Optimizing Scheduling for Outpatient Clinics

A combination of developing a generic tool and immediate application

A.M. Hölscher

Professor R.J. Boucherie

Professor P. Harper

Dr J. Morgan

February 14, 2016

UNIVERSITEIT TWENTE.

Preface

Proudly I present my Bachelor Project for Applied Mathematics at the University of Twente. For a period of four months I was given the opportunity to work on a challenging project of Cardiff University. Assisting the project of research associate Jennifer Morgan of Cardiff University, who was working with Cardiff and Vale University Health Board. The project provided the opportunity for me to apply already studied theory to solve a real life problem and to learn about many new parts of theory, while working in a completely new environment. Because of all this, I developed a lot in both a scientific and a personal way.

Working on my Bachelor Project in Cardiff was a great experience and there are a lot of people I want to thank for that. First of all I would like to express many thanks to Professor Richard Boucherie, who made it possible for me to go to Cardiff in the first place. At Cardiff University I was received friendlily by Professor Paul Harper and Doctor Jennifer Morgan. I would like to thank Professor Paul Harper very much for all the guidance, insight and extra materials I was provided with and above all for all the enthusiasm and positive stimulation.
For very similar reasons I would like to express my great appreciation for everything Jennifer Morgan has helped me with. On a day to day basis she was a personal mentor to me. I would like to thank Jennifer Morgan explicitly for always making time to answer my questions whether they were personal or project related, and contributing a huge part to my experience in Cardiff.

To be able to work in close contact with Jennifer Morgan, Andrew Nelson permitted me to work in his department. I am very grateful that I got the opportunity to work at the hospital site this way. I would like to thank the whole department for including me, teaching me some Welsh and making me feel welcome. In particular I would like to thank Rhian Thomas and Helen Bennett for creating the perfect working atmosphere at the office. It really made me enjoy working on my project even more.

In a part of my project I made use of the work of PHD student Geraint Palmer of Cardiff University. I would like to express many thanks to him for making so much time to help me getting familiar with his work and answering all my questions at any time. I also highly valued his patience in discussing his work in the beginning. Later on I appreciated his efforts in discussing the extra features I wanted to add to make it useful for my application. It was a pleasure to work with Geraint.

Last but not least I want to express even more thanks to Professor Richard Boucherie for making time to discuss the progress with me regularly during my time in Cardiff. I realize how much priority for my project this must have meant. On top of that I am very grateful to Professor Richard Boucherie for making sure I got everything I wanted from the project, not just regarding the project but also everything else about the experience. Altogether my project in Cardiff has resulted in a great experience that could not have been better in my opinion.

February, 2016
Lieke Hölscher

Management Summary

Outpatient clinics* from many different departments cope with the problem that they have to slot in both new patients and follow up patients. In this project a method was developed to find the strategy that best optimizes the scheduling for these outpatient clinics. The method was applied to a selection of clinics from the Ophthalmology Department of the University Hospital of Wales, but kept as generic as possible.

It needs to be started with determining the demand on the system. If the available historic data provides enough information, forecasting methods can be used. The best forecasting method for the specific time series needs to be determined. For the time series used in this project the best forecasting method turned out to be Holt's Linear Exponential Smoothing. Unfortunately in our situation there was not enough data available to be sure of previous demand. Because of governmental targets however, new patients have recently been prioritized. Thus the historic appointment data for new patients only could be accurate as historic demand and forecasting could be used on only new patients.

In case historic data does not provide enough information, a simulation, based on follow up structure, can be used to determine demand. This simulation was developed and carried out for our specific situation. An average monthly demand of 213 patients requesting an appointment was found, 45 new patients and 168 follow up patients. If forecasting new patient's demand was possible, this can be used as a part of the input for this simulation.

To be able to find the best allocation of capacity slots between new patients and follow up patients an optimization, minimizing total waiting time, can be carried out. Applying this optimization method for our situation provided us with an optimal allocation of 44 slots for new patients and 161 slots for follow up patients a month.

Finally, the capacity can be implemented in the earlier mentioned simulation. This way, it can be analysed what happens to waiting times if the capacity is divided between new patients and follow up patients. In our situation it was concluded that the waiting times were distributed very unfairly. To make it fairer the Cardiff and Vale University Health Board can try a couple of solutions. It can be chosen to alter the optimal distribution of slots. This means total waiting time will increase but the waiting times can be distributed in a fairer way. If increasing total waiting time is not a possibility it can be chosen to increase total capacity. A new allocation of capacity can be determined by applying the optimization.

The final capacity simulation can also be used to analyse the influence of other factors. In this project specific attention has been paid to the influence of dividing capacity over time. The waiting times with the current weekly master schedule for our situation were generated. It was concluded that in particular new patient waiting times can be decreased significantly by changing the weekly master schedule.

Table of Contents

1. Introduction

1.1 Problem Definition

Scheduled care in UK hospitals can broadly be grouped into inpatient and outpatient services. On the one hand there are inpatients, which require to be admitted to the hospital to be closely monitored both during the procedure and afterwards. On the other hand there are outpatients, which do not require any hospital admission. In this project it was focused on only the outpatient services.
Outpatient clinics from many different departments cope with the problem that they have to slot in both new patients and follow up patients. In many of those clinics new patients have to wait a significant amount of time before being seen for an appointment. However, this is subject to the government targets for waiting times for new patients. The appointments of follow up patients need to be fitted around the demand for new patients, which can result in them being delayed.

1.2 Research Question

In this project the main goal is to develop a model for scheduling that minimalizes the waiting times for new patients, while still seeing the follow up patients as timely as possible. This model will give information about for example how best to divide the capacity over new patient and follow up patients. It is intended to make this model as generic as possible so it can easily be extended to a model that can be used for any outpatient clinic in any department as long as there is enough applicable data.

The above can be translated into the following research question:

What strategy best optimizes the scheduling for outpatient clinics?

To help answering this question the following sub questions will be used:

1. What would happen if a fixed number of slots were assigned to new patients and follow up patients?
2. What would happen if the number of capacity slots or the division of capacity over time (master schedule) was changed?
3. What would happen if the number of no-shows could be decreased?
4. What would happen if there would be an unexpected change in demand?

1.3 Research Methodology

To get a grip on the current available data, the project was started with applying forecasting theory. The available time series of registered appointments over time is analysed.
It is intended to use an optimization program to assign (forecasted) demand to the available capacity. The objective function will be to minimize the waiting times for both new and follow up patients. To be able to do a proper optimization it is necessary to create an appropriate picture of demand.
This means that first there should be a focus on finding a way to determine demand.
After this the optimization can be carried out to find optimal capacity planning. The results will be tested by means of queuing theory and a simulation. The base scenario will be compared to results after applying minor changes to answer the sub questions.
Finally, all findings can be translated into an advice about how best to optimize scheduling.

1.4 Data Overview

Because a patient can be seen for different conditions, the follow ups for those different conditions were separated into pathways. The unique combination of patient and pathway was called a *PaPa*.

1.4.1 Clinics

For our project it was necessary to use a 'good' dataset. In this situation 'good' means that target dates are available. Target dates contain the information about when a patient should have been seen. Since we are interested in the delay, we need the target date information in addition to the actual appointment date to determine this delay. The Ophthalmology Department of the University Hospital of Wales meets this condition.
This department consists of many different clinics. A clinic means a specific combination of a certain type of clinician and available equipment. For a specific condition, it is possible that there is only a small subset of the clinics that would contain the right combination of clinician and equipment. To be able to use a manageable dataset the project only regards a few clinics of this department. In fact, the project was started with just one clinic, "OPHT103".
Unfortunately, it was discovered that, in the available dataset, there were a lot of PaPas in this clinic with only one record of an appointment. All PaPas that were recorded in clinic OPHT103 were looked up in the larger dataset with all clinic information of the Ophthalmology Department. It turned out that over 75 percent of all the appointments from those PaPas took place in only three clinics (*see Figure 1*): OPHT103 itself (22%), OPHT15 (36%) and 987/988 (18%). Apparently, for most conditions that could be treated in OPHT103, 987/988 and OPHT15 contained the right combination of clinician and equipment as well.
Therefore, in this project the data from these three clinics combined was used.

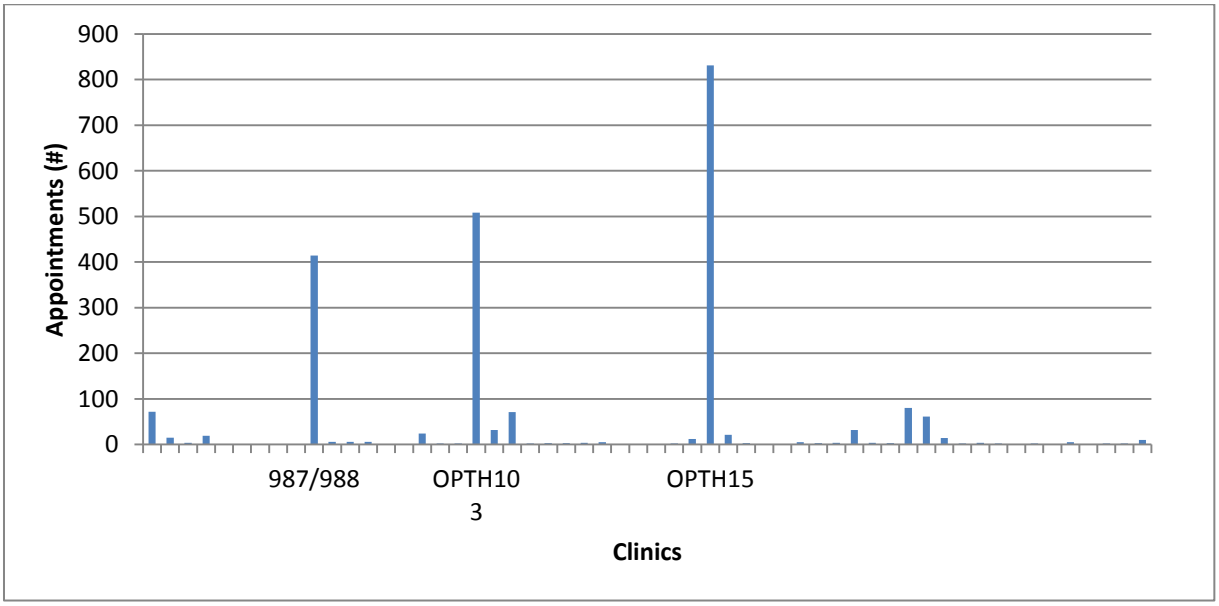

Figure 1.1 Clinic Attendance

1.4.2 Range

The available data start with appointments dates in 2011 and end with appointment dates from the beginning of October 2015. This allows looking for seasonality and trends in the data.

1.4.3 New Arrivals

When new patients get referred they are placed on a waiting list before getting assigned an appointment. From most patients the available data only contains the date of the first appointment of a patient. So there is no information available about when exactly they were referred. In order to determine waiting times and actual demand for appointments this is of critical importance.

Because of the governmental targets on waiting times of new patients, it can be assumed that the new patients have been prioritized over follow ups. Therefore, it can be assumed that the amount of new patients in a certain period corresponds with the amount of new patients slotted in in that same period. Assuming the data about new arrivals provides good information, the data can be used to forecast future demand. In this project the time series will be analysed and different forecasting methods will be tested.

1.4.4 Follow Ups

A similar problem as for new arrivals arises for follow ups. There is lots of data available about the appointment dates for follow ups, but even with target dates, there is still very little information available about when exactly the appointment should have taken place to be a timely follow up. In other words, there is no way to determine how much a follow up has been delayed or what exactly the demand for follow ups was at a certain point in time.

A generated stream of demand is necessary to find the optimal (use of) capacity. Therefore, a simulation was used, recreating the situation to be able to register the demand. This simulation makes use of both the (forecasted) new arrivals and the looping through follow up structure of returning patients to establish a demand pattern.

2. Literature

The very broad topic of this study is outpatient scheduling. A lot of papers could be found on this subject. Therefore, it was started with the focus on Ophthalmology. From there the search was broadened to relevant references. The resulting papers and their relation to this project were discussed, divided into three paragraphs.
In this project it was meant to do two main steps. First the actual demand on the outpatient system needs to be determined. Secondly, the capacity and demand will be compared to obtain as good as possible scheduling. Therefore this chapter was also split into those two paragraphs. Literature about forecasting and/or determining demand will be discussed in the first paragraph. In the second paragraph more general literature about outpatient scheduling will be discussed. In the third paragraph it is explained how this project is a valuable addition to the discussed literature.

2.1 Determining Demand

To be able to do good scheduling it is important to have an idea of future demand. In this project the first aim was to determine actual demand on the clinic. Because there were some data limitations, the exact demand in the current situation was not known. This led to a search for solutions on how to determine this. One of the ideas involved using a ratio of new patients and follow up patients. Such ratios for ophthalmic patients were discussed by Pan et al. (1). Both ratios mentioned, 3:7 and 5:8, were compared to the historic data of this project, but were too far off to use.

Another idea was to forecast by means of regression. This means certain influencing factors could be taken into account. Dechartres et al. (2) found that the level of complexity of consultations is correlated mainly with four factors. The type of referral, the consultation duration, the number of consultations in the previous year and the number of diagnostic tests performed. These factors were determined focusing on doctors' workload and the length of an actual consultation. This is different from what was focused on in this project. In this project we are only interested in the planning of the consultations, i.e. appointments, assuming each appointment takes exactly one time slot. It might be that the same factors have some influence on the follow up structure however. Therefore these factors were analysed in relation to the follow up structure. The results were not sufficient to use regression. It was decided to use plain forecasting methods on the registered activity.
During the research on literature, nothing was found on trying to determining demand, assuming historic data does not provide an appropriate picture. This part of this project will therefore be a valuable addition to the reviewed literature.

2.2 Scheduling Outpatient Systems

To be able to make a good appointment schedule, one first needs to determine what exactly should be optimized. Most appointment schedules use minimizing waiting times as the objective function. Waiting time is a wide understanding though. It consists of patient waiting time, split in both direct and indirect waiting, and doctor's idle time. Direct patient waiting time is understood to be the time between the moment the patient arrives and when the patient is served. Indirect patient waiting time is the time between the moment the patient requests an appointment and when the appointment takes place.
On top of that, according to Gupta and Denton (3), we can take in account the fairness of the distribution of the waiting times. That could be by either setting a maximum to the waiting time, or minimizing the variance or just by purely looking at efficiency. Unfortunately, in this project there was not any time left to try all three approaches, but it will be discussed in the recommendations.

Tugba and Veral (4) wrote an interesting review of literature about outpatient scheduling in health care. They conclude that you have to deal with the following list of complications:

- Punctuality of the patients
- Number of doctors
- Number of appointments per clinic session
- No-shows
- Emergency walk-ins

In this project we are not interested in either emergency walk-ins or punctuality of patients. The other aspects can be taken into account. In chapter 5 it will be looked at how best to divide appointment slots over a week with how many clinics working at the same time. This means different numbers of doctors and numbers of appointments per clinic session will be tested. About no-shows Tugba and Veral (4) wrote that it is best handled by shortening appointment intervals. It is especially mentioned that overbooking is not a good method. Since it can be concluded from the data, combining the number of appointments scheduled and the known capacity, overbooking has been used in the past in the clinics that was focused on for this project. By creating a more efficient planning tool it was aimed that this will not be necessary anymore.

By Pan et al. (1) it was investigated how best to improve scheduling. They used a discrete event simulation to compare new results with the old situation. It was concluded that a wider distribution of slots during the week and rearrangement of new patient slots and follow up slots are good ways to do this. In this project a simulation will also be used to test how best to distribute slots weekly. An optimization will be used to look at how slots can best be divided into new patient slots and follow up patient slots.

2.3 Related Literature

De Vuyst et al. (5) used an analytical approach to evaluate appointment schedules in health care. In most reviewed literature a simulation was used however. In this project it was also chosen to use mostly simulation.

It was remarkable that a simulation was used in many of the reviewed literature. For example Harper and Gamlin (6) wrote about reducing outpatient waiting times by means of improving appointment scheduling with a simulation modelling approach. Only this was focusing on direct waiting times, instead of indirect waiting times as we want in this project.

Su and Shih (7) also used simulation to manage an appointment system in outpatient clinics. They were focusing on a mixed registration-type appointment system however. This means they were concerned with both scheduled patients and walk-ins. Similar to what will be done in this project, it also comes down to finding an optimal allocation of slots. In this project all appointments are scheduled however and again we bump into the difference between optimizing direct and indirect waiting times.

The allocation of slots, what was aimed to do for new patients slots and follow up patient slots, was found in a paper by Zonderland et al. (8). They focused on the planning and scheduling semi-urgent surgeries. This means they had to deal with a very unpredictable demand. In this project optimizing allocation of slots was based on a predictable demand.
A part of an efficient appointment system is adapting well to non-attendances. Potamitis et al. (9) wrote about how best to reduce non-attendances. In this project non-attendances are accounted for,

but there was not enough time to analyse what effects reducing them would have. This would be an interesting direction for further extension of the model in this project.

According to Braaksma et al. (10) there is a lot to be gained by using online appointment scheduling in health care. To be able to do this, the study of this project about dividing slots into new patient slots and follow up patient slots in advance might be useful.

The focus and methodology of this project, optimizing indirect waiting times for patient using a combination of an optimization and a simulation, is a valuable addition to the reviewed literature in this chapter.

3. Methods

3.1 Forecasting Methods

In forecasting there are two types of basic methods to distinguish: averaging methods and exponential smoothing methods. Averaging methods use *equal* weights for all observations used to forecast, whereas exponential smoothing methods use *unequa*l weights. Unequal weights can give more weight to more recent observations. In this project the averaging methods used is the moving average. The exponential smoothing methods used are adaptive-response-rate single exponential smoothing (ARRSES) and Holt's Linear Exponential Smoothing.
Another forecasting method is Survival Analysis. In this project there was concluded that it could be a helpful method, but it was not used in the end.

3.1.1 Moving Average

The principal of a moving average is that a new average is calculated every time a new observation becomes available. A fixed number (k) of latest observations is used every time a new forecast is calculated, the newest observation replacing the oldest one. With F_t and Y_t representing the forecasting value and the observed value at time *t* respectively, the moving average forecast of order k is given by:

$$F_{t+1} = \frac{1}{k} \sum_{i=t-k+1}^{t} Y_i \tag{1}$$

This method is not appropriate when the observed data exhibits any trend or seasonality. For example, if a time series contains a linearly increasing trend line, it is not possible to use the above mentioned formula. A value higher than the already observed values is bound to come but it is not possible to forecast this using only the average of previously observed values.
The same holds for seasonality. The more previously observed values are taken into account, the smoother the forecasting graph becomes. This means the peaks or drops in values due to seasonality will not be taken into account.

The formula mentioned earlier represents the simplest way to use a moving average, by using equal methods for a consecutive number of previous observations. There exist however more interesting ways of using moving average. To be able to adapt the method better to our situation, the formula was changed a little.
The observed values used to calculate the forecasting values can instead of consecutive be chosen with gaps. For example the forecasting value can be calculated by means of the last observed value, the 3rd to last observed value and the observed value ten time units previous. Also, the weights can differ for each previously observed value. Now a new formula can be set up:

$$F_{t+1} = \frac{1}{k} \sum_{i=t-k+1}^{t} w_i * Y_i \tag{2}$$

In this formula the forecasting value and observed value at time t are again given by F_t and Y_t respectively and the order of the moving average forecast by k. The weights w_t can be different for every position of the previous observations. For example the last observed value can be given a higher weight, because it is probably more significant. In case of seasonality, the last observed value exactly a year ago can for example be given a higher weight.

In other words, the weights can be adjusted to a specific time series. In fact, the weights can even be optimized for a specific time series. Obviously, the objective here will be to minimize the difference between forecasted values and observed values of the same moment in time. To be able to do this, a sensible number of previous time units that will be taken into account have to be chosen. Then the combination of weights which gives the smallest errors can be found.

An advantage of this method is that it is very adaptable to different time series, since it can be chosen to optimize a lot of weights. The downside of too many weights however is that all that weights will be small and a small change in the data can have large impacts on the forecasting errors. In other words, the method can lack in robustness.

3.1.2 Adaptive-Response-Rate Single Exponential Smoothing

The ARRSES forecasts by adding weighted values of both the last forecasted value and the last observed value. The special aspect of this method is the fact that the weight (α) can be changed when changes in the pattern of data occur. The forecasting equation is given by:

$$F_{t+1} = (1 - \alpha_t)F_t + \alpha_t Y_t \tag{3}$$

The weight (α) changes by means of changes in the values A_t and M_t. Those values are determined as follows:

$$A_t = \beta E_t + (1 - \beta)A_{t-1} \tag{4}$$

$$M_t = \beta|E_t| + (1 - \beta)M_{t-1} \tag{5}$$

With:

$$E_t = Y_t - F_t \tag{6}$$

In these formulas E_t represents the forecasting error at time t. The value β is, in contrast to α, a fixed parameter between 0 and 1. There can now be derived that A_t denotes a smoothed estimate of the forecasting error and M_t a smoothed estimate of the absolute forecasting error. The (adapted) weight α can be calculated as follows:

$$\alpha_{t+1} = \left|\frac{A_t}{M_t}\right| \tag{7}$$

This method only uses the last observed value to forecast with and by using the last forecasted value, also the second to last observed value a little. Therefore no trend or seasonality can be taken into account so this method works well when the data is non-seasonal and shows no trend.

Similar to the moving average, the ARRSES method can be adjusted to a specific time series. The parameter α changes by itself, but, as mentioned before, the parameter β is fixed. This last parameter can therefore be optimized according to the time series to which the ARRSES method will be applied. Optimal clearly means the parameter with which forecasting results in forecasting values with the smallest error compared to observed values.

An advantage of the ARRSES method is that it allows for changes in the data. In this project this can be convenient in case the actual demand turns out to be different from the registered activity in some time periods. A disadvantage is that this method cannot nearly as well be adapted to a specific time series, since there is only one parameter to optimize which influence is limited.

3.1.3 Holt's Linear Exponential Smoothing

In contrast to the other two discussed forecasting methods, Holt's Linear Exponential Smoothing is able to take a possible trend into account. To achieve this, L_t and b_t, the level of the series and the estimate of the slope of the series at time *t* respectively, are introduced:

$$L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + b_{t-1}) \tag{8}$$

$$b_t = \beta(L_t - L_{t-1}) + (1 - \beta)b_{t-1} \tag{9}$$

In these formulas both α and β are fixed smoothing parameters between 0 and 1. The actual forecasting value F_{t+m} for time t+m, forecasting m periods ahead is calculated as follows:

$$F_{t+m} = L_t + b_t m \tag{10}$$

The initial values for L_t and b_t have to be estimated. For L_1 the first observed value in the data can be used and for b_1 the difference between the first two values.

This method is applicable when the time series does not contain any seasonality. Because the method only uses the last observed value to calculate the forecasting value it is not possible to take seasonality into account. As mentioned earlier, this method can take trend into account.

Similar to the other methods this method can be adapted to a specific time series. In this case the fixed parameters α and β can be optimized, minimizing the error over a number of forecasts.

The advantage of Holt's Linear Exponential Smoothing over the other two mentioned methods is the fact that with this method it is possible to forecast more than one time period ahead. Of course it quickly becomes less accurate if the amount of periods to forecast increases. But for forecasting only one time period ahead this method is fairly robust. This means that a small change in the data will not provide large errors in forecasting.

