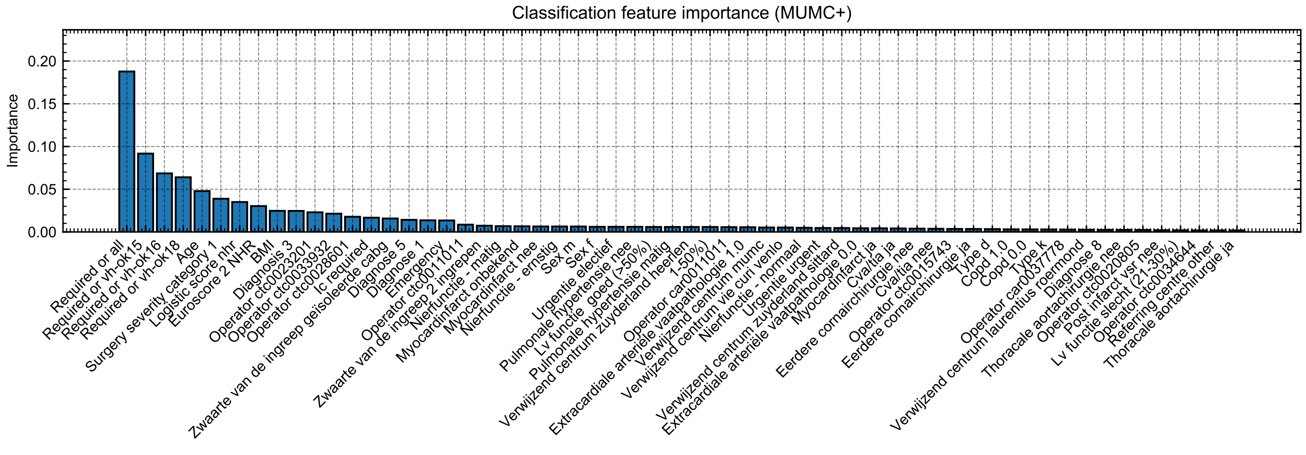
Appendix E

Feature importance classification

The feature importances of the classifiers that are trained to predict the surgery clusters for surgeries that are to be scheduled are provided in this appendix. Chapter 7 discusses the classification method and results. Among these results the feature importance of the 10 most important features per classifier are described. This appendix provides a more complete overview, of the 60 most important features per classifier. Note that this is not the complete set of features used for classification. The feature importance of features used in the random forest classifier are defined by the Gini importance. The Gini importance is the normalized reduction of the Gini criterion induced by a feature in a decision tree classifier. For the ensemble of tree classifiers used by the random forest, the Gini importance is averaged over all trees in the ensemble. Figure E.1a and E.1b show the feature importance of the classifier trained to assign clusters to orthopedic surgeries and cardiothoracic surgeries respectively.



(a) Feature importance orthopedic surgery classification



(b) Feature importance cardiothoracic surgery classification

Figure E.1: The feature importance of the random forest classification models trained to predict surgery clusters to be used in scheduling. The classification model parameters are discussed in section 7.3. These figures display the 60 features with the highest feature importance, features with smaller importance are left out for readability.

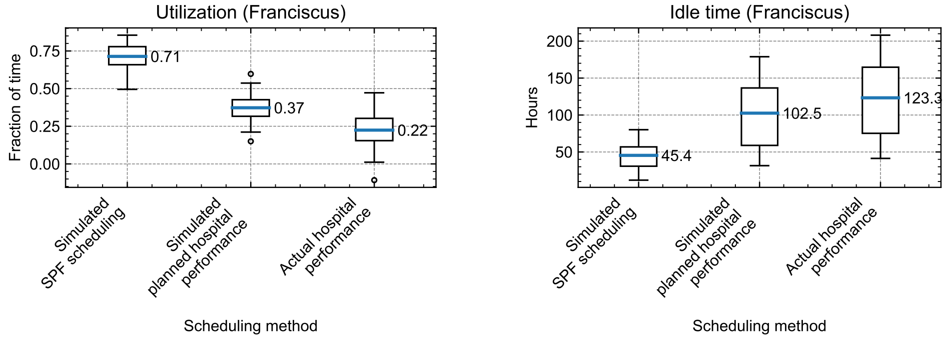
Appendix F

Heuristic scheduling performance

This appendix provides a more exhaustive overview of the scheduling performance measures obtained by scheduling with the heuristics SPF and LPF-BW in orthopedic and cardiohtoracic surgery scheduling, respectively. Chapter 8 discusses the results of scheduling with various heuristic strategies and determined the SPF and LPF-BW scheduling strategies to be the most suitable for the hospital cases studied in this thesis. Additionally, this chapter showed these scheduling strategies outperform the scheduling realized in the hospitals in terms of the scheduling performance measures identified in chapter 5. Section F.1 provides the distribution of the scheduling performance measures resulting from scheduling with the SPF heuristic for orthopedic surgery scheduling at Franciscus Gasthuis & Vlietland. Section F.2 describes the distribution of the scheduling performance measures resulting from scheduling with the LPF-BW scheduling strategy for cardiothoracic surgery scheduling at MUMC+.