3.1.4 Comparing Methods

There exist a lot of forecasting methods, but which one works best depends on the data series. To be able to compare different methods, they all have to be applied to the same data series. But even then there are different ways to compare. In this project the mean squared error (MSE) and the mean absolute percentage error (MAPE) were used. The MSE can be calculated as follows:

$$MSE = \frac{\sum_{t=1}^{n}(F_t - Y_t)^2}{n} \tag{11}$$

In this formula *n* represents the number of forecasted values. By squaring the difference between forecasted and observed values, the MSE gives exponentially more weight to large error values. This method is useful if occasional large errors are to be prevented as much as possible.
A disadvantage of the MSE is that it is hard to interpret the obtained value. If you however compare it to other MSE results it can be very useful. This method of error calculation was in this project used for the determination of the error in optimizing the forecasting parameters.

The MAPE is described by the following formula:

$$MAPE = \frac{\sum_{t=1}^{n} \left| \frac{Y_t - F_t}{Y_t} \right| * 100}{n} \quad (12)$$

Again *n* represents the number of forecasted values. The MAPE calculates the error subject to the magnitude of the value that was observed. An advantage of this method is that the outcome can easily be interpreted. It provides the average percentage that the forecasting errors are of the observed values.

To compare the different methods, all methods can be optimized to the available time series. The different errors for the same observed values can be calculated (either MSE or MAPE) and the results can be compared. However, the optimized parameters are now suitable for the complete time series. But in reality the parameters will not be optimized every time a new observed value is obtained. Therefore it is also useful to take about three quarters of the time series to optimize the parameters with and then forecast the last quarter. Comparing the errors obtained in this way will give a better impression of which methods work well on the specific time series.

3.2 Queuing Theory

To set up the queuing model used in this project, two rules for Poison arrivals were used. The first rule is that two different Poisson arrival processes can be added as follows:

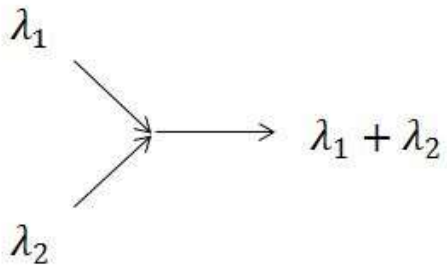

Figure 3.1a: adding arrivals

The second rule states that when splitting a Poisson arrival, the new streams can be treated as new Poisson arrivals with new rates. Visually this comes down to the following:

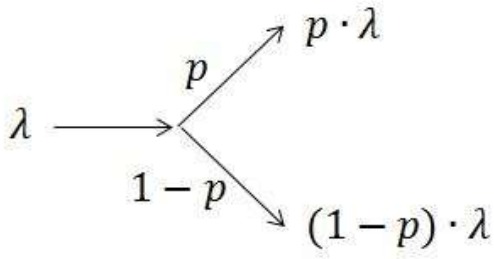

Figure 3.1b: splitting arrivals

For this project a network of queues will be set up. In this set up (a part of) the output from one queue will form the input of another queue. Therefore more information is needed about the departure process of a queue.

We will be dealing with an open queuing network. The most important property of this system is quasi-reversibility. The Markov process associated with a queuing system is said to be quasi reversible if the state of the process (for all classes of patients) at time t is independent of the arrival process after time t and independent of the departure process prior to time t. (12) If a quasi-reversible queue has class-dependent Poisson arrival processes, then the departure process of class c customers is also Poison in steady state.
For example, a system like the one shown below consists of quasi-reversible queues. This example was discussed by Bunday (11).

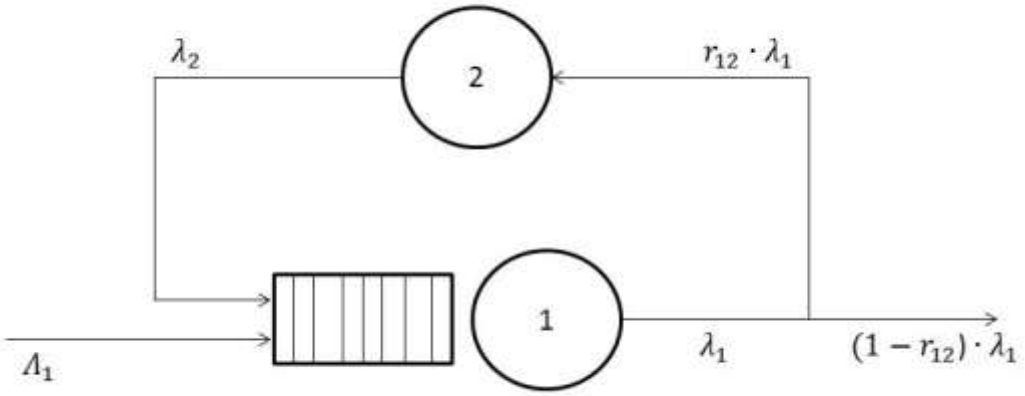

Figure 3.2: Queuing Model

With:
λ_i = effective arrival rate at node i
Λ_i = external arrival rate at node i
r_{ij} = part of customers moving from node i to node j

For a system as in the example above a product form solution is valid. This means the traffic equations can now be formulated as follows:

$$\lambda_1 = \Lambda_1 + \lambda_2 \tag{13}$$

$$\lambda_2 = r_{12} \cdot \lambda_1 \tag{14}$$

Solving the traffic equations yields:

$$\lambda_1 = \frac{1}{1 - r_{12}} \Lambda_1 \tag{15}$$

$$\lambda_2 = \frac{r_{12}}{1 - r_{12}} \Lambda_1 \tag{16}$$

According to Jackson's theorem the two nodes can now be treated separately, with arriving rate λ_i and service rate μ_i. In this project node 1 will be an M/M/1 queue and node 2 an M/M/∞ queue. This yields the following performance measures for node 1 and node 2.

Node 1

$$E[\#\ in\ system] = \frac{\lambda_1}{\mu_1 - \lambda_1} \tag{17}$$

$$E[\#\ in\ queue] = \frac{\lambda_1}{\mu_1 - \lambda_1} - \frac{\lambda_1}{\mu_1} \tag{18}$$

$$E[time\ in\ system] = \frac{1}{\mu_1 - \lambda_1} \tag{19}$$

$$E[time\ in\ queue] = \frac{\lambda_1}{\mu_1(\mu_1 - \lambda_1)} \tag{20}$$

Node 2

$$E[\#\ in\ system] = \frac{\lambda_2}{\mu_2} \tag{21}$$

$$E[\#\ in\ queue] = 0 \tag{22}$$

$$E[time\ in\ system] = \frac{1}{\mu_2} \tag{23}$$

$$E[time\ in\ queue] = 0 \tag{24}$$

3.3 Simulation

In this project an existing simulation shell, developed by Geraint Palmer with Python, was used. ASQ Simulates Queues is a simulation that can be used to simulate a queuing network of any architecture with multiple classes of customers and a variety of service distributions. To be able to use this simulation shell for this project a list of changes needed to be made. Therefore, it is important to obtain a good understanding of the ASQ Simulates Queues. In this paragraph the setup of the existing simulation code and the event structure will be explained.

3.3.1 Set up Simulation Code

The main object of the framework is called “Simulation”. This contains information about the queuing network itself, global variables, and the methods that run the simulation and write data to a file. Three different pieces of code feed into the “Simulation” code, these three parts are called “Exit Node”, “Arrival Node” and just “Node”. As can be concluded from the names, “Exit Node” deals with the customers that leave the system. This means “Exit Node” functions as a dummy node to send and store information about the individuals that leave the system. “Arrival Node” does the opposite. This creates all newly incoming customers. The individuals never stay here, but are immediately transferred to the relevant node in “Node”. The remaining piece of code “Node” deals with the queue for service, service itself and moving customers amongst nodes. This object also contains methods to give “Individuals” information like arrival date, service start time, service end time and waiting time. It also writes the “Data Records” of “Individuals”. To make the right adjustment to the simulation for this program, “Servers” was added as an object. “Servers” contains methods to change shifts and therefore add and delete servers of relevant nodes.

Every time a customer finishes service a line of the following information is stored:

Table 3.1: Stored Information of a Customer Who Finished Service

Patient ID	Class	Node Number	Arrival Time	Waiting Time	Service Start Time	Service Time	Service End Time	Exit Time

A visual overview of the code of the simulation, consisting of seven different parts, can be viewed below.

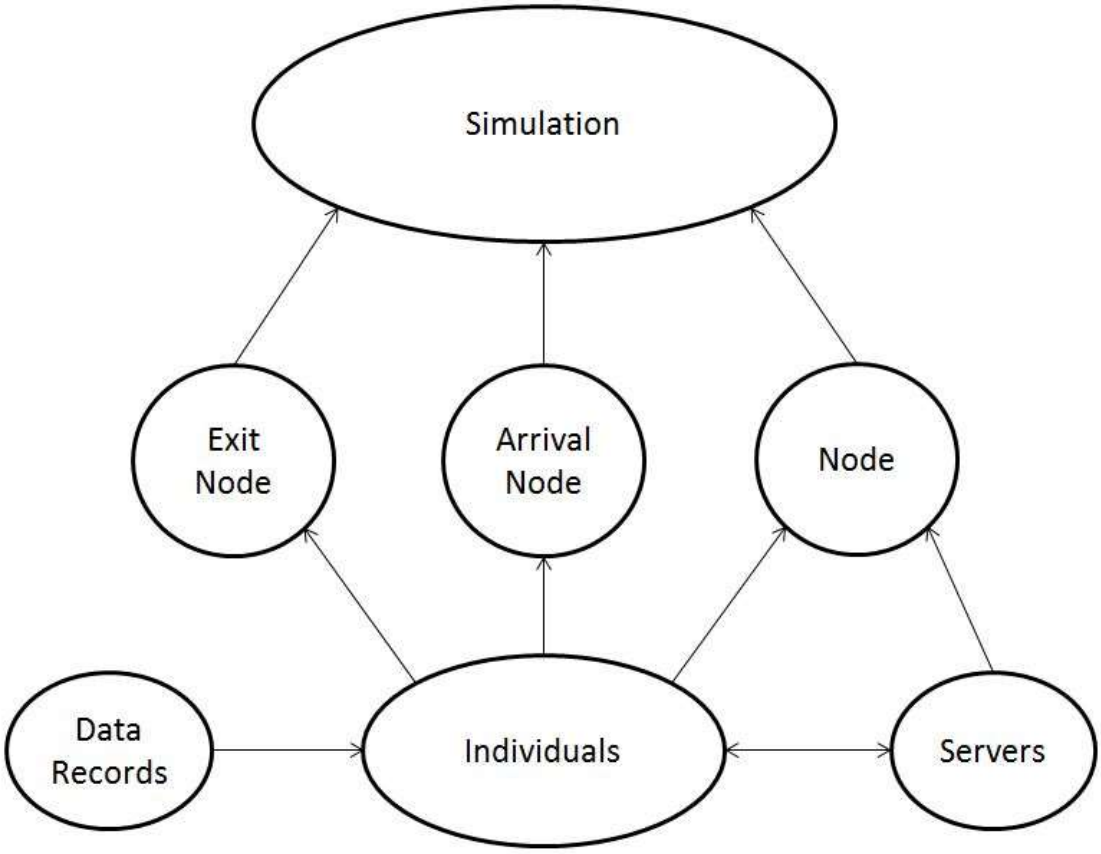

Figure 3.3: Overview Different Parts of Code

3.3.2 Event Structure

For the simulation a so called three-phase-simulation-approach, described by Stewart Robinson (13), was used. This is shown in figure 3.4.
In Phase A the simulation clock is moved to the time of the next event. There are two different types of events to distinguish. Only the B events are scheduled at a certain point in time. The C events are conditional events and just occur as a result of another event. Naturally the B phase carries out the B events and the C phase the C events. C events can be triggered by other C events. Therefore the extra check needs to be carried out whether all the C events have been taken care of. This process repeats itself until the end of the simulation time is reached.

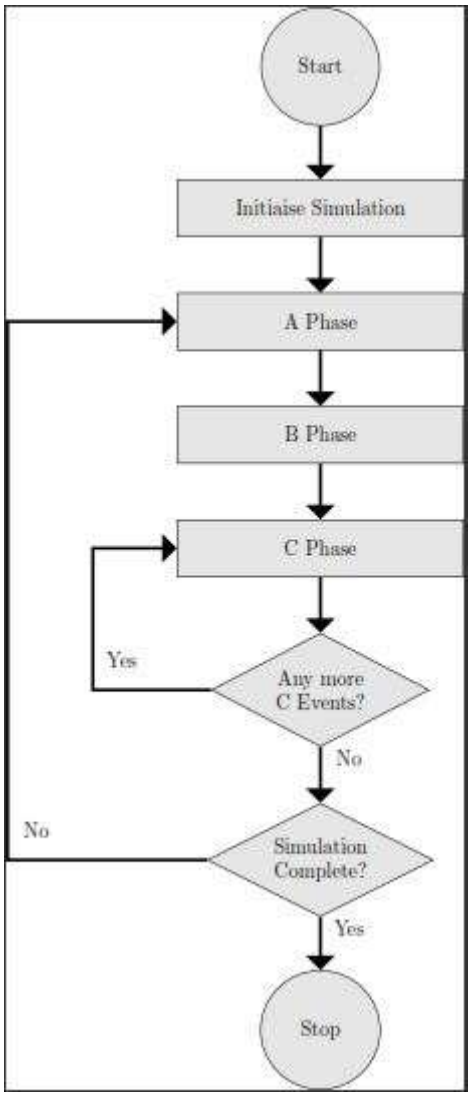

Figure 3.4: Three-phase-simulation-approach

In this simulation there were two B events; an external arrival of a customer and a customer ending service. The three B events (given in circles) with their following C events (given in squares) can be viewed in the figures 5 and 6 below.

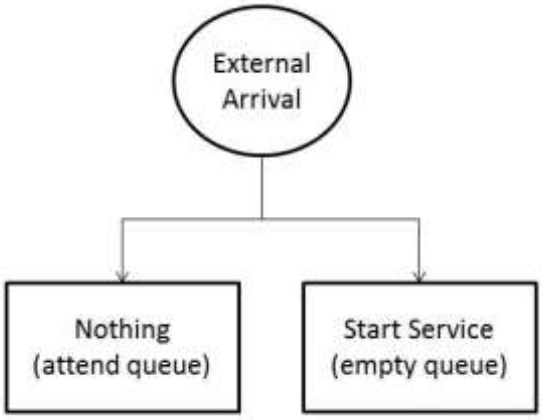

Figure 3.5: B event 1

After a new customer enters the system at a node, there can either be a queue or not. If there is a queue, the customer just attends the queue and nothing needs to be done. Of course if there is no queue the customer can immediately enter service. This means the C event “Start Service”, which comes down to generating the time this customer’s service will be finished, needs to be carried out.

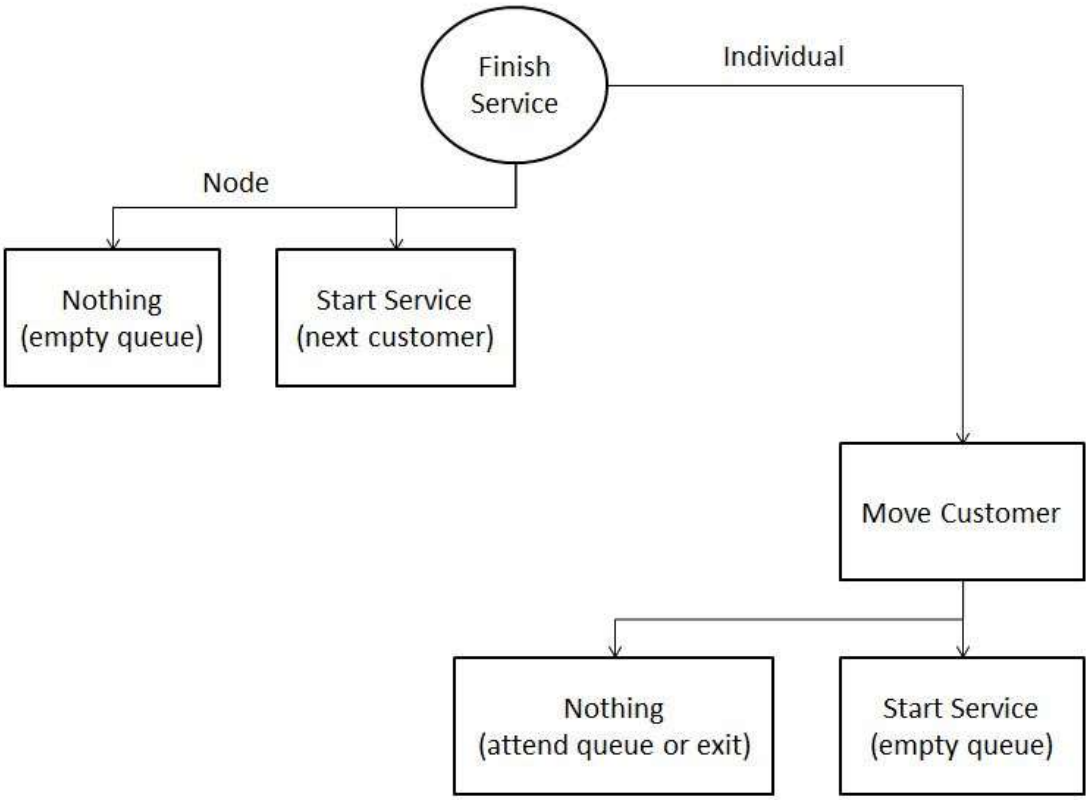

Figure 3.6: B event 2

After a customer finishes service, there need to be made changes to both the relevant node and the individual. To the node there are again two possibilities of the following situation. The queue for the node can be empty in which case no C event will occur. If the queue is not empty the next customer will start service. In this C event the time for the B event, when this customer finishes service, will be generated.

When an individual finishes service and leaves the node, the customer needs to be moved. In this C event it is determined to which node the customer will be transferred. The customer can either go to a new node, return to the same node where service was just finished or ‘exit’. If the customer exits the system no more C events occur. When the customer is transferred to a node, either a new one or the previous one, the customer can find an unoccupied server there or not. The customer can then start service or attend the queue. As mentioned before, in the first case the following C event will determine the moment in time that the customer will finish service. In the second case no other events occur.

4. Demand Modelling

To develop an optimal scheduling strategy, the demand for appointments and the available capacity have to be compared. In this section the aim is to determine a stream of demand. The demand was examined in two main steps. Firstly, it was looked at forecasting methods to apply on historic data. Secondly, a simulation was carried out, reproducing the situation, to be able to store the relevant data for this project.

4.1 Problem Definition

To be able to construct a smart scheduling strategy, there are two things that need to be established. Firstly the number of patients requesting an appointment per time unit is needed, in other words; the demand on the system. Secondly the number of available slots, the capacity, is needed. In this section it was focused on the first part, the demand.
In contrast to capacity, the demand is not given for the future. That is why this section is started with looking at different forecasting methods. It was intended to use historic data to get a grip on future demand by means of plain forecasting methods. In doing so a problem was discovered. The available dataset provides information about when appointments are booked in the system. The problem is that the demand is bigger than the current capacity. Therefore the appointment dates do not provide applicable information about the actual demand.
The initial choice for the specific clinic OPHT103 was made because of the relatively good target data. The target data does give an impression about when the follow ups should have taken place, though there is still very little data available compared to the whole set. Using this target data there still can be created a better understanding of historic demand. Unfortunately, it was not sufficient to rely on completely.
For this reason the system was also analysed by means of a Queuing Model, which gives an idea of the busyness of the system. To get a clear picture of the actual demand, by recreating the situation of the system, a Simulation was carried out. As a last part of the problem, input data needs to be gathered for this simulation. In the part of recreating the process of new patients coming in, the forecasting methods come in handy again. As mentioned before, it can be relied upon that recently new arrivals have been prioritized over follow up patients because of the governmental targets. This means forecasting methods applied on the historic data can provide useful information about just the new arrivals.

4.2 Forecasting Methods

To be able to forecast a stream of demand, a historic series of demand is required. As already mentioned in the Problem Definition, determining real demand is hard because the available data about demand is constantly limited by capacity. The question is how to use the historic data that is available. In this project two different methods of using historic data were used.
Firstly, the column appointment dates from the original dataset can be used, completely ignoring whether it is a new patient or a follow up patient. It can be assumed that the real demand was actually higher than the listed demand. After forecasting, the complete forecasted graph can be lifted a little to represent the capacity limitations.
Secondly, there can be determined two separate historic series of data. Again the appointment dates can be used but now they can form one series for only new patients and one series for follow ups. Because of the target dates placed on the new patients, it can be assumed that all new patients have been seen immediately. This means the forecasted series for new patients would be correct and only the forecasted values for the follow up series should be increased.

Either way, a historic series of data needs to be forecasted. There exist a lot of different forecasting methods, only a few of them were discussed in the section "Methods". To determine which ones would work best, the historic data needs to be examined.

4.2.1 Trend and Seasonality

In order to narrow down the options of forecasting methods, there was searched for trend and seasonality in the data. The monthly total of registered appointments is given in the graph below. The black line represents the linear trend.

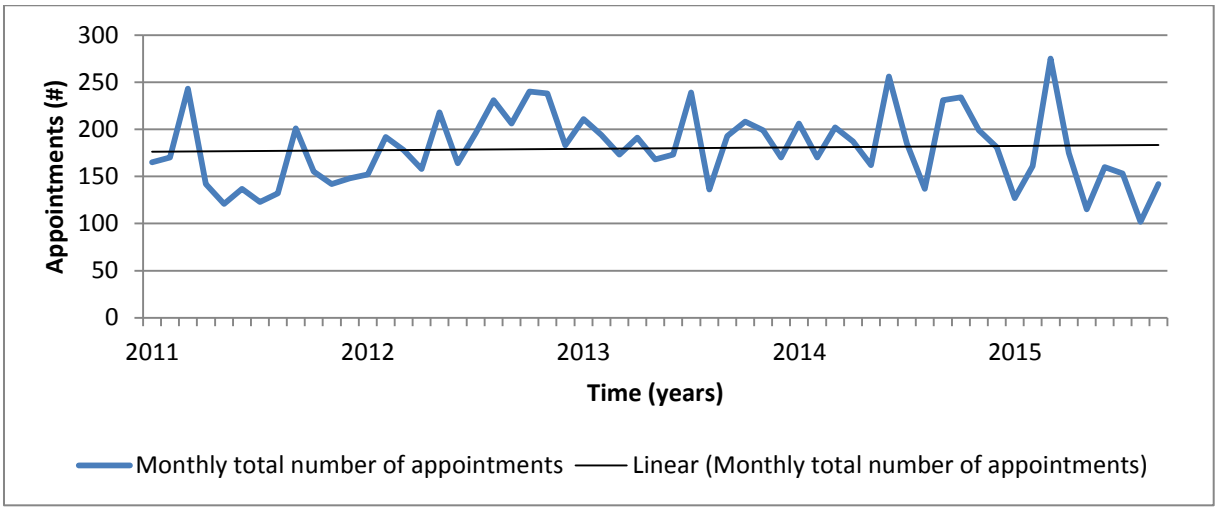

Figure 4.1: Trend, monthly total number of registered appointments

From this graph can be concluded that there has been no consistent increase or decrease in the trend over the past 5 years. In the graph below the monthly numbers of new patients and follow ups can be viewed separately. The black lines are linear trend lines.

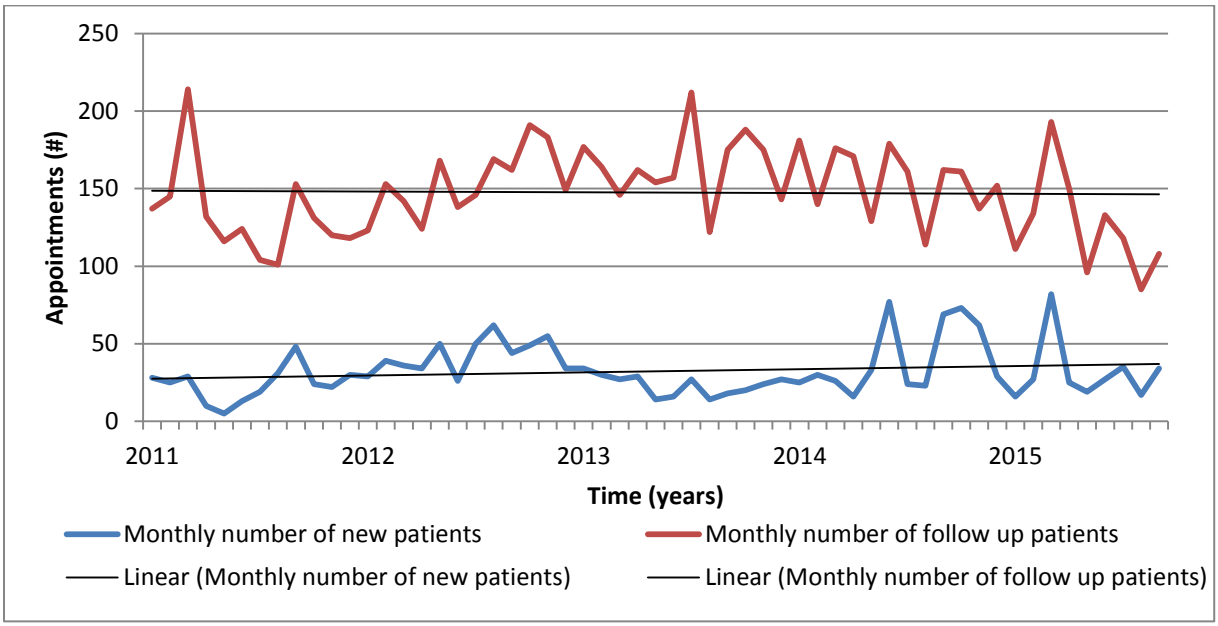

Figure 4.2: Trend, new patients and follow up patients separately

The same conclusion can be drawn as from the first graph. There has been no consistent change in the series over the last 5 years. Therefore, there is no apparent trend in the historic series. Only in the new arrivals there seems to be a slightly increasing trend, but it was suspected that this is a result of the current planning methods. Especially the graph for new patients oscillates a lot near the end. This could be a result of overbooking to shorten the waiting lists for first appointments of new patients.
In this project it can thus be focused on forecasting methods that work well without apparent trends in the data. Still it was considered useful to also test a (few) forecasting method(s) that can take some trend into account.

To search for seasonality the historic data was split into yearly graphs. Below the total registered activity of the last 5 years can be viewed.

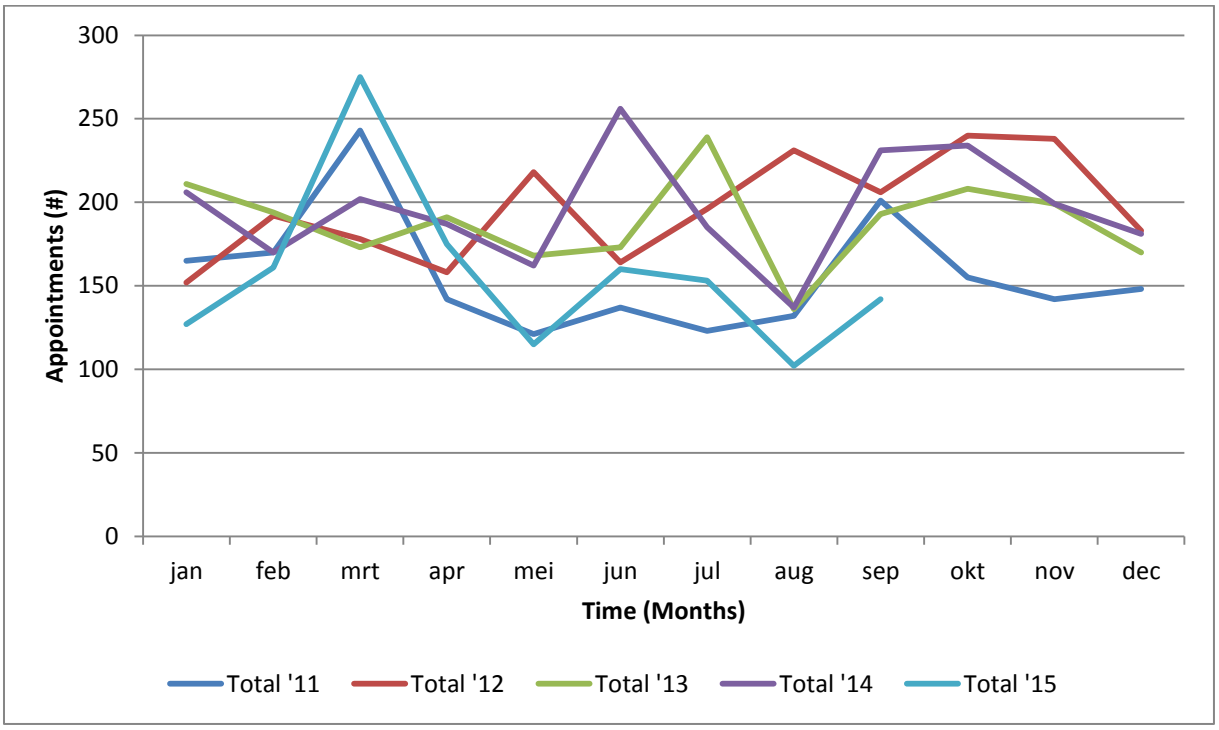

Figure 4.3: Yearly total registered activity

In this graph there does not seem to be any particular behaviour depending on the season. To make sure the situation is not different when dividing new patients and follows up patients, the graph below was created as well.

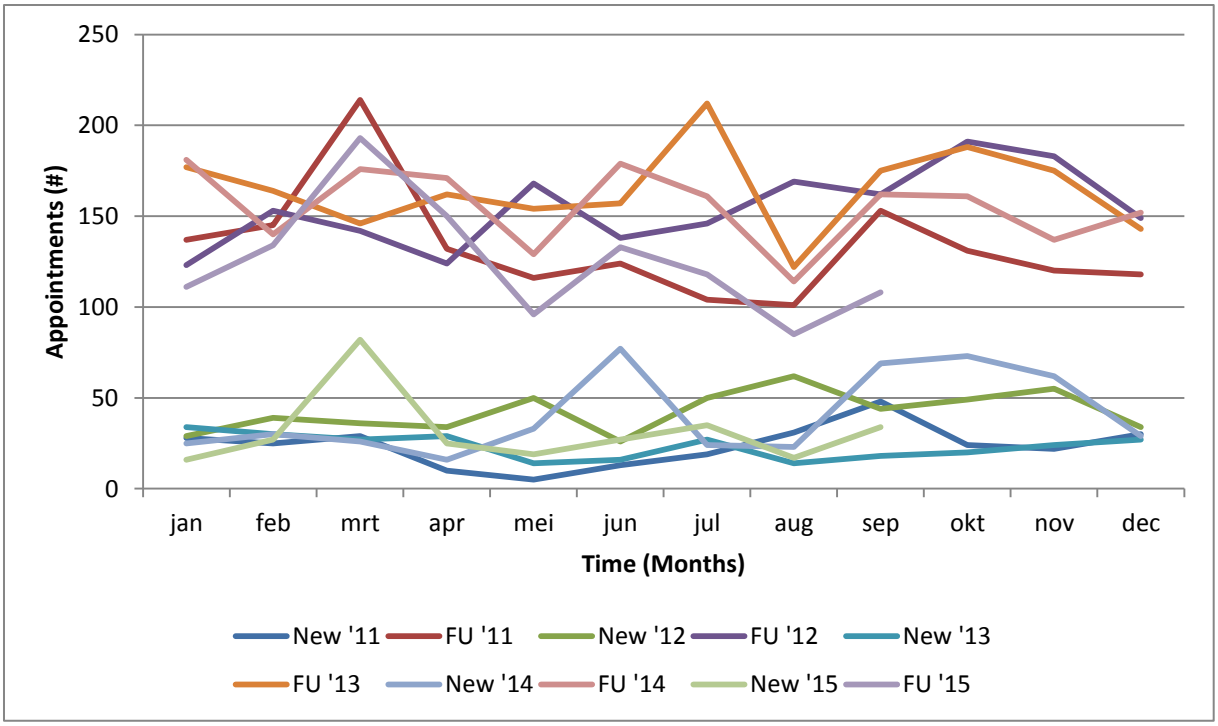

Figure 4.4: Yearly registered activity for new patients and follow up patients

Still there seems no seasonality to be discovered. However, because seasonality is harder to spot than a trend in just a graph, seasonality was also tested by means of the autocorrelation factor. After plotting a time series, this method can help to describe the relationship between various parts of the time series which are a certain time distance apart. The autocorrelation factor consists of autocorrelation coefficients of different order. A coefficient represents the correlation between two moments in time, within the time series, with the number of the order in time units in between. The formula that describes the autocorrelation coefficient with order k is as follows:

$$r_k = \frac{\sum_{t=k+1}^{n}(Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^{n}(Y_t - \bar{Y})^2} \tag{25}$$

With of course Y_t and $\bar{Y}$ representing the observed value at time t and the mean observed value respectively.
In this project, the autocorrelation coefficients until order 12 of the time series were calculated. The number 12 was chosen to make sure to catch possible yearly seasonality, but to make the number much larger would not make sense because there were only a few years of data available. A plot of all the autocorrelation coefficients together forms the autocorrelation factor:

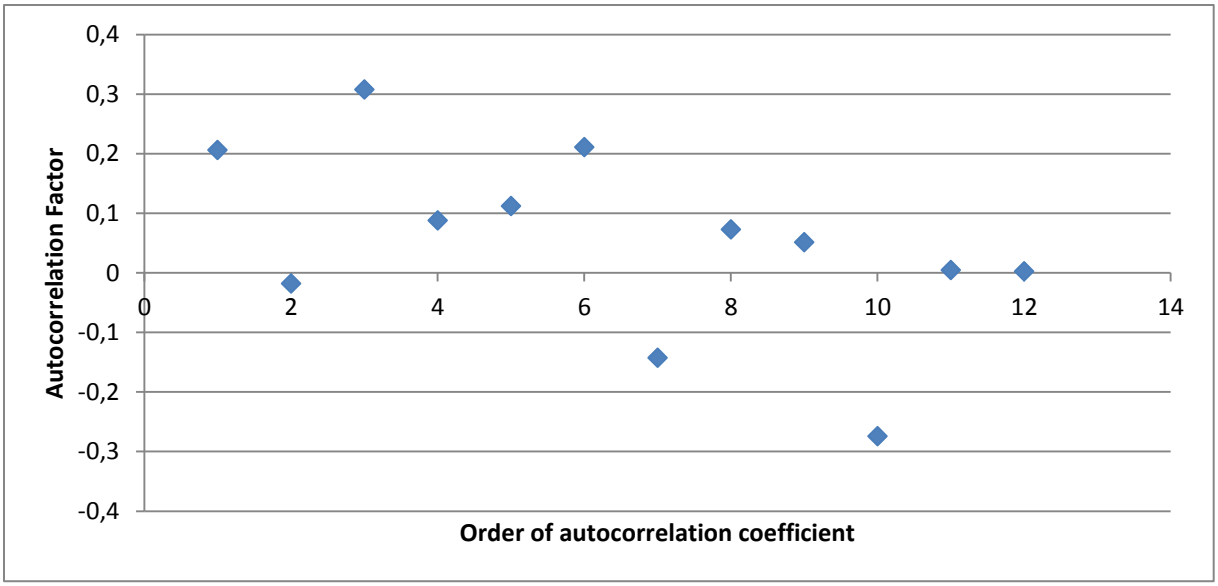

Figure 4.5: The Autocorrelation Factor

Because all the coefficients, with a very occasional exception, stay near the x-axis, it can be concluded that there was no apparent seasonality. Therefore, in choosing forecasting methods there can be focused on methods that work well without any seasonality in the data series.

4.2.2 Applying different methods

After analysing the available time series, three appropriate forecasting methods were selected. In this paragraph those methods and their results will be discussed. The results were analysed by different error measures and compared.

4.2.2.1 Moving Average

As discussed in the section "Methods", the particular form of moving average used in this project first needs a maximum number of previously observed values to take into account. To allow for some seasonality to appear it was started with taking the last 12 months into account. Then the weights were optimized. However, for the time periods more than 6 months the weights turned out to be approximately zero. Therefore the number of previously observed values to take into account was narrowed down to only 6 months. Optimized over the complete time series this resulted in the following weights:

Table 4.1: Optimized Weights Moving Average

1 month previous	0.23
2 months previous	0.0
3 months previous	0.43
4 months previous	0.0
5 months previous	0.19
6 months previous	0.15

Though it was concluded that there was no apparent seasonality, this results suggest at least that the demand three months ago has a significant impact on the current demand. This does not have to be seasonality, but it is a definite relationship within the time series which might be used later.

The associated MSE (explained in the section “Methods”) in this optimized situation comes down to a value of 1190.

4.2.2.2 ARRSES

For this method only one parameter, β, can be optimized over the complete time series. Also the starting value of the parameter α, which represents the weight of the previous observed value relative to the previous forecasted value, can be optimized. It is part of the forecasting method ARRSES that the parameter α changes over time by itself. The results of the optimized parameters were as follows:

α	0.17809
β	0.185597

This yielded for the complete time series an MSE with a value of 3877.

4.2.2.3 Holt’s Linear Exponential Smoothing

This method has the clear advantage in contrast to the other two methods that it can forecast more than one time period ahead. To be able to compare it to the other method however it was here only used with forecasting one time period ahead. Two parameters can be optimized, both α and β. Optimizing over the complete time series these parameters adopted the following values:

α	0.2134
β	0.0.081959

This yielded an MSE for the complete time series with a value of 1550.

4.2.3 Comparing Methods

In this paragraph the MSE and MAPE values, discussed in the section “Methods”, of all the methods will be compared for optimizing over the complete time series. After this another test will be carried out. This includes optimizing over only a part of the time series and using the obtained parameters to forecast over the part of the time series that was not used to optimize over.

4.2.3.1 MSE and MAPE

In this project the values were optimized with the objective function of minimizing over the MSE error measure, because this is the most common method. The MAPE was however calculated as well. The results were gathered in a table:

Table 4.2: MSE and MAPE Values for All Methods

	MSE	MAPE
Moving Average	1190	16.35
ARRSES	3877	26.5
Holt’s Linear Exponential Smoothing	1550	19.54

It can be concluded from both error measures that moving average performs best. However, Holt's Linear Exponential Smoothing is not far behind. The second method, ARRSES, performs very badly compared to the other two methods, due to the fact that it smooths too much and can very poorly be adapted to a specific time series. In fact ARRSES performs so much worse than the other two methods that it was not considered any further in this project.

4.2.3.2 Forecasting Further

Until now all weights and parameters were optimized over the complete time series, which means all the available data from 2011 until 2015. When a forecasting method is chosen however, the parameters will not be changed for every forecast. This means the parameters will never be optimized for the complete past time series. Therefore the methods Moving Average and Holt's Linear Exponential Smoothing were also tested differently. For both methods the weights and parameters were again optimized, but now using only the data until the end of 2014.
For Moving average the following weights were adopted:

Table 4.3: Optimized weights Moving Average 2

1 month previous	0.13
2 months previous	0.05
3 months previous	0.46
4 months previous	0.13
5 months previous	0.02
6 months previous	0.02
7 months previous	0.19

The weights have changed considerably compared to the old obtained numbers. Even an extra previous month needs to be taken into account. For Holt's Linear Exponential Smoothing the newly obtained parameters are as follows:

α	0.235818
β	0.0.061111

It is remarkable that the parameters barely changed for this method. It is clearly much more robust than Moving Average. This also shows in the results in MSE and MAPE calculated over 2015, which will adopt the following values:

Table 4.4: MSE and MAPE Values over 2015

	MSE	MAPE
Moving Average	3682	37.75
Holt's Linear Exponential Smoothing	3031	30.93

Clearly, it can now be concluded that Holt's Linear Exponential Smoothing performs best on the available time series for this project.

4.2.4 Conclusions

In this paragraph, "Forecasting", it was searched for the best way to forecast the available time series. After analysing the time series and comparing appropriate forecasting methods it was concluded that Holt's Linear Exponential Smoothing would be the best method to forecast with.

Along the way another interesting conclusion could be obtained. The activity graph of our three clinics showed a lot of oscillation, especially recently. This suggests that a very irregular planning has been used, probably to decrease waiting lists. This implicates that only forecasting historic activity would not be enough to get a grip on real demand.

4.3 Simulation

In the previous paragraph it was concluded that forecasting with historic activity does not provide all the necessary information to create an impression about real demand. In this section another way of obtaining the right information was carried out. Instead of using historic activity, there will now be more relied upon the follow up structure, of which we know it exists. With the forecasting methods discussed, the follow up structure of the situation can only be taken into account to a limited extend. Therefore, a simulation was also carried out. In this paragraph it is explained how the existing simulation was changed to make it useful for this project and how the necessary parameters to run the simulation were determined. The simulation will be carried out and a possible stream of demand will be produced.

4.3.1 Set up

The simulation shell can be used for a queuing system with nodes and different types of customers that are called classes. Every node contains one queue and a chosen number of servers. Incoming customers arrive in the system at a queue from a node with Poisson distributed inter arrival times. The value λ of this distribution can vary per unique node-class combination. In Figure 1 an example with two nodes and three classes can be viewed. Every $\lambda_{i,j}$ represents the Poisson distribution parameter at node *i* for class *j*.

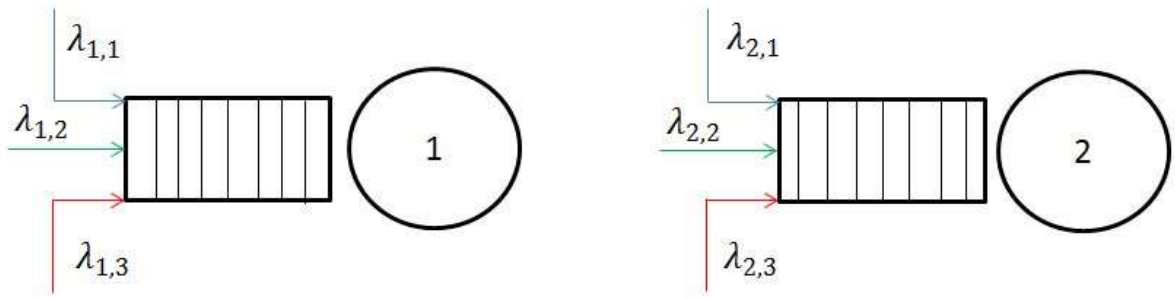

Figure 4.6: New Arrivals

When customers reach the front of the queue of their node, they will go to the first free server. The service time at a specific node depends on both the node itself and the class of the customer. A distribution can be chosen from a predetermined list of possibilities. In this project, only the exponential and deterministic distributions were used. Every node-class combination contains a unique distribution of service times.

After the service is finished, a costumer can either leave the system or be transitioned to another node, again according to the current class-node combination. When a costumer does not leave the system, the new queue that will be attended is allowed to be the queue from the node, where the customer just finished service. This all happens according to transition probabilities between nodes which can be different for every class. For one specific class (0) the situation is as follows:

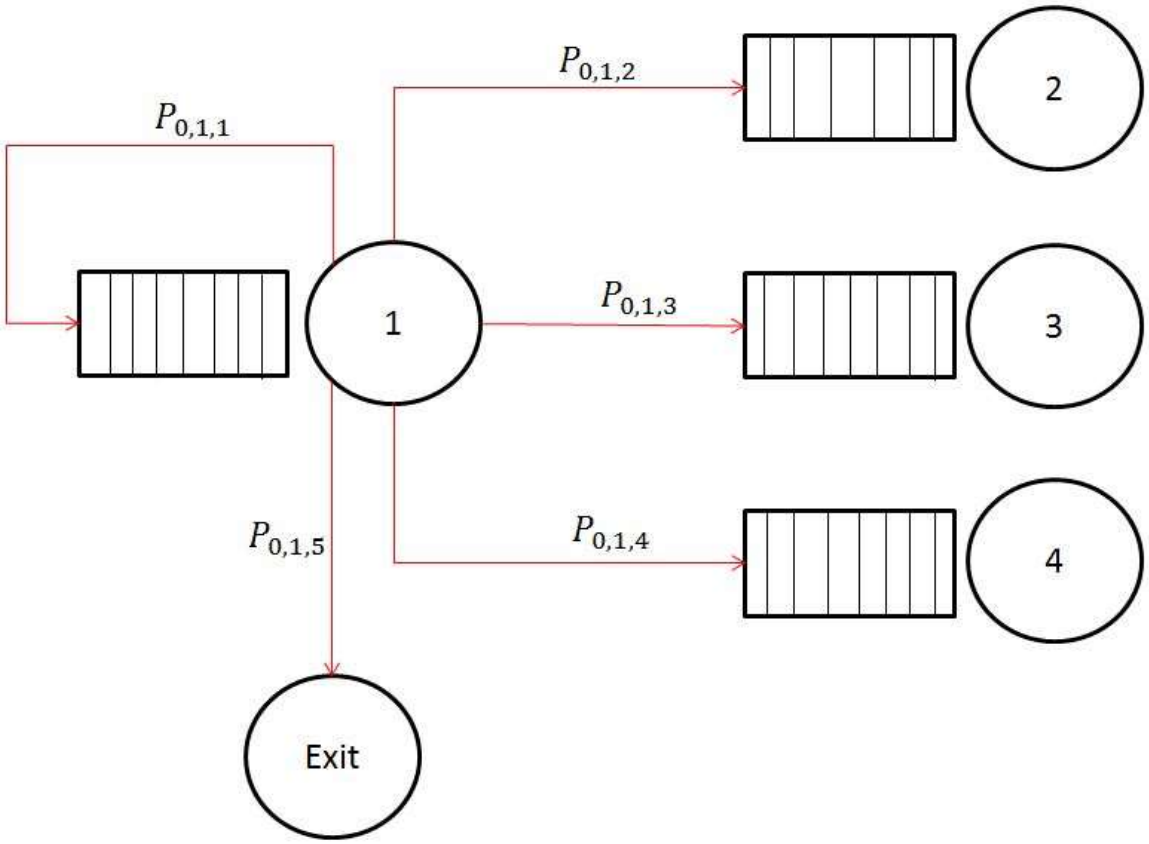

Figure 4.7: Node Transition Probabilities

This diagram can be drawn for every different class and every different node. The variable $P_{k,m,n}$ represents the probability for a customer from class *k* to go from node *m* to the queue of node *n*. Of course, for every *k* and *m* the following constraint should be taken into account:

$$\sum_{n} P_{k,m,n} = 1 \tag{26}$$

In summary, the simulation consists of different classes of customers, who each have their own arrival rates per node, service distributions per node and transition probabilities from each node.