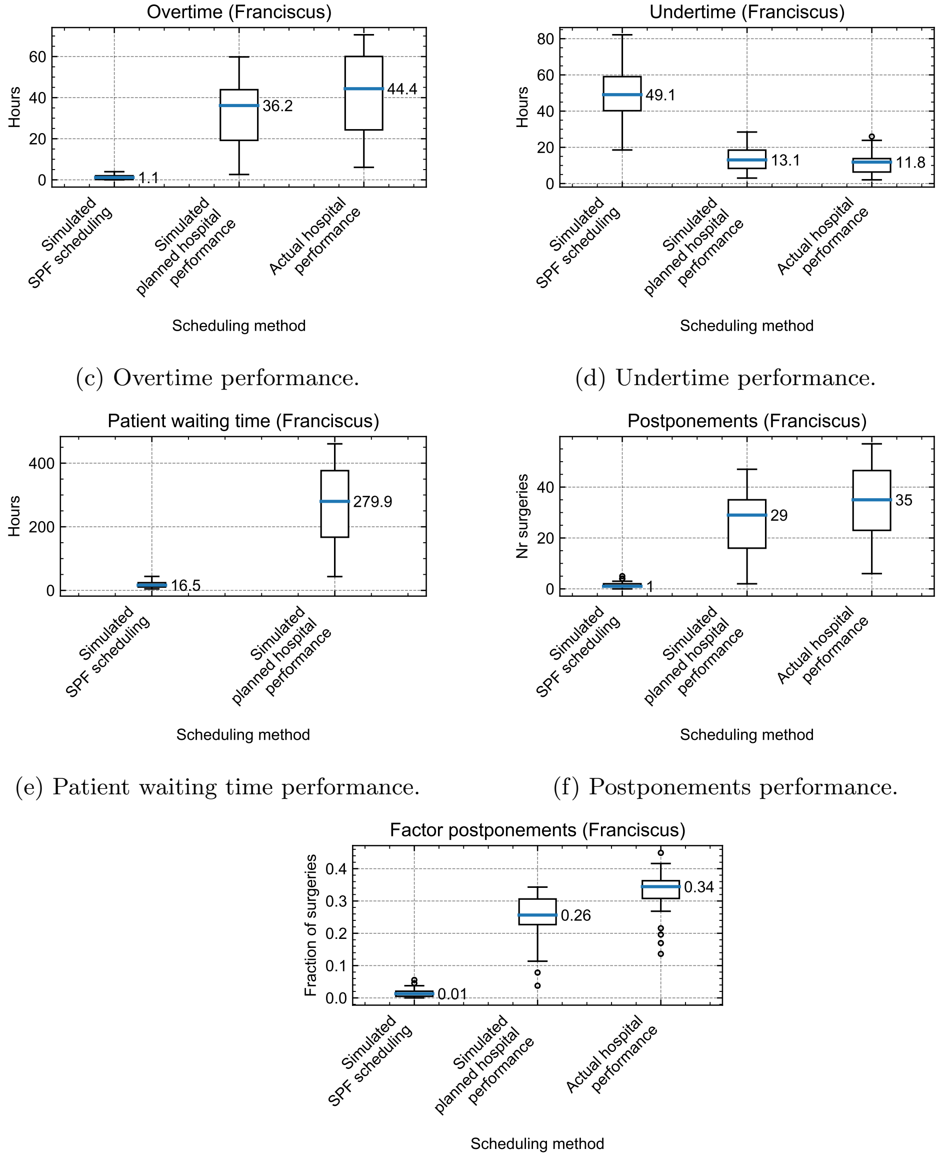
F.1 SPF scheduling performance

Figure F.1 compares the distribution of performance measures obtained by scheduling the orthopedic surgeries at Franciscus Gasthuis & Vlietland with the SPF heuristic to the planned schedules developed by the hospitals and their realization. Note that the patient waiting time is not available for the realization of the hospital schedule. The patient waiting time of a surgery is defined as the difference between the scheduled start time and the realized start time of that surgery.



(a) Utilization performance.

(b) Idle time performance.



(c) Overtime performance.

(d) Undertime performance.

(e) Patient waiting time performance.

(f) Postponements performance.