4.3.2 Model

In this paragraph a mathematical model of the situation will be established. Therefore we firstly recollect the current situation as (partly) described in the Introduction. New patients enter the system randomly. After being seen a patient can either be assigned a follow up or be discharged. This structure can be viewed in the following diagram:

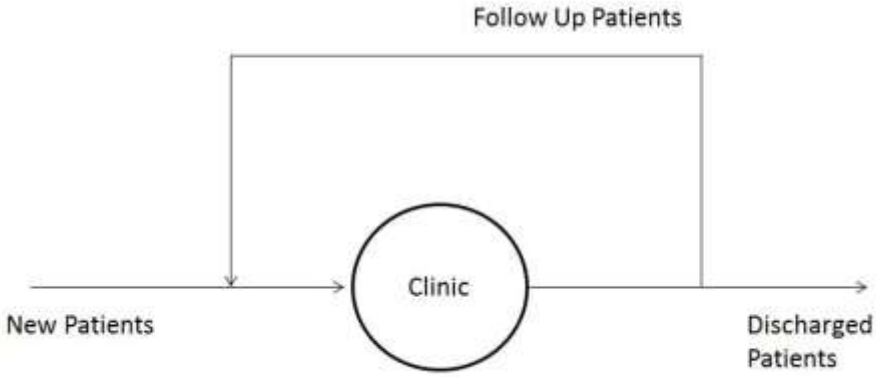

Figure 4.8: Clinic with Follow Up Structure

This was translated into a simulation point of view (See Figure 4.9). All new patients enter the system at the queue before the clinic node (1). The service represents the given appointment slot. After service a patient can either be discharged or being assigned a follow up. The follow up pause node (2), holds every patient for the time until the new appointment should take place. The service time at node 2 exactly represents the time there should be between appointments. Everybody can enter service at once in this node. To make sure there will never be a queue, this node has an infinite capacity. After service at node 2 all the patients return to the back of the queue for node 1.

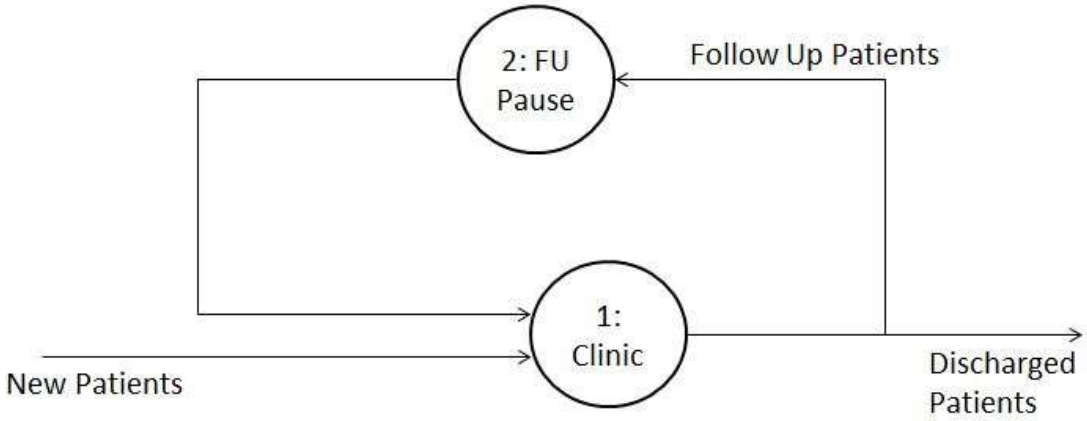

Figure 4.9: Simulation Model

For both new patients and follow ups the waiting time in the queue for node 1 corresponds exactly to the waiting time between requesting an appointment and attending an appointment. In this section only the demand, the number of requested appointments at node 1, is of importance. Therefore node 1 contains (for now) infinitely many servers and there will never be a queue.
The relevant information needed for this model can be obtained from the historic data. Firstly the process of newly arriving patients needs to be established. Then the probabilities for patients to be discharged and the average time needed between two appointments are needed.

4.3.3 Goals

With this simulation the aim is to get a grip on the real demand for appointments. Because there is only good data available about actual appointment dates and not about the request dates, a simulation without capacity limitation is needed. With this simulation, in contrast to the forecasting methods, the follow up structure of the system will be taken into account. Interesting figures to be

obtained are the monthly numbers of appointments for both newly arriving patients and follow up patients.

4.3.4 Description of Methodology

Most of the parameters that are needed to carry out the simulation can be gained from historic data. To determine a distribution for the length of the follow up pause the model needs to be refined by analysing the follow up structure. After refining the model there will be looked at the suitability of the existing simulation shell. Necessary changes will be made and explained. Finally the historically based input parameters will be discussed.

4.3.4.1 Follow Up Analysis

From the historic data, the times between appointments when a follow up was assigned have been retrieved. These times were summarized in the bar graph below. The horizontal axis represents the number of days between two appointments and the vertical axis represents the number of appointments that had a follow up after corresponding amount of time.

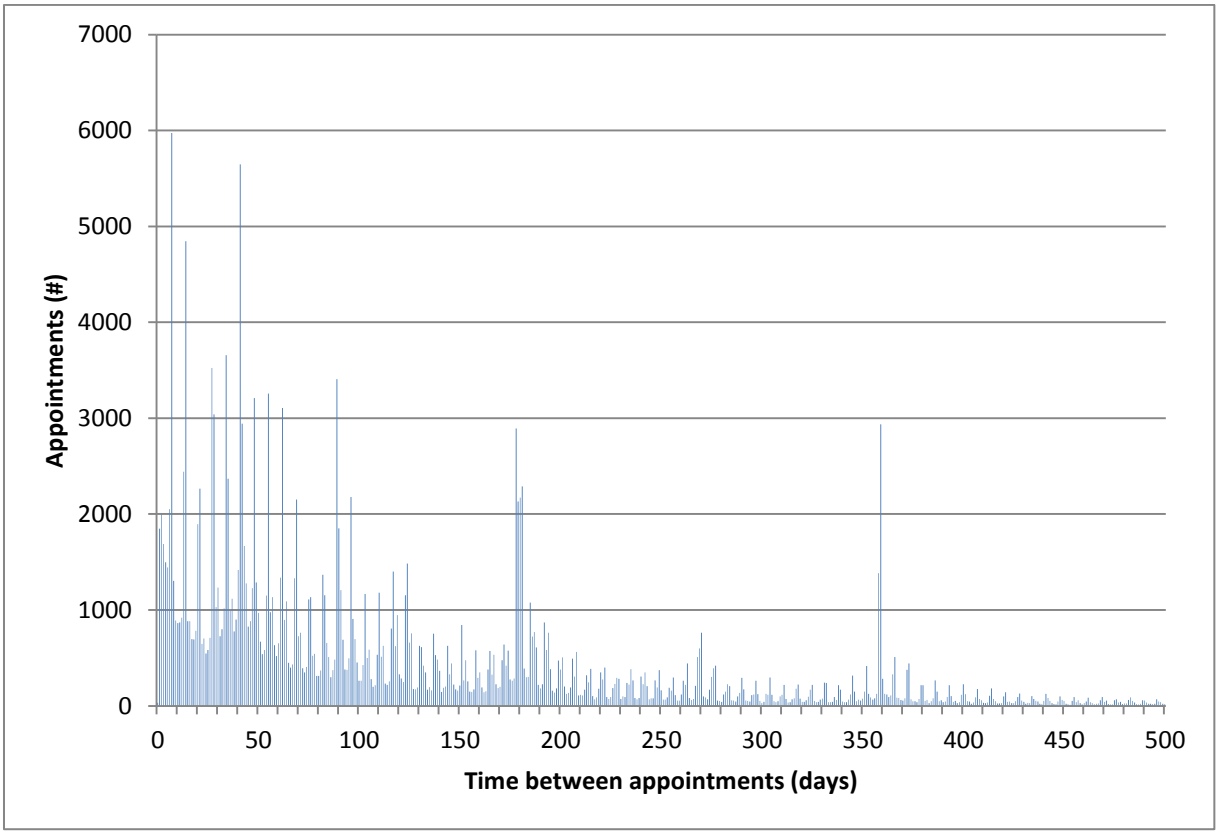

Figure 4.10: Time between Appointments

It can be concluded from this graph that there are a few clear peaks. Because the first peak can be originated from did not attendees that had to get a rearrangement as soon as possible, this project focused on the other four peaks. The shorter the time between two appointments, the larger the chance of being delayed was according to target dates and estimation. Therefore the first two peaks were adjusted a little and the second two were not. This means the follow up patients can roughly be ordered in the following four categories:

Table 4.5: Categories of Follow Up Patients

Category	Time between follow ups
1	30 days
2	90 days
3	180 days
4	365 days

This seems appropriate since it corresponds with one month, three months, six months and a year of follow up time. It is also in line with the optimized weight values obtained when using Moving Average to forecast. The strong relationship between two moments in time with three months in between is hereby explained. In the simulation the patients can now be divided in these four categories, in other words: classes.

4.3.4.2 Changes to Existing Simulation Shell

The four different classes of patients can easily be implemented in the existing simulation shell. The only thing that has to be added to Figure 6 is that the black lines (except the one for new patients) now consist of four different classes of patients. These four different classes of patients can now all have their own distribution of pausing time at the Follow Up Pause node and their own chances of being discharged.
The problem that arises is about the changes amongst classes. It was concluded from the historic data that a lot of patients have different lengths of time between follow ups in the same pathway. For example, a patient could be seen every month for a year, but after that the patient could go back to being seen only once every three months. In this case the patient should switch class in the simulation.
To be able to take these class changes into account an extra matrix with class change probabilities at each node was added to the input parameter file. The function of changing classes was added to the code. Every time a patient finishes service, the class change will be applied before the node transition. This means that the event structure of the simulation will slightly change. An extra C event, the change of class, will be added before every customer is moved, after the B event of a patient finishing service. This is shown in the following diagram:

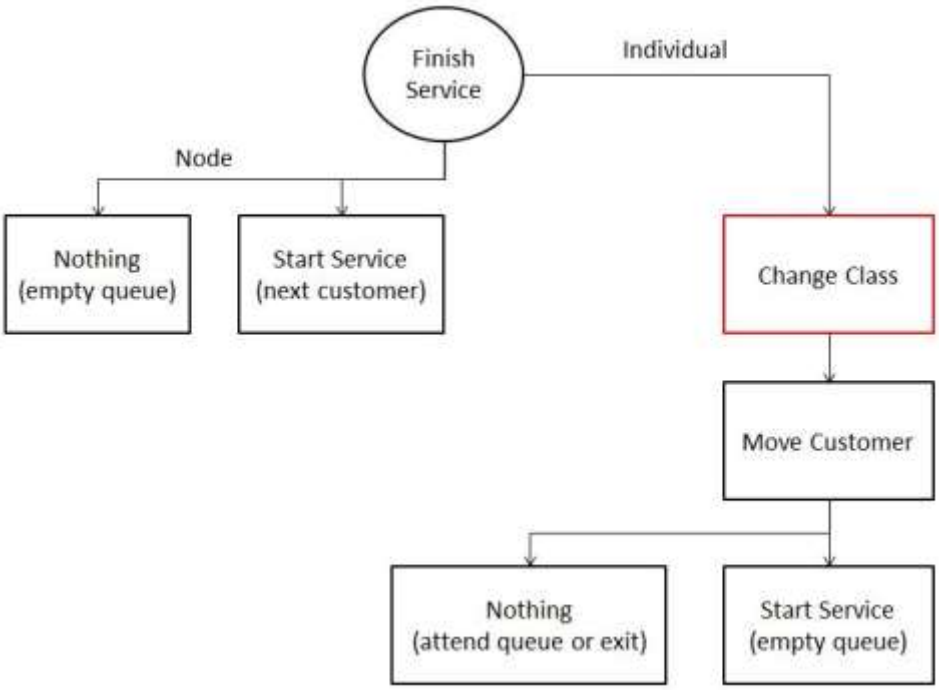

Figure 4.11: B event 2 after Class Change

It is important to conclude that, after a patients has finished service, the class will be changed before it is decided where the patient will go afterwards. Since the transferring probabilities are dependent on the class, the new class influences the following movement of the patient. Because of the complexity of the code, a lot of pieces of the code needed to be changed. Changes were made to "Simulation", "Individuals", "Data Records" and "Node".

4.3.4.3 Input Parameter File

To run the simulation, some parameters have to be determined. As discussed earlier, in this project **two nodes** and **four follow up classes** will be used. In addition to those four follow up classes (class 0,1,2 and 3) a class for newly arriving patients (class 4) and a class for exiting patients (class 5) are used. For all combinations, the arrival rates, service times, class changes and node changes need to be determined.

The most logical is to start with the arrivals into the system. The way the simulation is set up the arrivals need to be per class and Poisson distributed.
From the historic data could be retrieved when in the last years the appointments of new arrivals have taken place. This turned out to be on average 1.12 a day. Because of the target dates set by the government, there can be assumed that new patients have been prioritized and that this number is therefore fairly accurate. Since a separate class for all new arrivals was created, it holds that $\boldsymbol{\lambda_{i,j}}$ **= 0** for both nodes i and for all classes j except for class 4. New patients always arrive at the node that represents the clinic (node 1). After all nobody can start with waiting for a follow up appointment without having been seen for a first appointment. Therefore the parameter $\boldsymbol{\lambda_{1,4}}$ **= 1.12** at node 1 for class 4 patients.
The service rates were set to deterministic distributions to start with. Since there is no interest in queuing at node 1 yet, the number of servers is set to **infinity** and the service rate should be really low. This way everybody will pass node 1 fairly quickly and it is possible to examine the number of patients that would pass node 1 if everybody could be seen. Therefore the fixed service times at node 1 were set to $\boldsymbol{\mu_{1,i}}$ **= 0.01** days for all *i* classes. At node 2 the service times have to correspond

with the time until the next follow up should be. For every patient the pause time until the next appointment starts immediately, consequently node 2 needs **infinitely many servers** as well. With μ_i representing the service time for class *i*, the service times at node 2 were set to $\boldsymbol{\mu_0}$ **= 30** days, $\boldsymbol{\mu_1}$ **= 90 days**, $\boldsymbol{\mu_2}$ **= 180** days and $\boldsymbol{\mu_3}$ **= 365** days. Class 4 and 5 patients will never enter node 2 because of the way the simulation is set up. Class 4 patients enter the system at node 1 and will be changed into another class at the end of service at node 1. Class 5 patients leave the system with probability 1 from node 1 and since the class change takes place after service, class 5 patients will never have service at any node.

As mentioned in the previous paragraph, classes of patients can change during one pathway. For the changes amongst classes two matrices are used; one for each node. At the follow up pause node (node 0), no class changes are made. This matrix is an **identity matrix**. For the class changes at the clinic node (node 1), changes within pathways were analysed in the historical data. The percentages retrieved from the historical data for the 4 follow up classes were as follows:

$$\begin{pmatrix} 0.59 & 0.24 & 0.10 & 0.07 \\ 0.19 & 0.37 & 0.29 & 0.15 \\ 0.11 & 0.20 & 0.37 & 0.32 \\ 0.12 & 0.16 & 0.29 & 0.43 \end{pmatrix}$$

It was expected that most follow up patterns would stay in the same category. Therefore, the numbers on the diagonal were expected to be highest. This is true, though without a large margin. There is a possible explanation for this however.

In which category a follow up was placed, was based on the time until the next appointment was booked. In some cases, this time might be longer than it should have been according to the original target date, causing the follow up to end up in a larger class than it should have been. In this project there was target date information available on a small portion of the original dataset. From this information it could indeed be concluded that a small percentage (about 6%) of the appointments should have ended up in a smaller class. Therefore the numbers were corrected to create a stronger diagonal. The class change matrix was adjusted to the following:

$$\begin{pmatrix} 0.65 & 0.21 & 0.08 & 0.06 \\ 0.19 & 0.43 & 0.25 & 0.13 \\ 0.11 & 0.20 & 0.43 & 0.26 \\ 0.12 & 0.16 & 0.29 & 0.43 \end{pmatrix}$$

Of course it was taken into account that the further a number was from the diagonal, the less it should be changed. After determining this matrix the remaining two classes have to be examined. Of course no patient will ever change to a new arrival class patient. The column which represents changing to class 4 should therefore contain only zeros.

Class 5 represents discharge rates. This means the last column contains the chances of getting discharged per class. The discharge rates were first based on the historical data column “event outcome”, in which a specific code a discharge meant. After analysis however, it was discovered that this information was not accurate enough and yielded too low discharge rates. Revised discharge rates were then based on all the patient-pathways that did not have any appointments after 2013. This means this patients could only still be in the system when they had a 2 year follow up to begin with, which is highly unlikely compared to the concluded follow up structure.

The concluded chances of being discharged per class are given in the column below.

$$\begin{pmatrix} 0.14 \\ 0.15 \\ 0.19 \\ 0.25 \\ 0.33 \\ 1.0 \end{pmatrix}$$

The first number for example can be interpreted as follows: 14% of the patients coming from a follow up pause of type 0 will be discharged after service at the clinic. The fourth row gives a relatively high number, this seems logical because this is the chance of a new arrival immediately being discharged. The last number of the column is irrelevant because, as mentioned before, a class 5 patient will never enter any service and therefore never again change class.
Last, if new patients are not immediately being discharged, they need to be sorted into one of the four follow up classes. To be able to divide the arrivals between the four classes there was looked at the historical data again. Around a quarter of the inter appointment times falls into every category. In other words, they are evenly divided within the system. Therefore the choice was made to divide the arriving patients evenly in those four categories.
Everything discussed about class changes can now be combined in one matrix for node 1. The percentages from the matrix above have to be recalculated, to make sure every row still adds up to 1 combined with discharge rates:

$$\begin{pmatrix} \mathbf{0.559} & \mathbf{0.1806} & \mathbf{0.0688} & \mathbf{0.0516} & \mathbf{0.0} & \mathbf{0.14} \\ \mathbf{0.1615} & \mathbf{0.3655} & \mathbf{0.2125} & \mathbf{0.1105} & \mathbf{0.0} & \mathbf{0.15} \\ \mathbf{0.0891} & \mathbf{0.162} & \mathbf{0.3483} & \mathbf{0.2106} & \mathbf{0.0} & \mathbf{0.19} \\ \mathbf{0.09} & \mathbf{0.12} & \mathbf{0.2175} & \mathbf{0.3225} & \mathbf{0.0} & \mathbf{0.25} \\ \mathbf{0.1675} & \mathbf{0.1675} & \mathbf{0.1675} & \mathbf{0.1675} & \mathbf{0.0} & \mathbf{0.33} \\ \mathbf{0.0} & \mathbf{0.0} & \mathbf{0.0} & \mathbf{0.0} & \mathbf{0.0} & \mathbf{1.0} \end{pmatrix}$$

Apart from class changes, transitions among nodes occur after service. Information about this is stored in so called transition matrices. Per class there was a 2x2 transition matrix created. For class 0 until 3 this matrix is the same and looks like the following:

$$\begin{pmatrix} \mathbf{0.0} & \mathbf{1.0} \\ \mathbf{1.0} & \mathbf{0.0} \end{pmatrix}$$

This means every patient is always transferred to the other node than where the patient was coming from. As discussed earlier, nobody can ever change into a patient of class 4, so the transition matrix for class 4 is never used, thus irrelevant. The transition matrix for class 5 is interesting however. All the patients with this class leave the system immediately. This means the transition matrix will only consist of zeros. The simulation will make all remaining patients leave the system, which will be all of them in this case.
Since at both nodes the amount of servers is set to **infinity**, there will never be any queue. Therefore the queue capacity is irrelevant and can be set to **infinity** as well.
The simulation time has to be a few years to warm up the system, because of the relatively large amount of follow up time for class 3 patients. Thus, a reasonably high amount of **5000** days was used as simulation time.

4.3.4.4 Warm up time

It was intended to find out more about the current demand. The Simulation made it possible to recreate the situation, but of course it starts with an empty system. For the results of new patients

this does not matter, since their arrivals are independently Poisson distributed. An example of what this new patients demand could look like can be viewed in figure 4.12 below.

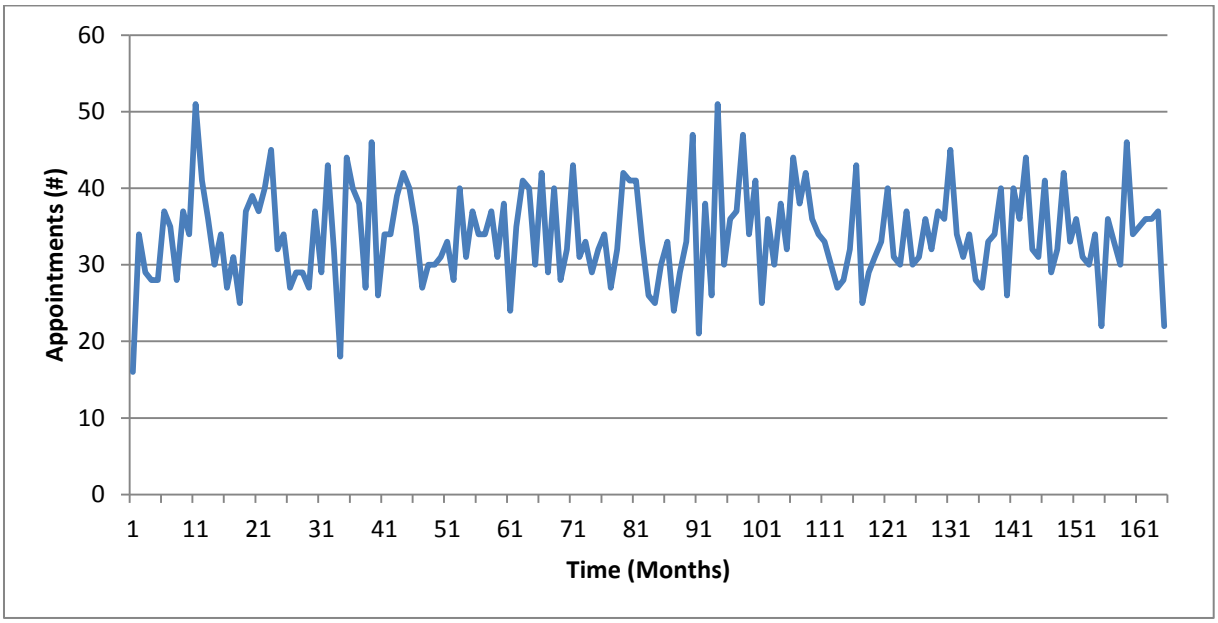

Figure 4.12: Demand New Patient Appointments

For follow up patients however, it is a different situation. The more patients are in the system, the higher the demand for follow ups will be. Therefore, to measure the simulated demand will only be relevant when the system reaches steady state. Because patients are discharged (i.e. leaving the system) according to percentages of the total number of patients, it is clear that the system will reach a steady state at some point. In the graph below, figure 4.13, the monthly demand for follow up appointments is shown. The system seems to reach the steady state after approximately 80 months. That is why 80 months was chosen as a warm up time. All the information of the first 80 months of simulation will therefore be neglected. In the next paragraph it will be explained how the relevant information was stored.