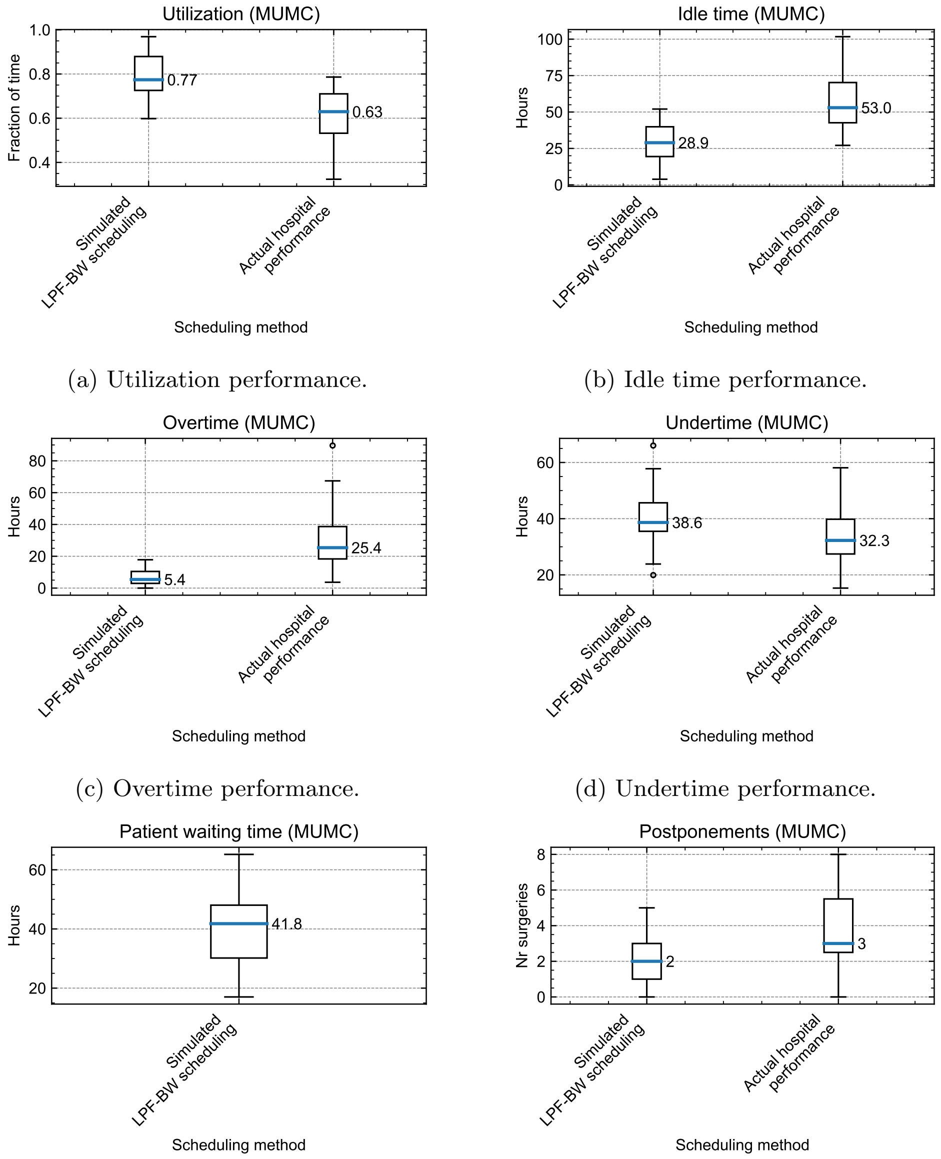
(g) Factor postponements performance.

Figure F.1: The scheduling performance measures obtained by scheduling orthopedic surgeries with the shortest processing time first (SPF) heuristic, compared to the simulated hospital planning and realized hospital planning.

As the actual hospital schedule does not have a (simulated) realization, the actual hospital schedule does not have a patient waiting time that can be compared to the patient waiting times of planned schedules.

F.2 LPF-BW scheduling performance

Figure F.2 compares the distribution of performance measures obtained by scheduling the cardiothoracic surgeries at MUMC+ with the LPF-BW heuristic to the realization of the schedules developed by the hospital. Note that for MUMC+, no planned schedules are available to simulate and compare. Furthermore, similarly to the results of orthopedic scheduling in section F.1, the patient waiting time for the realization of the hospital schedule is unavailable.



(a) Utilization performance.

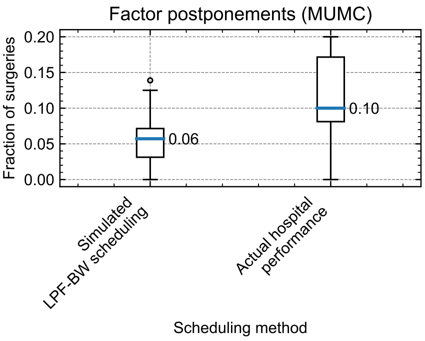
(b) Idle time performance.

(c) Overtime performance.

(d) Undertime performance.

(e) Patient waiting time performance.

(f) Postponements performance.



(g) Factor postponements performance.

Figure F.2: The scheduling performance measures obtained by scheduling cardiothoracic surgeries with the longest processing time first (LPF) heuristic using the Bailey-Welch (BW) rule, compared to the realized hospital planning.