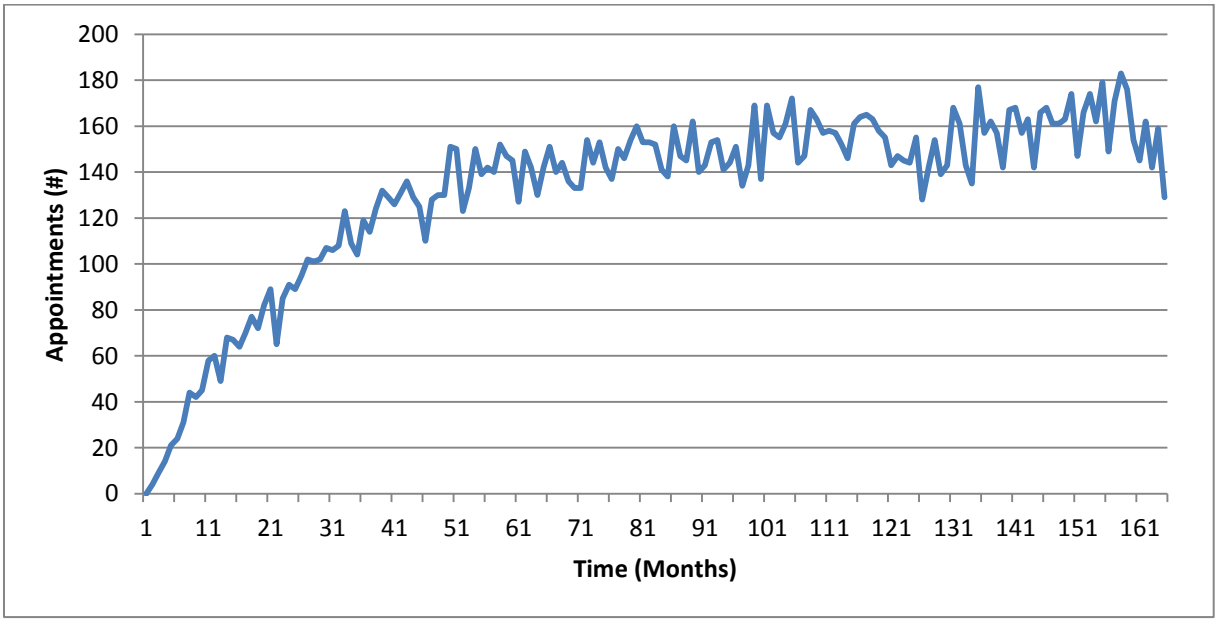

Figure 4.13: Demand Follow Up Appointments

4.3.4.5 Number of runs

To be able to get reliable information, one run is not sufficient. Therefore, multiple runs needed to be carried out. It was decided to start with 85 runs and to test whether this would be sufficient. The number 85 was chosen because this was the maximum number of runs of which the results could be easily stored in one excel file.
An extra function, “multiple runs”, was created to make the simulation run multiple times, with as input the number of runs. The output will contain the old output for every run, but also some extra output files were created. In both extra output files every row represents a single run. In one output file every row consists of all the monthly number of appointments for new patients that were registered. In the other output file every row consists of all the monthly number of appointments for follow up patients that were registered.

4.3.5 Analytical Verification

In this paragraph it was checked whether the simulation ASQ actually does what it was meant for. This can be done by testing extreme cases and analysing the output data. It was started with some simple checks.
It was started with looking at whether the dynamic classes work in the way they were supposed to. We can distinguish two classes that were used in a distinctive way in this project. For example it was chosen to make class 4 the class for all newly arriving patients. This means class 4 patients can in our situation only be served at the clinic node and never at the pause node. This is a result of the fact that all arriving patients enter the system at the clinic node and after service at the clinic node every patient undergoes a class change. Since no patient can change into a class 4 patient, not even a class 4 patient himself, in the rest of the system there can never be a class 4 patient. This was checked in the output files and it was confirmed that no record of an appointment at the pause node of a class 4 patient could be found.
Another example of a distinctive way to use a class is the one of discharged patients, class 5. Because any patient receiving the class 5 will immediately be discharged, there can be no record of a class 5 appointment at either node. This was confirmed by checking the output files as well.
After checking dynamic classes it was looked at some actual figures. The number of patients, both new arrivals and follow up patients, that passed through clinic was counted. In the table below the results can be viewed for a particular run:

Table 4.6: Number of Appointments New and Follow Up

Follow up patients	17912
New patients	5623
Total	23535

It could be concluded that this makes sense with the parameters that were chosen. These numbers could be expected with the chosen arrival process and discharge rates discussed in the paragraph “Parameter File”.
The numbers of appointments for every class at the pause node were also counted. In the table below an example can be viewed for the same particular run that was used as an example in the table above.

Table 4.7: Number of Appointments per Follow Up Class

0	5568
1	4667
2	4617
3	3644
Total	18496

These numbers are also in line with the expectations that could be concluded from the parameters used. In this case the change matrix for classes and discharge rates suggest this kind of results.

On top of the short analysis above an extra tool was used to test the working of ASQ. The same simulation was set up in Simul8. The advantage of Simul8 is that it uses a very simple interface and creates a more visual model. It is also much easier to deduce extra values, which can help to check for mistakes. Unfortunately, the model in Simul8 is much less easy to extend than the model in ASQ. This means that is much less appropriate for our project. On top of that Simul8 is not an open source program, which makes it less accessible than ASQ. This is why Simul8 was only used as an additional check on ASQ.
The Simul8 model was built at the same time as developing the ASQ model. Comparing along the way helped in finding mistakes in an early stage and the Simul8 model can again be of use in the final stage by means of comparison.

To simulate the new arrivals, in both simulations a Poisson process with the same parameters were used. Of course they yield therefore very similar results. It is more interesting however to compare the number of follow up appointments at the clinic. In the table below some interesting figures, results of 85 runs, of both simulations were compared:

Table 4.7: Comparing Simul8 and ASQ

	Simul8	**ASQ**	**difference**
average monthly number of FU appointments	109,4938	107,8567	1,637022
standard deviation	2,361687	2,49072	-
average monthly number of FU appointments after warm up	128,9046	126,5546	2,349933
standard deviation	3,460474	3,138737	-
average monthly number of FU appointments from the last simulated year	130,5843	127,9819	2,602413
standard deviation	5,67118	5,705159	-

It can be concluded that Simul8 and ASQ yield very similar results. The small difference that can still be detected is due to a different set of random numbers. It was concluded that the simulation in ASQ works in the way it was meant to.

4.3.6 Validation

In the previous paragraph it was concluded that the simulation in ASQ does exactly what it was meant to do. In this paragraph it will be discussed whether it provides information that is in line with reality.
First it was looked at the appointments in clinic. The results of 85 runs, after warm up time, were compared to the registered activity at the three clinics.

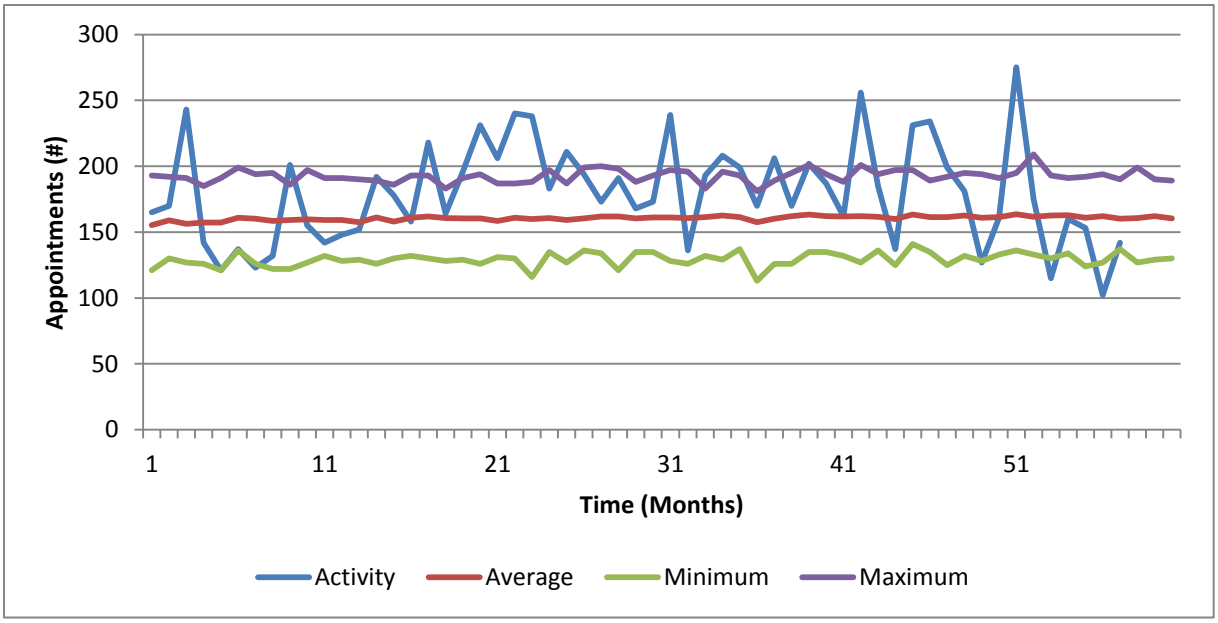

Figure 4.14: Simulated Demand Total and Activity Total

It seems like the simulated demand is too low compared to the registered activity. Two factors we could not take into account, because of the limited amount of time, could have influenced this. Firstly, the way the new arrival process was determined could have been a co-cause of the too low results. This can be viewed in the following graph:

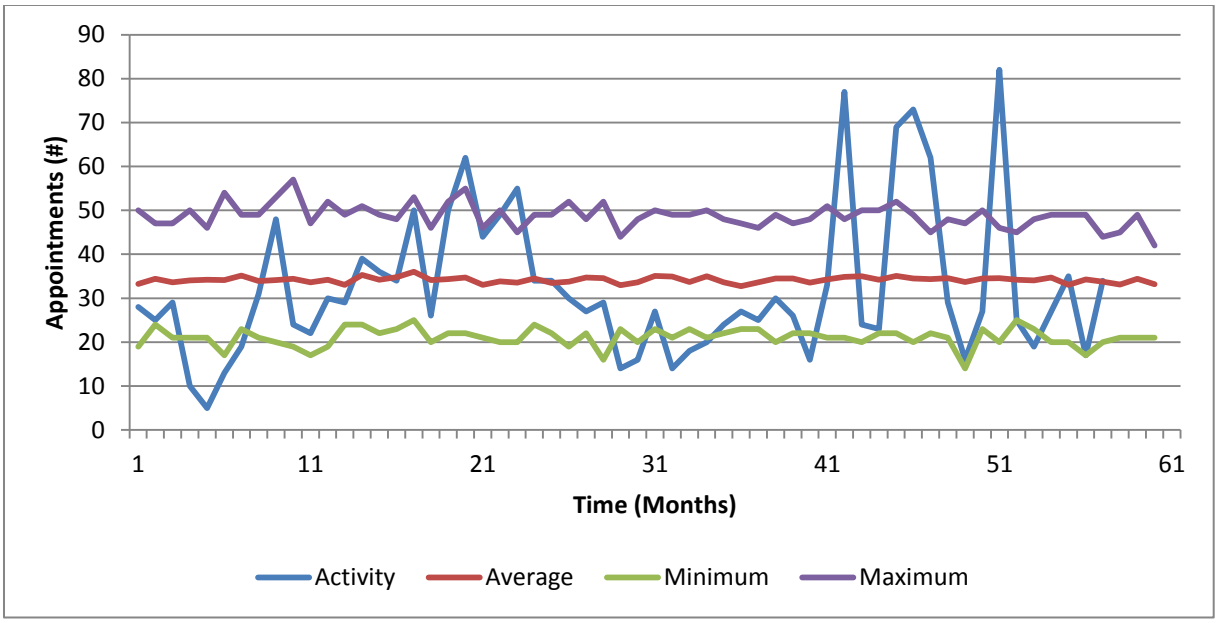

Figure 4.15: Simulated Demand New and Activity New

Remark: For only new arrivals it is not necessary to make use of a warm up time because it is only a Poisson process. To make it comparable to the results above however, the graph above does represent the results after warm up time.

It was assumed that the new arrivals were all seen by a clinician, because of the governmental targets. Safer to assume is that most new arrivals were seen but still not all. To take this into account the arrival rate was increased by 5 percent, 10 percent and 15 percent. The results for new arrivals are shown below.

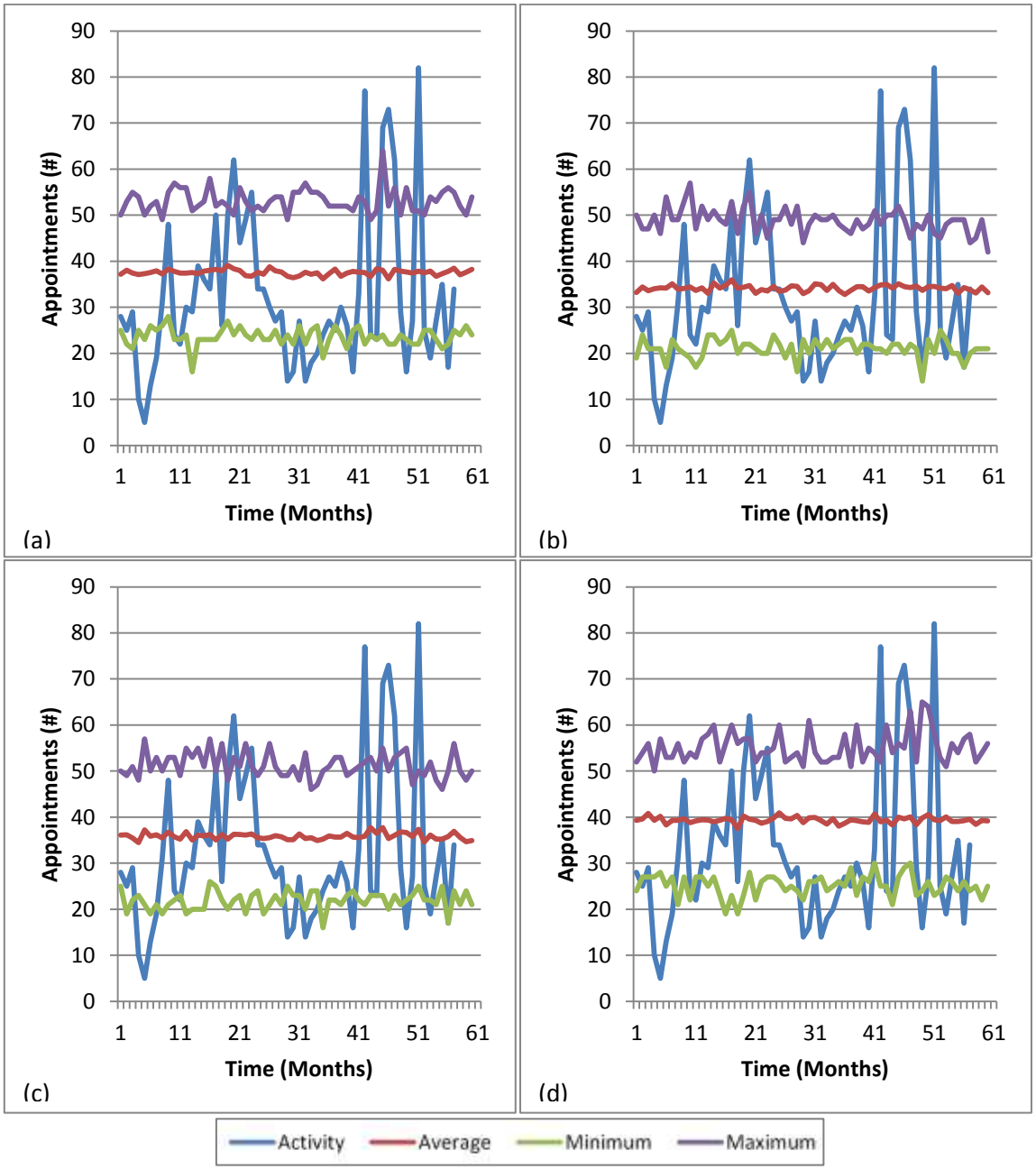

Figure 4.16: (a) 0,(b) 5,(c) 10,(d) 15 Percent Increased Demand New and Activity New

Increasing more than 15 percent is highly unlikely because of new patient activity. As can be seen in the last graph the average demand is already much higher than the average activity. Increasing even more would make the demand on new patients so much higher compared to follow up patients that the drops in the activity graph would be very weird.

Out of these four situations the best one needs to be chosen however. Which of those situations is most realistic, is hard to decide. To be able to make it a little easier, for all four situations it was checked what effect it would have on the total demand.

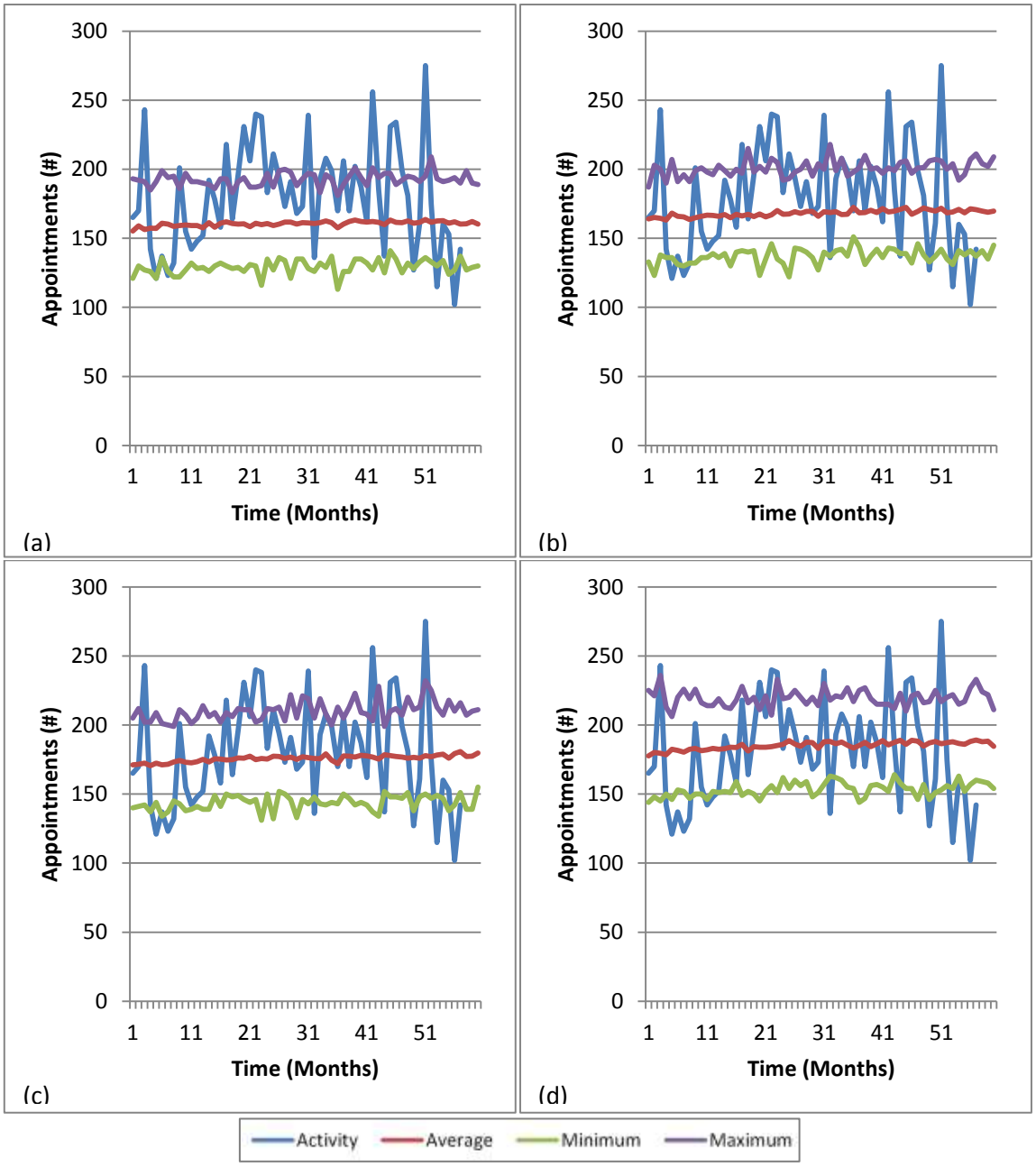

Figure 4.17: (a) 0,(b) 5,(c) 10,(d) 15 Percent Increased Demand Total and Activity Total

The last two simulations of demand seem to be most in line with the registered activity. In fact, the linear trend line of the total activity can be drawn and ends up exactly between the average lines of the total demand resulting from 10 percent and 15 percent increased new arrivals. This is shown in the figures below.

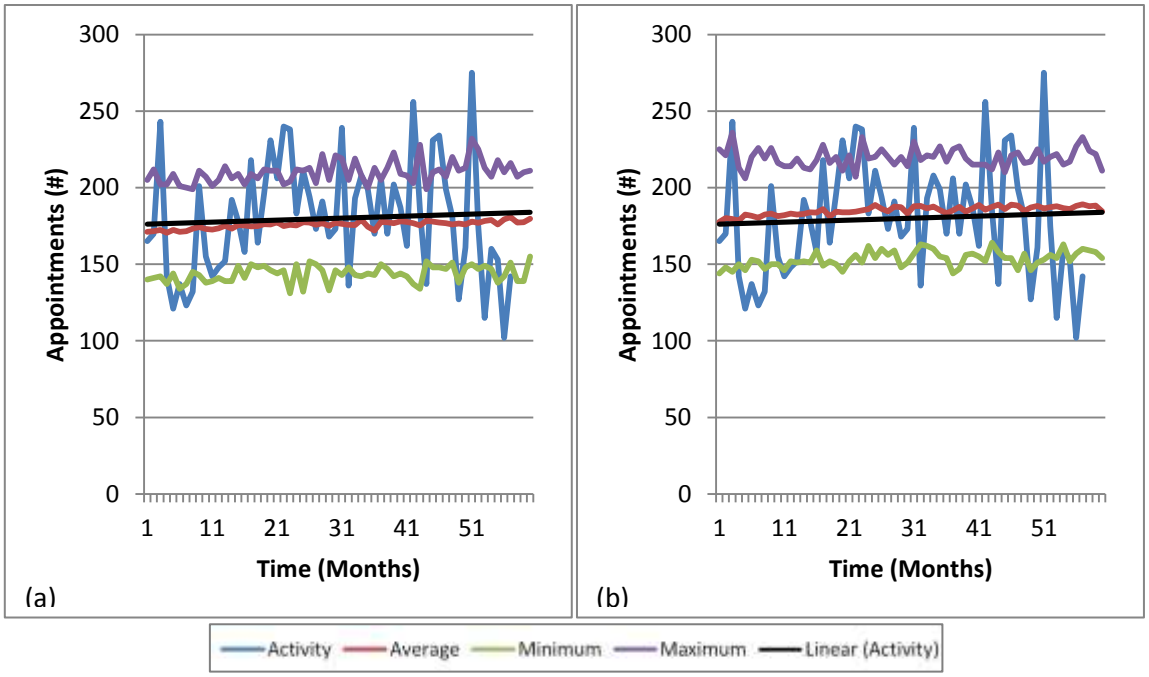

Figure 4.18: (a) 10,(b) 15 Percent Increased Demand Total and Activity Total Including Trend lines

The reason for this project was that patients in general have to wait very long to get an appointment. This suggests that the actual demand is higher than the registered activity. Because we have years of data, it is very unlikely that the system is still compensating for an enormous peak in demand before. Therefore, increasing new demand by 15 percent seems most realistic. Increasing more than 15 percent would not be appropriate in relation to new patient activity, but there is another way of improving the simulation further.

The second factor that was not taken into account at first is the proportion of Did Not Attendees (DNAs). In the historic data this seems to be quite a large percentage, 16 percent in 2015 and 20 percent if calculated over all the available data.
This means that all the patients who were DNAs, who needed a new appointment to be scheduled and whose original slot was wasted, actually used two slots instead of one for only one appointment. In other words the demand would be higher if this was taken into account. Therefore the complete line should be lifted a little. This is not very simple to implement in the simulation because a DNA should keep the old class. Therefore the complete situation was just lifted by again 5, 10 and 15 percent. Again it is hard to decide which one would be most realistic.

It was decided to work with the 15 percent increase, because this puts the most pressure on the capacity. For the rest of the project this makes it most interesting. This means it is assumed that almost all DNA slots are wasted. A further study of DNAs would show whether this is a correct assumption. Since this percentage is not a part of the simulation it can very easily be changed with hindsight.

The concluded demand that we will work with in the next chapter looks therefore as follows:

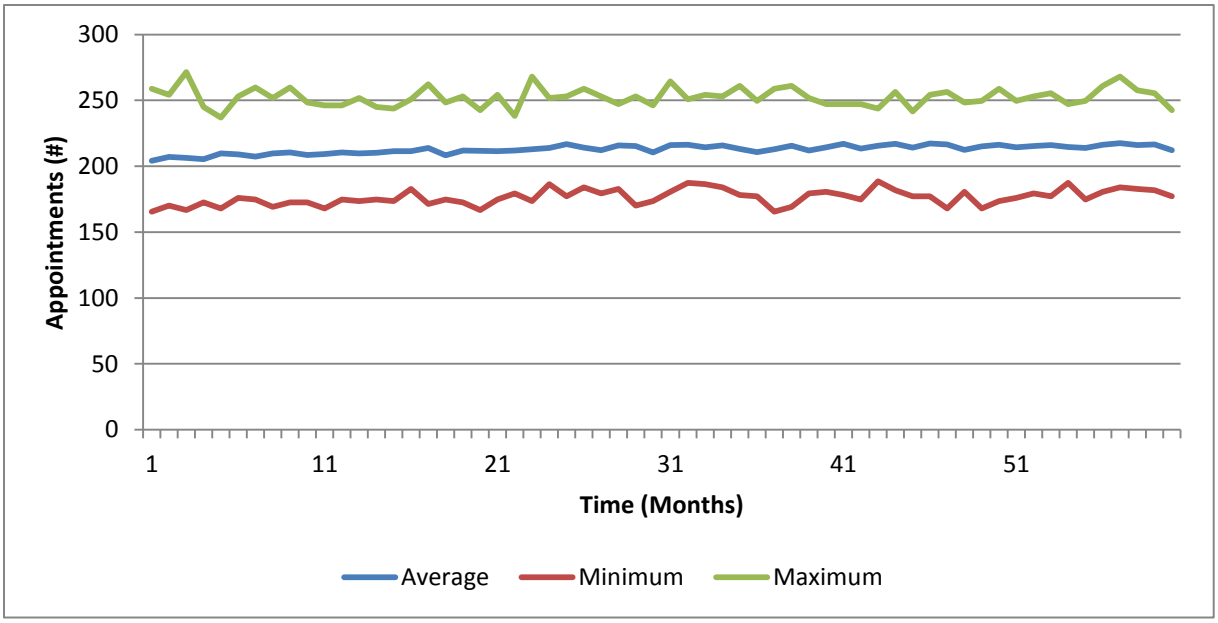

Figure 4.19: Concluded demand

4.3.7 Numerical Results

The results of many runs can best be summarized and viewed in a so called boxplot. A boxplot provides a graphical way of summarizing numerical data. The bottom of the box represents the first quartile and the top represents the third quartile. The band in the middle of the box represents the median. The vertical lines above and beneath the box are called the whiskers and indicate the variability outside the box. The occasional circles are outliers and the numbers attached are the associated numbers of the data points. The small horizontal lines at the end of the whiskers represent the minimum and maximum values of the data, outliers excluded.
We start with a boxplot of the total determined demand over 60 months in steady state for 15 runs. This can be viewed in the figure below.

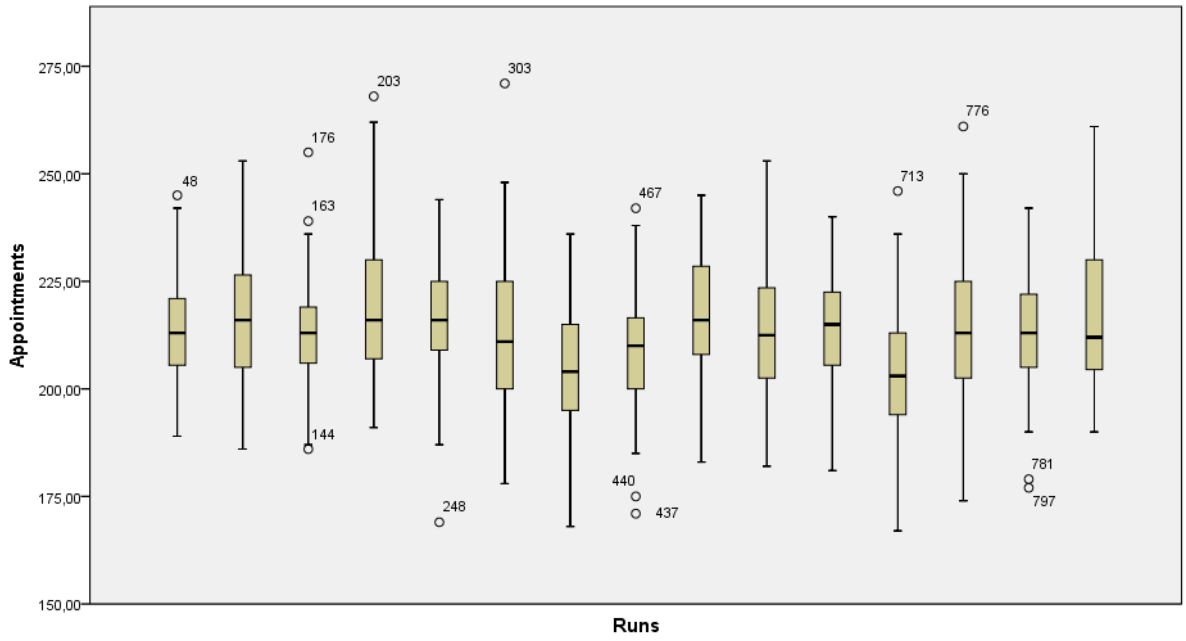

Figure 4.20: Boxplots of 15 runs over 60 months in steady state

These 15 runs provide very similar results. With the results of all 85 runs, including the 15 used above, one large boxplot was created as well. Also for the same 60 months of steady state for every run. This boxplot can be viewed below.

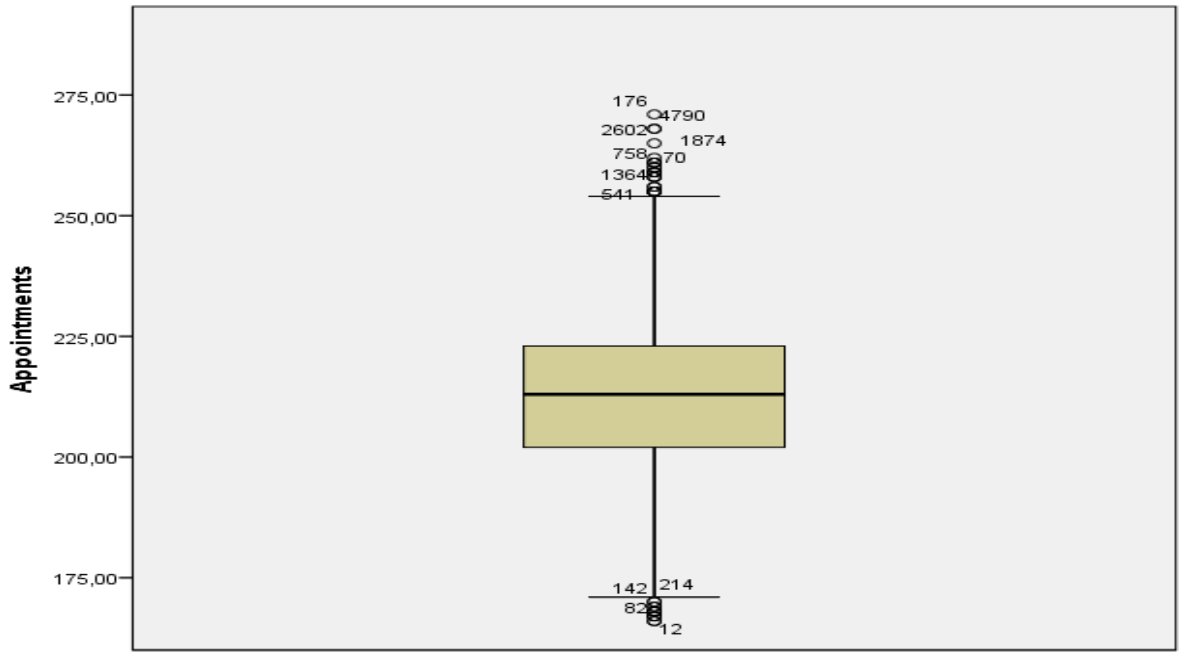

Figure 4.21: Boxplot of all 85 runs over 60 months in steady state

4.3.8 Conclusions

By means of the simulation in ASQ the demand in a system with a follow up loop can be determined. The time between two appointments, in other words the time before a follow up is needed, can differ for different types of patients. It is also possible to make the types of patients, classes, dynamic. This means that classes can be changed during the process. Different discharge rates can be added for every different type of patient. Only the arrival process needs to be Poisson, though the parameter can be changed.
The follow up structure for the three clinics OPHT103, OPHT15 and 987/988 was simulated and a stream of demand was generated. The used model can easily be extended to include more sophisticated follow up patterns and more classes of patients. What cannot yet be taken into account however, are the Did Not Attendees. The obtained demand should therefore be lifted according to the Did Not Attendees percentage of the specific clinic(s).
The information obtained about the demand in the situation of this project was summarized in the following graph:

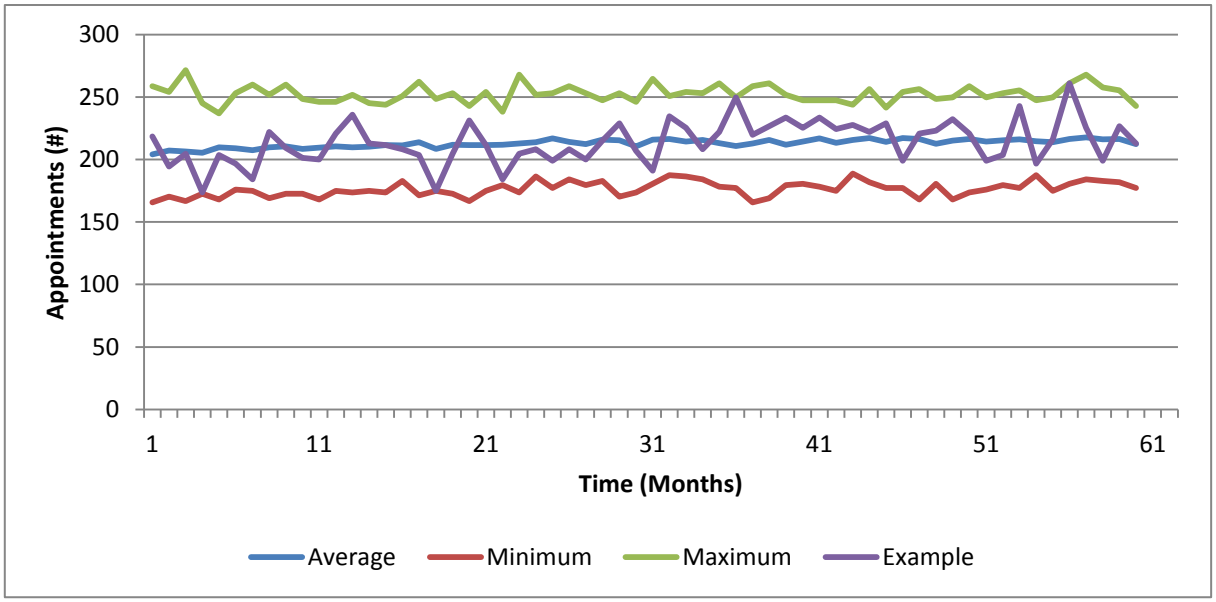

Figure 4.22: Concluded Demand Including an Example Run

According to the carried out simulations the actual demand for appointments is a lot steadier than the activity graphs show. This implicates (again) that a lot of improvement is possible with a smarter capacity planning tool. An example run, as can be viewed in the figure above, can be taken out of all the runs to be used to optimize the use of capacity with.

4.4 Summary

It was intended to find a way to determine how best to make use of capacity. To be able to conclude something about this, the demand is necessary. In this section it was intended to get a better grip on the real demand on the system of the three clinics OPHT103, OPHT15 and 987/988. In two different ways the historic data was used to obtain this better grip.
Firstly, it was chosen to analyse the available data and use forecasting methods. The capacity could then be optimized in relation to the forecasted demand. Holt's Linear Exponential Smoothing turned out to be the best way to forecast historic activity with. The problem that arose however was that, after analysing the data, it was concluded that the historic activity does not provide a clear picture about the previous demand because it was limited by capacity. Therefore another way of determining the demand needed to be tried.
Secondly, the historic data was used to determine the necessary parameters to run a simulation with. This simulation, built with Python, was based on the follow up structure of the system rather than previous activity. A simulation model was created, which is very easily extendable and adaptable to (slightly) different situations for other clinics or even departments. It was checked with a simulation using a completely different, more visual software package, Simul8, whether the simulation works properly.
For the specific situation used in this project multiple runs were carried out to obtain the better grip on demand. Upper and lower bounds were determined for demand of both new patients and follow up patients. An example run can be used to base the decision about how to allocate capacity on. In the next chapter the optimal use of capacity, with respect to the determined demand, will be discussed.

5. Capacity Planning

In this chapter it will be looked at how best to divide the available capacity over new patients and follow up patients. This will be done by means of an optimization. After that a queuing model will be used to get an impression about the pressure on the system. Finally, a simulation will be carried out again but now including the restriction of capacity. This provides more precise information about how (planning of) capacity influences waiting times.

5.1 Current Situation Capacity

In the previous chapter a good impression of the actual demand on the system has been determined. To be able to give this demand proper meaning it has to be compared to the capacity. About the three clinics that are used in this project some information about the *current* weekly capacity was provided. In the following table the information is shown:

Table 5.1: Current capacity

	New Patient Slots	Follow Up Patient Slots
OPHT103	0	24
OPHT15	32	104
987/988	12 (Monday)	80 (48 Monday and 32 Thursday)

This comes down to a total number of 256 slots a month. This is much more than the registered activity in the last few years, which are 180 a month on average. This might be due to recent changes in capacity. But there exists another factor as well, which could have caused this difference. Most clinics do not run a 100 percent of the time during the year. The clinicians have of course annual leave and they get days to put to the use of training as well. This means the clinics can only run about 80 percent of the time. This changes the monthly capacity to only 205 slots on average to work with.

5.2 Optimization

5.2.1 Goals

If the capacity cannot be adjusted, the only decision that can be made is how to divide the available capacity between new patients and follow up patients. Clearly, the decision variable in this optimization will be number of slots assigned to either new patients of follow up patients. The remaining slots will automatically be assigned to the other one. This decision needs to be made, aiming for a minimization of waiting times for patients. Therefore, the objective function of the optimization will be a count of the waiting time.
Furthermore, the optimization is meant to be a usable tool that, with a fixed capacity and two streams of demand (one for new patients and one for follow up patients) as input, provides an output consisting of the allocation of the given capacity.
For the subset of clinics this project was focused on, the optimization will be carried out. It should be kept in mind however that the optimization has to be extendable for a much larger problem.

5.2.2 Xpress MP Optimization

With a stream of demand and a fixed capacity number, the optimization written in Xpress MP will give the best way to allocate the capacity over new patients and follow up patients over the chosen period. This best way is defined as the way in which as little people as possible cannot be helped in

their requested month. It does not take into account how much many time units patients have to be passed through, but just the number of patients. The reason it was implemented like this is about software limitations. In this version it is not possible to make loops that use decision variables. This makes it impossible to link waiting patients from different time units. Even if this were possible it is debatable whether it would be useful, because it will demand a lot of computations. Since a first come first served system is used, it was expected that this will not have a lot of impact. In the next paragraph this will be discussed further.
In the previous chapter a list of possible demand streams were constructed. It was chosen to work with the most pessimistic one, to test the system for the hardest pressure possible. From the demand streams obtained in the last chapter, 15 runs were randomly chosen to carry out the optimization. Out of every run the same 6 months out of the steady state were chosen to optimize on. Optimizing over 6 months was chosen to make it realistic for actual use of forecasted values. The results of dividing the capacity of 205 slots a month are as follows:

Table 5.2: Xpress MP Optimisation Results

	New Patient Slots	Follow Up Patient Slots	Wait
Run 1	46	159	66
Run 2	46	159	46
Run 3	39	166	28
Run 4	43	162	72
Run 5	44	161	50
Run 6	40	165	140
Run 7	43	162	67
Run 8	41	164	69
Run 9	38	167	85
Run 10	41	164	92
Run 11	41	164	81
Run 12	47	158	2
Run 13	45	160	20
Run 14	51	154	39
Run 15	45	160	43

5.2.3 Excel Solver

In this project it was focused on a relatively small dataset. Also the optimizing is done over only two things, new patients and follow up patients. Therefore it is possible to do the same optimization as explained in the previous paragraph with Excel Solver. In this optimization the loop through the stream of demand is possible. It has to be kept in mind that this version of the optimization is really hard to expand and is therefore not suitable for use on a bigger scale. In addition to that, Excel Solver uses a very crude way of optimizing, which makes the results dependant on the starting values. It is possible that the program gets stuck in a local optimum and does not provide the right answer. This optimization was therefore only used to check whether the optimization in Xpress MP gives appropriate results with respect to waiting times. To be able to compare the results, the same 15 streams of demand of 6 months as in the previous paragraph were used as an input for this optimization. The results for dividing the capacity of again 205 slots a month are as follows:

Table 5.3: Excel Solver Optimisation Results

	New Patient Slots	Follow Up Patient Slots	Wait
Run 1	47	158	138
Run 2	44	161	105
Run 3	42	163	61
Run 4	41	164	188
Run 5	50	155	111
Run 6	44	161	437
Run 7	42	163	176
Run 8	44	161	249
Run 9	36	169	265
Run 10	35	170	287
Run 11	38	167	256
Run 12	47	158	2
Run 13	43	162	34
Run 14	53	152	75
Run 15	51	154	135

5.2.4 Comparing Methods

The results for dividing capacity of both optimizations seem to be very similar. As mentioned before, the Xpress MP optimization is more extendable and uses linear programming. This means a reliable way of optimizing, but it does have its limitations. On the other hand Excel Solver is not very easily extendable, but it uses a more crude way of optimizing, which allows for optimizing on the waiting time.
The outcome regarding the division of slots is very easily comparable. Both the averages and standard deviations come very close to each other. Comparing the resulting waiting times is harder however. For in the first case the wait means the number of patients that could not be helped immediately, and in the second case wait means the actual number of months people have to wait. Therefore the average numbers of waiting time and number of patients were calculated for both methods. It should be kept in mind however that, though these numbers can be calculated with hindsight, it does not change the fact that the methods did not optimize on them.
In the following table the two methods are compared on a few characteristics:

Table 5.4: Comparison of Optimization Methods

	Xpress MP	Excel Solver
Easily Extendable?	Yes	No
Optimizes on:	Number of patients that have to wait	Number of months that patients have to wait
Possible to provide wrong answer?	No	Yes
Average New Patient Slots	43.33	43.8
Standard Deviation New Patient Slots	3.46	5.23
Average number of patients that have to wait	60	62.4
Average number of months patients have to wait	181	168

Because both optimizations treated the system on a first come first serve basis, the difference between optimizing on waiting time and number of patients waiting is very small.

The most important question is whether both optimizations provide similar divisions of capacity. Therefore in Figure 5.1 it was shown how many slots were assigned to new patients for each run using the different methods. According to standard deviations the Xpress MP line should be more flat than the Excel Solver line.

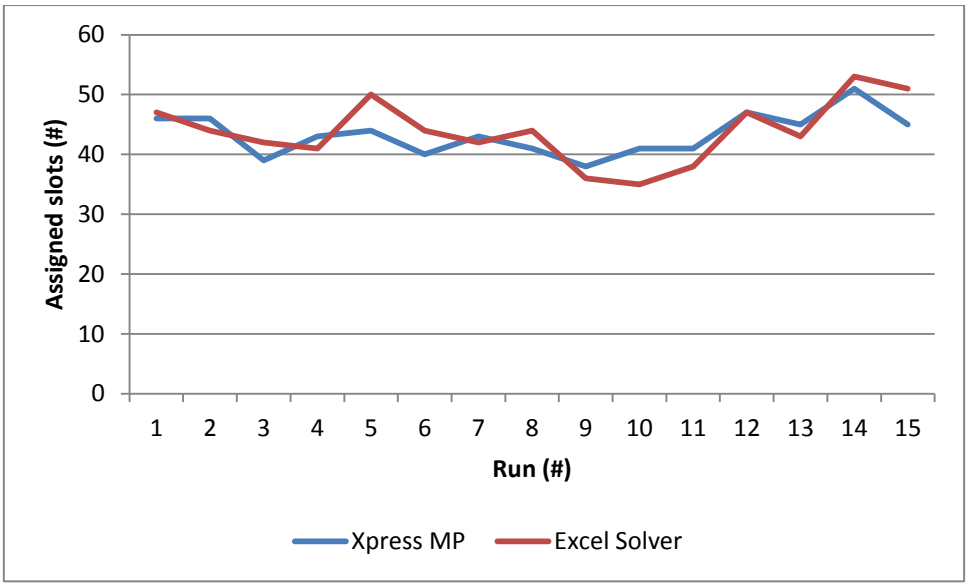

Figure 5.1: Slots assigned to New Patients

From this graph it can be concluded that not only in average number of slots, but also for each separate run the results come really close to each other.

In summary, the table and figure implicate that the optimization in Xpress MP gives satisfying results for the division of capacity and can therefore be used for larger datasets as well. Excel Solver was a good tool for testing this.

5.2.5 Conclusions

If capacity is given, a very good division of capacity can be determined using the Xpress MP optimization. This method is easily extendable and reliable.
For the situation in this project the optimization was carried out 15 times, optimizing over 6 months. A monthly capacity of 205 was used and the highest possible demand stream was chosen to put as most pressure as possible on the system. On average the number of slots that should be assigned to new patients turned out to be 43.33 slots. This is not a slot less than the Excel Solver, which optimizes on actual waiting times instead of number of patients, provides.
The current capacity should be, if the waiting times are to be minimized, divided into 44 slots for new patients and 161 for follow up patients. This is exactly how it is done at this moment, so nothing needs to be changed if the total capacity has to stay the same.

5.3 Queuing Model

In the previous paragraph it was analysed how to divide a given number of capacity slots over new patients and follow up patients. In this paragraph it will be looked at what if this division could be ignored and what happens if changing capacity is possible.

5.3.1 Model

In the chapter "Methods" an example of a queuing system was given. The situation in this project can be viewed in a similar way. In figure 5.2 it was shown how the queuing model mentioned earlier is applicable for this project.

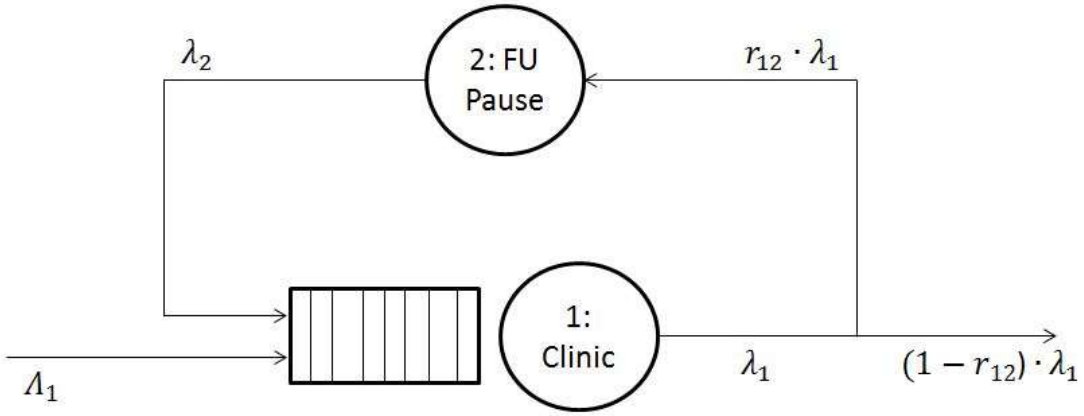

Figure 5.2: Queuing Model

The arrival process of new patients stays the same as discussed in the previous chapter for simulation usage. It is a Poisson Process with parameter Λ_1, because the arrivals can only be at node 1. Node 1 represents the clinic. After an appointment at the clinic, a patient can either come back for a follow up, with probability r_{12}, or be discharged. After some time in the follow up pause node, which works the same as earlier mentioned in simulation usage, all the patients go back to clinic. In contrast to the simulations carried out in the previous chapter, a capacity is now added. The parameter μ_1 represents the capacity at node 1. This means that a queue can now build before clinic, which is exactly what we are interested in.
This is a simplified model compared to what is discussed in the previous chapter. It can therefore only provide indications of waiting times in the queuing system. It is however a quick way to analyse what impact changing capacity would have.

5.3.2 Parameters

To be able to calculate the performance measures mentioned in the chapter "Methods", some parameters need to be determined. The first parameter needed is the arrival process parameter Λ_1. From the last chapter it can be concluded that 1.288 a day, so 38.64 a month, is the best value to use.
Secondly, the percentage of patients needing a follow up, needs to be determined. This percentage was derived from the historic data. It turned out that 69.5 percent of all the appointments were registered to need a second appointment. This means that the parameter r_{12} receives the number 0.695.
The third parameter, μ_1, is equal to the number of slots available. As mentioned earlier in the paragraph "Current Situation", the capacity is currently 205 slots a month. In the model discussed in the previous paragraph, DNAs are not taken into account. If it is desirable to take this into account it can be done in relation to the capacity instead of the demand. On average, the demand was 185 in the paragraph "Validation" of the previous chapter. This was lifted by 15 percent which would come down to almost 28 extra slots of demand. Instead of adding this to demand, it can now be subtracted from the capacity, which should have the same effect. The parameter μ_1 should now be set to 177 slots a month.

The fourth parameter, μ_2, represents the average time between two following appointments. It does not have any effect on what we are interested in, the queuing at node 1, but it was determined anyway. On average 129.7 days, so 4.32 months, are between two following appointments. This means the parameter μ_2 should be set to the opposite; 0.231.
With the parameters mentioned above the effective arrival rates can be calculated. With the arrival rates the performance measures for both nodes can be determined. Since in this project there will only be queuing at node 1, these are the only performance measures we are interested in. This means that both the service rate and the arrival rate of node 2 are irrelevant in this situation. Therefore only the performance measures of node 1 will be calculated and discussed. This will provide information about the queue before node 1.

5.3.3 Performance Measures

The parameters discussed in the previous paragraph were used to calculate the effective arrival rate and the performance measures of node 1. In the table below the results can be viewed.

Table 5.5: Queueing Theory 1

Parameters	
Λ_1	38,64
r_{12}	0,695
μ_1	177
Effective Arrival Rate	
λ_1	127
Performance Measures	
# in system	2,54
# in queue	1,822
time in system	0,02
time in queue	0,0144

The first thing to be noticed here is the effective arrival rate of 127 patients a month. This is a lot lower than the determined demand in the previous chapter of on average 185 patients a month. To make it more reasonable, either the arrival rate Λ_1 or the return rate r_{12} can be adjusted. The latter was chosen since the first one was completely analysed already. It was discovered that changing this rate r_{12} to the value of 0.791 resulted in an effective arrival rate of 185. The problem with queueing theory however, is that the service rate always needs to be higher than the arrival rate to be able to calculate (appropriate) performance measures. If we ignore the extra precautions to take DNAs into account again, then we are back to a capacity of 205 slots a month. In the table below it is shown what these changes do to the performance measures.

Table 5.6: Queueing Theory 2

Parameters	
Λ_1	38,64
r_{12}	0,791
μ_1	205
Effective Arrival Rate	
λ_1	185
Performance Measures	
# in system	9,25
# in queue	8,35
time in system	0,05
time in queue	0,0451

Because we are speaking of a simplified model, both sets of performance measures determined here are based on a lot of assumptions. Nevertheless it is interesting to see that according to the second set of parameters, the average time in the queue is 4.51 percent of a month, so more than a day. And this is even without taking weekends into account and all the available slots are evenly divided over a month. On top of that queueing theory works on first come first serve base, which means waiting times can hardly be more than a month in this scenario. In the table below it can be viewed that breaking it up into a daily process would make the waiting time in the queue even longer.

Table 5.7: Queueing Theory 3

Parameters	
Λ_1	1,288
r_{12}	0,791
μ_1	6,8
Effective Arrival Rate	
λ_1	6,16
Performance Measures	
# in system	9,63
# in queue	8,72
time in system	1,56
time in queue	1,42

It indicates that waiting times could be a lot higher in reality. The turning point for queueing theory will always be the effective arrival rate. The capacity can never be lower than that. The capacity seems to fulfil this restriction as long as the DNA assumption is not made.

5.3.4 Conclusion

Queueing theory can be used to provide an indication of the waiting time. For the situation in this project a simple version of the model was applied, which led to a number of performance measures. It is now possible to see what effect changing capacity would have. It is best to keep it relative since the indicating performance measures are estimated to be lower than in reality. The definite turning point for capacity is 127 slots according to the queueing model based on parameters gathered from the data. For our situation it can be concluded that the current capacity should be sufficient.

5.4 Simulation

Until now a demand was determined and it was analysed how the current capacity should be adjusted to this in terms of division between new patient slots and follow up patient slots. In this paragraph this division will be tested by means of simulation. After the optimization it was looked at the pressure on the system by means of queueing theory. This can provide information about what happens if the number of capacity slots was changed. To be able to also obtain more information about how the waiting times are dependent on how the capacity is divided over a week, a simulation with capacity restrictions was carried out. The capacity in this simulation was implemented as a weekly server schedule. This makes it possible to study the impact of dividing the capacity differently over a week. This simulation is based on the same simulation that was used in the previous chapter. The model needed to be changed only a little and some new features needed to be added to be able to implement the capacity in this specific way.

5.4.1 Model

Two models will be used to analyse waiting times. Firstly, the same simulation as in the previous chapter will be used in this chapter. Only now the capacity has become part of the model. This means that the clinic node (1) will not have infinitely many servers anymore. A weekly server schedule will be used for the capacity of node 1. This means that in contrast to the simulation to model demand, there will be a queue building at node 1. The capacity of the queue itself stays infinitely large. The model from the previous chapter, now including a queue before the clinic node, was shown in Figure 5.3 below.

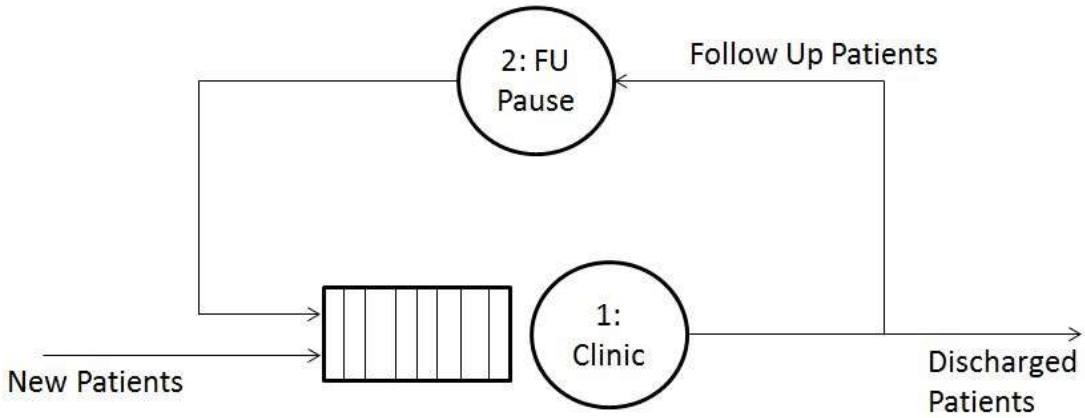

Figure 5.3a: Simulation Capacity Model 1

The second model allows for the division in capacity between new patients and follow up patients. An extra node was added to create an extra clinic. The capacity assigned by the optimization can now be implemented separately for new patients and follow up patients. Also the waiting times can be analysed separately in this model. The second model is shown in Figure 5.3b below.

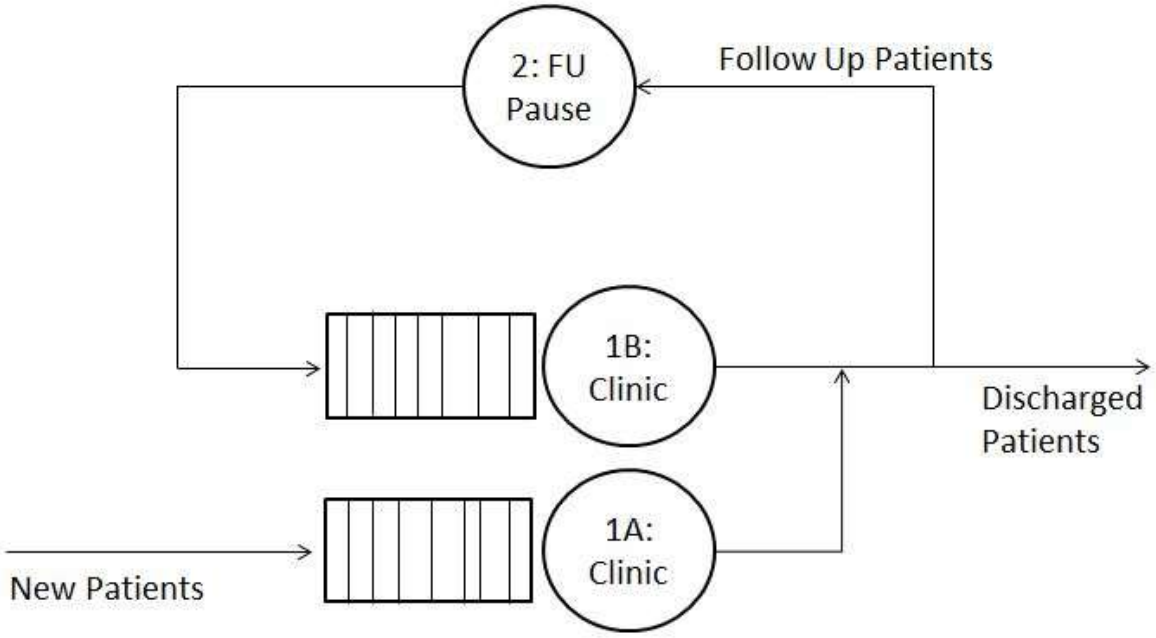

Figure 5.3b: Simulation Capacity Model 2

5.4.2 Goals

In this simulation the most interesting figure is waiting time for patients in the queue of the clinic node. The simulation can be used to test two elements of the system. The impact of allocating capacity slots to new patients and follow up patients as discussed in the optimization step can be tested. The simulation is also useful to test the outcome regarding waiting times for different hypothetical capacity schedules. The main aim is to develop a simulation that can be used to test how both division of slots between new patients and follow up patients and division of slots over time influence waiting times. For our specific situation the aim is to see what the waiting times look like using the current division of capacity over new patients, follow up patients and time. Based on this information it can be decided whether changes are necessary for this set of clinics or not.

5.4.3 Description of Methodology

To obtain more information about waiting times in different scenarios four simulations will be carried out. The first two simulations focus on the influence on waiting times of division of slots over time. The first one adds a weekly capacity server schedule (the current situation) to the model that can be viewed in Figure 5.3a. To compare the influence of the weekly server schedule, the second simulation divides the same amount of capacity evenly with 1 server over the week.

The last two simulations are meant to analyse the influence of division between new patients and follow up patients with. For these two simulations the model that can be viewed in Figure 5.3b was used. One simulation uses weekly server schedules for both clinic nodes, and the other uses evenly divided capacity over time again.

As mentioned before, not only changes to the model, but also some extra features to the simulation shell need to be added. In this paragraph the changes will be explained and the differently used parameters will be discussed.

5.4.3.1 Changes to Existing Simulation Shell

In the simulation shell that already existed, it was possible to implement capacity. This only consisted however of entering a number of servers for each node and service distributions and rates. This could not be changed over time. This means the clinic would always be working, also during weekends. This was changed by adding the possibility of server shifts. This means a cycle length can be chosen, for example 7 days to form a week. Then this cycle can be divided in shifts where 1 server works, multiple servers work or no servers at all.
This means shift changes need to be implemented in the simulation. Because we are working with exact slots of service for every patient, a shift change can never be in the middle of service of a patient. In general however, when service rates are not according to slots but more unpredictable, this could happen. It was therefore decided that at every shift change the whole shift would be changed. This means the old servers finish the service for the patient they are currently treating, and the new servers are already added. This means sometimes two shifts could work at the same time for a very short period, which would not be possible in some situations. In the situation of this problem however it does not cause any problems.

To explain how the changes to the simulation were done, we go back to the event structure of the simulation as discussed in the chapter "Methods". We start with the B events, linked to a certain point in time. Next to the two already existing B events, a new patient arriving or a patient finishing his service, a third B event needs to be added. This is the event where a server shift ends. If the next shift is to contain more than 0 servers, a C event follows. This C event is adding this server or those servers. If the queue at the regarding node is empty, nothing else happens. Otherwise, the next customer in line starts service at the new server. This C Event can happen multiple times at once if more than one server was added. In figure 5.4 this can be viewed the same way as B events (circles) and C events (squares) were represented in the chapter "Methods".

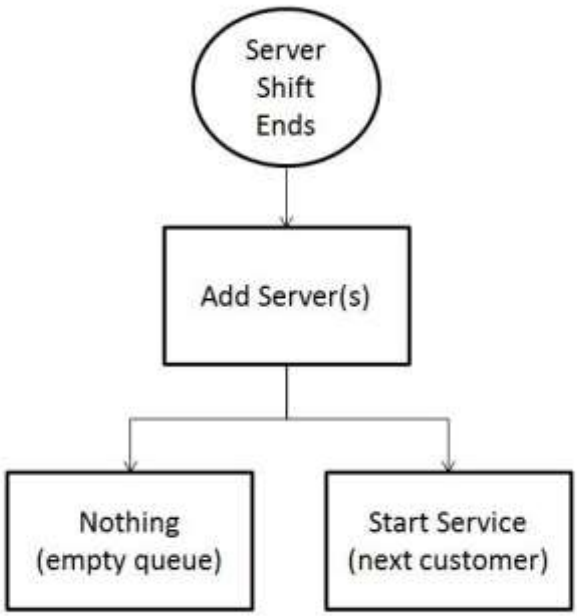

Figure 5.4: B event 3

Next to this new B event and its own conditional C events, there is also another new C event that needs to be added. This C event is a result of the B event when a patient finishes service.
When a server shift ends, the old servers are not immediately thrown out, but finish with helping the current patient and are then deleted. This means that deleting the servers will not be an event

occurring in time (B event), but a conditional event (C event) after a patient finishes service. The visual overview of this B event and all its C events, including the new C event, are shown in the figure below.

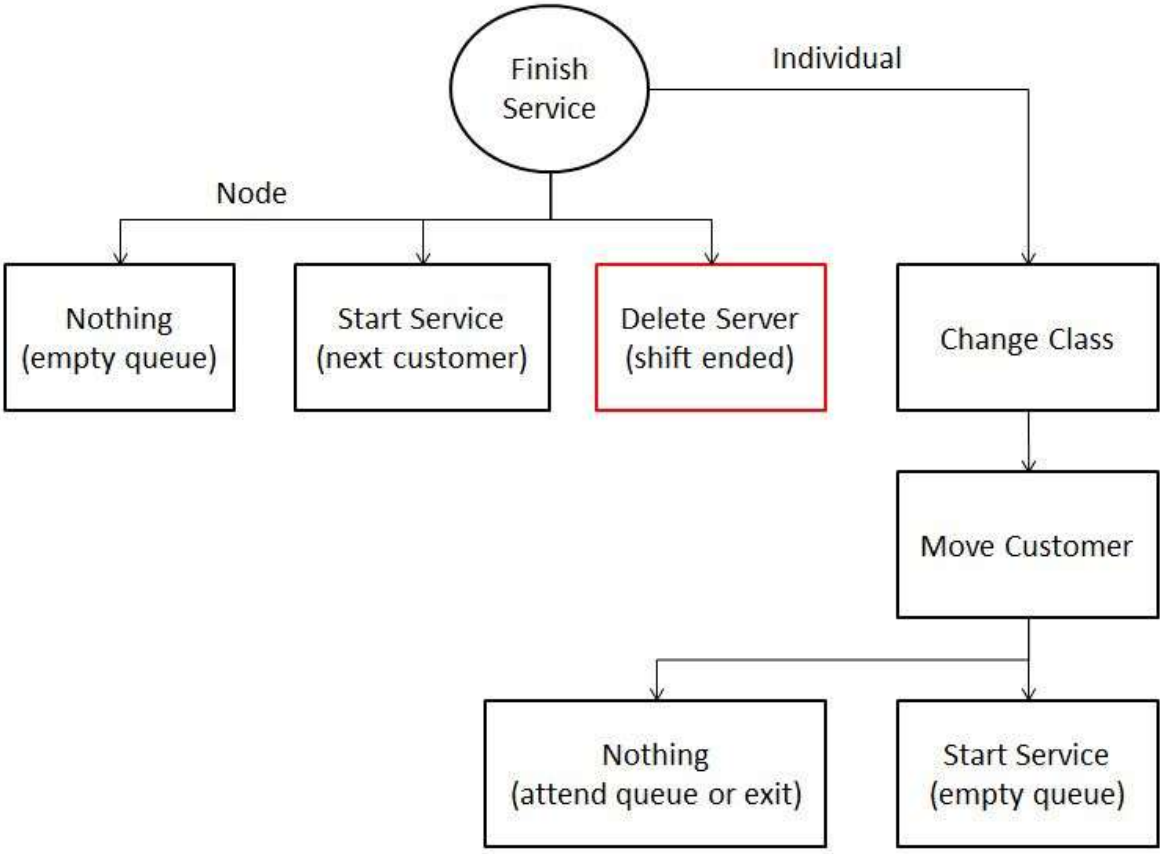

Figure 5.5: B event 2 changed

5.4.3.2 Input Parameter File

The parameters needed for all four simulations are explained separately in this paragraph.

5.4.3.2.1 Simulation 1

Compared to the old parameter file a few things need to be added and changed. A cycle length needs to be added, representing the after which the same server schedule will be repeated again. Then the schedule itself needs to be added, including the times relative to the cycle on which shift changes happen and the numbers of servers for all the shifts. Finally, the service rates at the clinic node (1) need to be changed. In the demand determination the service times were set very small, because we were only interested in the number of patients asking for an appointment. Now the service times need to be exactly a time slot.
There are many different ways how the parameters can be chosen to implement the situation. The important thing is that the three steps mentioned above need to be tuned to each other.
It was chosen that for now a weekly schedule is most interesting. This means the cycle length was set to **seven** days.
As discussed earlier in this chapter the capacity a month was believed to be 205 slots. Since the simulation cannot take DNAs into account as part of the demand, it was chosen to do this as part of the capacity. This was applied the same way as in the paragraph “Queueing Theory”, meaning that the on average almost 28 slots that was originally added to demand will now be subtracted from

capacity. The new capacity is 177 slots a month, which comes down to approximately **44** slots a week.
According to the data a slot takes 15 minutes, this means approximately **0.01** day. Therefore the service rates at the clinic were all set to deterministic with rate 0.01.
For the clinic 988/987 it was known that 15 slots were available on Mondays and 8 slots on Thursdays. Assuming this division is the same for the other two clinics; we can work with 29 slots on Mondays and 15 slots on Thursdays. Starting at 9.00 am in the morning on Monday, using only **1** server, would mean a shift from **0.36** until **0.65** of the day. If we take the week from Monday until Sunday this immediately corresponds with the first two shift change moments. On Thursday the other 15 slots were also started from 9.00 am in the morning. Relative to the cycle this means **1** server from **4.36** until **4.51**. This results in the following server schedule with 5 shifts.

Table 5.8: Server Schedule Node 1

Time in Cycle	Servers
0.00 – 0.36	0
0.36 – 0.65	1
0.65 – 4.36	0
4.36 – 4.51	1
4.51 – 7.00	0

Because of the increased pressure on the system by the capacity, it will take longer for the simulation to reach steady state. Therefore the simulation time was adjusted to **20,000** days for each run.

5.4.3.2.2 Simulation 2

To be able to analyse the influence of a server schedule, the same simulation as mentioned above was also carried out with evenly divided capacity over time. This means for this simulation we work with **1** server at the clinic node, which is active all the time. The second thing that needs to be changed in the parameters is the service times, because there still need to be only 44 patients that can be treated in a week. Therefore all service times at the clinic node were changed to **0.16**. This simulation was carried out for only **10,000** days.

5.4.3.2.3 Simulation 3

This simulation is a lot like the first simulation. Therefore we only explain the differences with Simulation 1. Compared to Simulation 1 the first thing that needs to be added is an extra node (1B). The former clinic node (1) is now split into two; 1A and 1B. All arrivals are at node 1A and all patients returning to clinic after a follow up pause go to node 1B. This can be added to the transition matrices for all classes.
The queue capacity, class changes and service rates per class are all identical for both node 1A and node 1B and are the same as for node 1 in Simulation 1. The only thing left that differs for both nodes is the server schedule. Out of the 44 slots a week 9 slots were allocated to new patients and the other 35 to follow up patients. This is respectively the same division as was determined for 205 slots (44 and 161). According to our information all new patient slots are currently slotted in on Mondays. Therefore the server schedule for node 1A only consists of a server from Monday morning 9.00am during 9 slots.

Table 5.9: Server Schedule Node 1A

Time in Cycle	Servers
0.00 – 0.36	0

0.36 – 0.45	1
0.45 – 7.00	0

The 20 remaining slots on Monday and the 15 slots on Thursday will be used for follow up patients. This means the server schedule for node 1B looks like the table below.

Table 5.10: Server Schedule Node 1B

Time in Cycle	Servers
0.00 – 0.36	0
0.36 – 0.56	1
0.56 – 4.36	0
4.36 – 4.51	1
4.51 – 7.00	0

Note that we are talking about three different clinics. That means it is no problem to schedule more than 1 slot at the same time, which is now happening for some new patient slots and follow up patient slots.

5.4.3.2.4 Simulation 4

This simulation works the same compared to Simulation 3 as Simulation 2 works compared to Simulation 1. This means the only thing that is different is that the capacity is evenly divided over time. At both clinic nodes **1** server is used, which is active all the time. The service times for new patients and follow up patients are adjusted separately. For new patients, this means at clinic 1A, the service times were changed to **0.78**, this means 9 patients can be treated in 1 week, just like in the simulation above. At clinic 1B, for follow up patients, the service times were changed to **0.2**, which means 35 patients can be treated weekly. Again, the simulation was carried out for only **10,000** days.

5.4.4 Analytical verification

Most of the analytical verification of our model has already been done in chapter 4, paragraph 4.3.5. This is because the model we work with in this chapter is very similar. Therefore in this paragraph only the addition to the model of the demand simulation and the extra added features to the code will be tested.
To test our model we first check whether we can see that only new patients attend clinic 1A and only follow up patients attend clinic 1B. This was confirmed by means of analysing test runs. Also the waiting times compared to arrival times and starting service times were tested and confirmed to provide correct information.
The last of the four simulations in ASQ, with evenly divided capacity, could be compared to the Simul8 model, which provided an extra visual view. The results of average waiting times are shown below.

Table 5.11: Waiting Times Comparison (days)

	ASQ	Simul8
Average Waiting Time New Patients	53.7	51.2

The results are very similar, which leads to the conclusion that the ASQ model does what we want it to do.

Finally the new feature of adding a server schedule was tested. A few single runs were analysed and it was confirmed that no patients were treated at times when no servers should be active. Also, it

was confirmed that, at each clinic node, there was never more than one patient treated at the same time.

5.4.5 Validation

In this paragraph it was meant to analyse whether our simulation provides realistic results. Unfortunately, there was no data available about waiting times, which makes it harder to determine what is realistic and what is not. Therefore we compare with expectations of reality instead of actual reality.

First of all, as was expected to be the case in reality, the capacity forms a restriction on the situation. Therefore waiting times occur. For our simulations this also means the warm up time is longer than before. It takes more time for the system to fill up and reach steady state.

From all simulations can be concluded that the waiting times are a lot higher for new patients than for follow up patients. This seems to be in line with reality, as this could have been the reason for the government to set targets on maximum waiting times for new patients.

In our model we made the assumption that new patients and follow up patients are treated in different capacity slots. According to information obtained about the specific clinics this is what happens in reality. We also obtained a little information about how the slots were divided over time. The assumption that this information would hold for all three clinics we made ourselves. We can say it is realistic however because we talk about clinics which can treat mostly the same conditions. This means they need comparable equipment and clinicians and therefore there is a good chance those clinics work very similar.

The ending of server shifts was implemented as if every server would stay active until the first moment in time a patient finishes service while the shift has already ended. This means that a clinician would always at least work as long as scheduled. In our situation this assumption has almost no influence, since we work with fixed service times, the slots. Therefore in our situation this is a realistic assumption. In other situations this might be a wrong assumption.

5.4.6 Numerical results

Just as in the previous chapter the results of 85 runs for each of the four simulations were gathered. Those results regard average waiting times for both new patients and follow up patients. The average waiting times were calculated every month. It should be noted that the monthly demand of follow up patients is sufficiently higher than the monthly demand of new patients. In the diagrams below the average waiting times for new patients for all four simulations can be viewed.

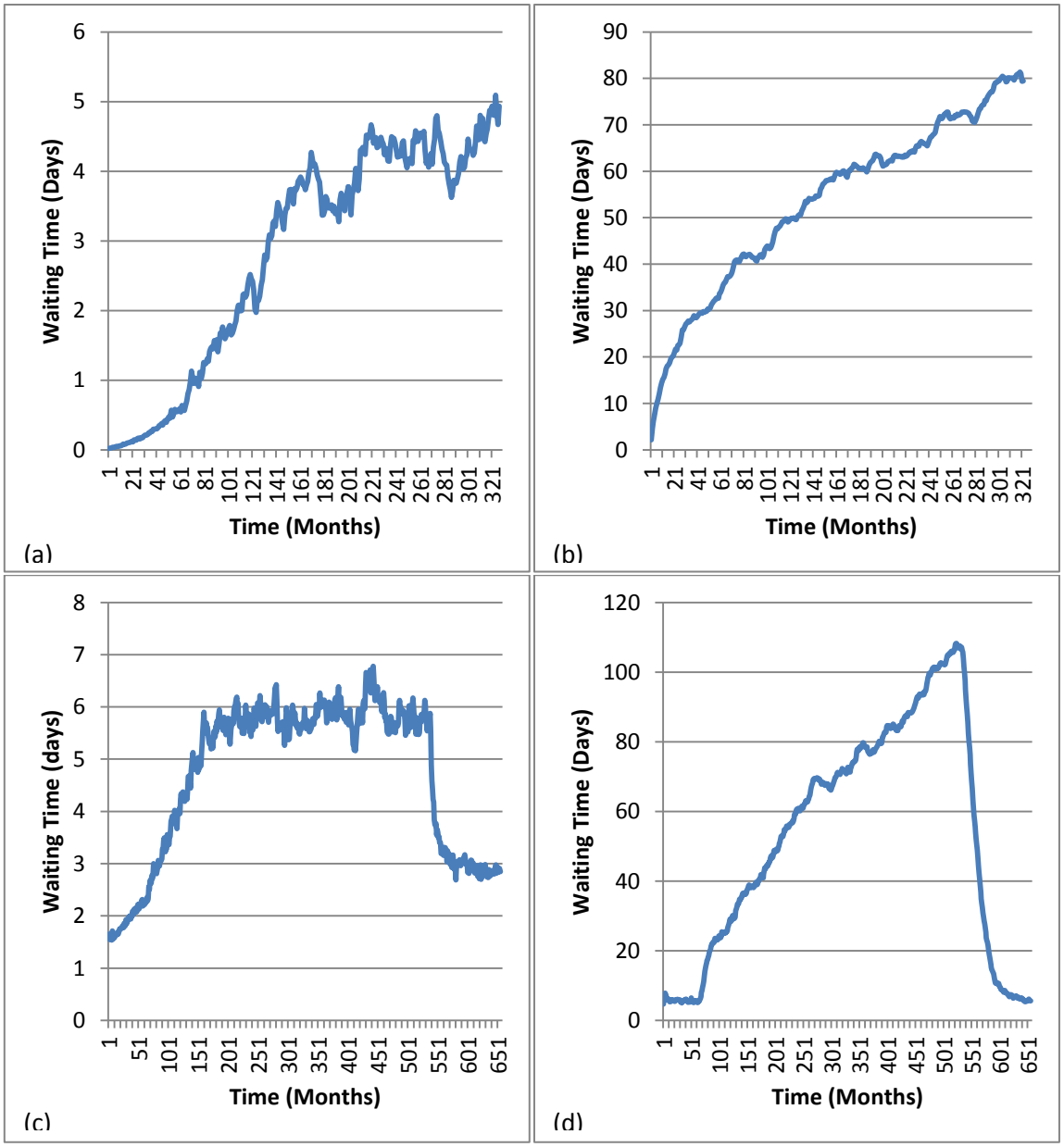

Figure 5.6 Average Monthly Waiting Times New Patients (a) 1 Clinic Node, No Schedule (b) 2 Clinic Nodes, No Schedule (c) 1 Clinic Node, Server Schedule (d) 2 Clinic Nodes, Server Schedule

The corresponding average waiting times for follow up patients are shown in Figure 5.7.

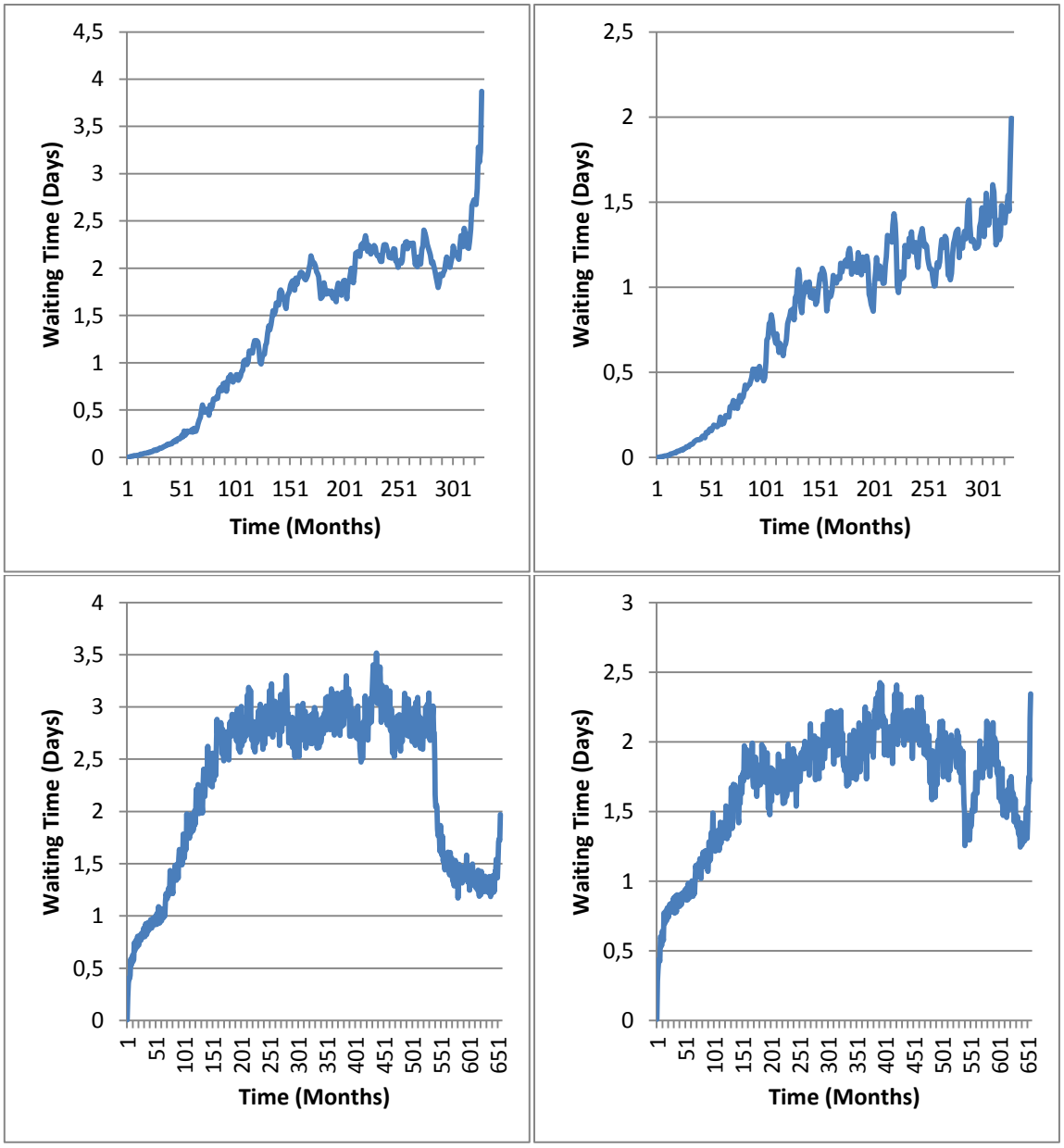

Figure 5.7 Average Monthly Waiting Times Follow Up Patients (a) 1 Clinic Node, No Schedule (b) 2 Clinic Nodes, No Schedule (c) 1 Clinic Node, Server Schedule (d) 2 Clinic Nodes, Server Schedule

5.4.7 Conclusion

In this paragraph we will discuss the graphs shown in the previous paragraph and draw conclusions from that. We start with comparing the outcomes of the different simulations. First we look at the influence of a server schedule. This means we compare the graphs (a) and (b) to (c) and (d) of the Figure 5.6 and 5.7 for new patients and follow up patients respectively. It can be concluded that using a server schedule instead of dividing capacity evenly over time results in longer waiting times. For follow up patients in particular the increase in waiting times is quite high in the beginning with a server schedule, but in the end the difference with no server schedule is not that large.

Now we compare the graphs (a) and (c) to (b) and (d) of the Figure 5.6 and 5.7 for new patients and follow up patients respectively. This means we compare dividing slots between new patients and follow up patients or not. For new patients the waiting times are immensely higher if the slots are

divided. For follow up patients the waiting times turn out to be even smaller if slots are assigned in advance. Apparently the current division of slots is to the advantage of follow up patients. Earlier it was concluded that in theory the system has enough capacity to work with. In the left graphs all waiting times seem to be acceptable, which means this is in line with the statement of sufficient capacity. In the right graphs we can see however, that dividing the slots, even though we use what should be the best division, can result in undesirably high waiting times.

We now compare the new patient's graphs with the follow up patient's graphs for all the individual simulations. First we look at the graphs of Figure 5.6(a) and 5.7(a). In fact this is the ideal situation, where capacity can be divided evenly over time and slots do not need to be divided between new patients and follow up patients in advance. All resulting waiting times are very acceptable and it seems therefore that the capacity is sufficient. The graph for new patients increases faster in the beginning, but that was expected since there are a lot more new patients in the system in the beginning.

In the (b) graphs of Figure 5.6 and 5.7 new patients and follow up patients were assigned separate slots to use. The chosen division of demand results in very high waiting times for new patients but not for follow up patients. This seems to be very unfair.

No division of slots but using a weekly server schedule results in the graphs (c) of Figure 5.6 and 5.7. Both graphs show a very similar pattern, which seems to reach steady state but then drops a little again. An explanation could be that the average waiting times are oscillating until they reach a steady state where the average waiting times for new patients and follow up patients are approximately equal.

The graphs (d) of Figure 5.6 and 5.7 represent the situation with the most constraints. The capacity is divided between new patients and follow up patients and uses a server schedule. The waiting times for new patients were expected to be higher because all slots for them are on 1 day during the week. Still the difference turned out to be surprisingly large. At the end the average waiting times seem to drop again dramatically. Since the demand for new patients on the system is fairly constant, this is a very surprising result. An explanation could be that the queue is back to zero and the system will build up again after that.

It can be concluded that mainly the division of slots between new patients and follow up patients puts a lot of pressure on the system. Even though the optimal division of slots is used, resulting waiting times for new patients are unacceptably high. This is because it was optimized on total waiting times, and the demand of new patients is a lot lower than the demand of follow up patients. Therefore, even though in theory the capacity should be sufficient, experimenting with some extra capacity could provide rewarding improvements.
Although it is less of a problem, the server schedule can also increase the pressure on the system. Therefore it could be rewarding to experiment with this as well. It can also be used to specifically decrease waiting times for new patients by making their server schedule more evenly divided over a week.

5.5 Summary

In this chapter it was first looked at how to divide a given capacity between new patients and follow up patients. It was concluded that the extendable optimization in Xpress MP provides accurate results for this. For our situation it was concluded that the current division of capacity should be optimal.
By means of queueing theory the pressure on the system could be analysed. It was concluded that for our situation the current capacity should be sufficient.

Finally a set of four simulations was carried out. Together they could provide us with the influence of dividing capacity both over time (which means adding a server schedule) and between new patients and follow up patients.
For our situation it was concluded that the waiting times for new patients are very high. This is in line with what was expected to be happening in reality. Even though the optimal allocation of capacity according to the optimization was used, this does not mean that the optimization is not correct. The optimization minimized total waiting times, only it turns out that it in our situation it results in a very unfair distribution, because the demand of new patients is much lower than the demand of follow up patients.
To be able to obtain a better distribution of waiting times for our situation a few things can be done. It can be chosen to divide waiting times fairer by using a different allocation of slots, but it has to be accepted that total waiting time will increase. Another possibility is to increase capacity. The effect of this can be analysed by the simulations. The last method of decreasing waiting times for new patients and therefore restoring the balance a little is changing the weekly capacity server schedule. Again the effects can be analysed by using the simulations.

6. Conclusion

The strategy which best optimizes the scheduling for outpatient clinics can be found in at most four steps. The first step is finding the best forecasting technique to use to determine future demand with. If historic data provides sufficient information about historic demand, the second step can be skipped. Otherwise the second step needs to be taken. It was concluded that a simulation, based on follow up structure rather than historic demand, can provide an appropriate picture of the real demand on the system. When the demand is determined, the third step is looking at how the capacity slots can best be divided between new patients and follow up patients. It was concluded that a developed optimization in Xpress MP provides an accurate answer on how to divide capacity, aiming for minimal total wait. The last step involves studying the impact on waiting times of the division of capacity over time and between new patients and follow up patients. The simulation mentioned earlier was adjusted to make this possible. After those four steps a conclusion can be drawn about how best to schedule in terms of capacity size and division of capacity over time and between different types of patients.

The four steps mentioned above were applied to a set of three clinics from the Ophthalmology Department of the University Hospital of Wales. It was concluded that Holt's Linear Exponential Smoothing is the best forecasting method to use on the registered past activity. It was also concluded that this past registered activity does not provide an appropriate picture of historic demand. Therefore, by means of the simulation, a stream of demand was generated. This was used in the optimization, which resulted in the conclusion that the capacity should be divided into 44 slots for new patients and 161 for follow up patients every month. This is exactly what was believed to be happening at the moment. So this would imply that no changes are necessary in that area. However, the optimization provides the best division of slots, aiming for minimal total wait. In our situation we deal with a lot more follow up patients than new patients. Using the simulation of the fourth step it was discovered that this resulted in much higher average waiting times for new patients than for follow up patients. If it is aimed for a fairer distribution of waiting times, slots should be allocated differently or the total capacity should be increased.
Finally the influence on waiting times of the current weekly capacity schedule was analysed. Although this influence was determined to be less in our situation than that of allocating slots, it was concluded that is still worth experimenting with. At least the influence of every suggested change can be checked by using the simulation.

In general the strategy which best optimizes the scheduling for outpatient clinics can be found using the four steps discussed above. Specific attention was paid to the first of the two research sub questions. The other two can also be analysed using the developed tools, but due to limited time it has not been done during this project. By looking at the answers to the first two research questions, the conclusion in our specific situation is that it should be considered to either increase capacity or allocate capacity differently, depending on what is determined as the "best optimization" by the Cardiff and Vale University Health Board.

7. Discussion

It is hard to develop a generic method for finding the strategy that best optimizes scheduling for outpatient clinics, because it can be different for every clinic. It will always be historic data dependent how well this can be done. On the other hand, if it is looked at the methods provided in this project, it can be based on that beforehand what information should be stored. This will make it easier to optimize scheduling.

The project was started with looking at forecasting methods. At the time it seemed a logical step since it was not concluded yet that registered activity was not the same as the demand and that a potential problem could not be fixed by using target dates. Looking back this forecasting step could have been skipped for our specific situation. However, in other situations where capacity has not been limiting activity in the past, this can still be a useful step.

The optimizations done for our situation were done monthly. This was decided because 6 months was the time that was recommended to optimize over. To optimize on more precise waiting times however it would be better to optimize over weeks. With a small adjustment to the recording of information in the simulation, the demand can be determined a week instead of a month. The optimization in Xpress MP can then easily be done in weeks.

The most important thing both the demand and capacity simulation rely upon is the follow up structure. In our situation we had to determine this follow up structure purely based on historic data. Although nuances were taken into account, it would be better to speak to clinicians about this as well. A list of conditions treated by the regarding clinic should be considered and average times until the next follow up discussed for each condition. This could provide an even more accurate impression of the follow up structure.

The second to last research sub question is about no-shows. Unfortunately, due to time limitations this project could not properly cover this topic. In the simulation of demand it also had to be dealt with the influence of “Did Not Attendees”. The influence was very roughly added to the model by comparing a few crude scenarios. It would have been better to make a more precise analysis of the situation of Did Not Attendees. The final simulation could also have provided interesting insights about the influence of non-attendances.

The same as for the second to last research sub question holds for the last one. Unfortunately it was not covered in this project. A little robustness was discussed however in chapter 4. It was concluded that the actual demand was a lot steadier than the registered activity suggests. This could mean that this research sub question is therefore less interesting than was initially expected.

The capacity simulation works with generating requests which then enter the queue before clinic. This means the waiting time is equal to the time between the moment the appointment should take place and the moment the appointment actually can take place. This is exactly what we were interested in. The downside is that a slot is only assigned to an appointment the moment the starting time of the appointment slot is reached. In reality an appointment needs to be made in advance, because naturally a patient needs to know when to come in.

8. Recommendations

This chapter was divided into application of the developed methods and further research. First the recommendations for the University Hospital of Wales will be discussed. Secondly it is discussed what the next steps would be to improve the methods provided with this project.

8.1 Application

For the University Hospital of Wales the provided methods are immediately applicable. For the three clinics on which were focused even the extra simulation model in Simul8 is applicable. We would recommend using this extra method for her advantages compared to the ASQ simulation. In Simul8 the forecasted values for the demand of new patients can be used as input instead of a Poisson Process. This might provide more accurate results. Also, prioritizing queues can easily be implemented. This means it can be studied what impact prioritizing new patients over follow up patients in clinic, or the other way around, would have on waiting times.

We would recommend the Cardiff and Vale University Health Board to start with using the conclusions of this project. This means, depending on what is decided to be called “best”, increasing total capacity, allocating capacity slots differently or both. The influence on waiting times of a possible decision can be tested using the developed capacity simulation.

Furthermore we would recommend the Cardiff and Vale University Health Board to apply the simulation to determine actual demand on all departments. After this the optimization can be used as well, but not only to divide slots over new patients and follow ups, but also to divide slots of a whole department over different clinics. We would suggest applying the optimization on different time scales. This means not just on a monthly basis but for example also on weekly and yearly basis. Annual leave of clinicians for example can this way be taken into account. The capacity simulation that follows can also be done with a yearly cycle length, additional to the weekly one. This allows for studying the best way of dividing clinicians’ annual leave and trainings over the year.

8.2 Further Research

The next step to improve the methods will be making better standard use of the capacity simulation. There are a lot of possibilities to test all kind of different scenarios. Creating a standard method of evaluating the influences of the following factors would be recommended:

- Decreasing no-shows
- Unexpected change in demand
- Allowance of a longer time until a follow up appointment (for example 1 week or 10 percent of the original time)

Evaluating these factors could improve the advice on what strategy best optimizes the scheduling for outpatient clinics. In this project “best” was defined as the shortest total waiting time for patients. For indirect waiting times it can also be relevant to look at the fairness of waiting time. As discussed in the chapter “Literature”, according to Gupta and Denton (3), this can be applied amongst others by setting a maximum to the waiting time, or minimizing the variance. These two suggestions could be a valuable extension to the model created in this project.

9. References

1. Pan, Chong, et al. "Patient flow improvement for an ophthalmic specialist outpatient clinic with aid of discrete event simulation and design of experiment." *Health care management science* 18.2 (2015): 137-155.

2. Dechartres, Agnès, Valérie Mazeau, Catherine Grenier-Sennelier, Antoine P. Brézin, en Gwenaelle M. Vidal-Trecan. "Improving the Organization of Consultation Departments in University Hospitals." *Journal of Evaluation in Clinical Practice* 13, nr. 6 (1 december 2007): 930–34. doi:10.1111/j.1365-2753.2006.00785.x.

3. Gupta, Diwakar, and Brian Denton. "Appointment scheduling in health care: Challenges and opportunities." *IIE transactions* 40.9 (2008): 800-819.

4. Cayirli, Tugba, and Emre Veral. " Outpatient Scheduling in Health Care: A Review of Literature ." *Production and Operations Management* 12.4 (2003): 519.

5. De Vuyst, Stijn, Herwig Bruneel, and Dieter Fiems. "Computationally efficient evaluation of appointment schedules in health care." *European Journal of Operational Research* 237.3 (2014): 1142-1154.

6. Harper, Paul Robert, and H. M. Gamlin. "Reduced outpatient waiting times with improved appointment scheduling: a simulation modelling approach." *Or Spectrum* 25.2 (2003): 207-222.

7. Syi Su, Chung-Liang Shih. "Managing a mixed-registration-type appointment system in outpatient clinics" *International Journal of Medical Informatics* (2003) 70, 31-40..

8. Zonderland, Maartje E., Boucherie, R.J., Litvak, N., and Vleggeert-Lankamp, C.L.. "Planning and scheduling of semi-urgent surgeries." *Health Care Management Science* 13.3 (2010): 256-267.

9. Potamitis, T., et al. "Non-attendance at ophthalmology outpatient clinics." *Journal of the Royal Society of Medicine* 87.10 (1994): 591-593.

10. Braaksma, A., Van De Vrugt, M., Boucherie R.J.. "Online appointment scheduling: a taxonomy and review" *Elsevier Editorial System (tm) for European Journal of Operational Research Manuscript Draft*. Received on 12 October 2015.

11. Bunday, Brian D. *An introduction to queueing theory*. Hodder Arnold, 1996. P155

12. Nelson, Randolph. *Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling.* Springer Science & Business Media, 2013. Paragraph 10.3.5 p455

13. Robinson, Stewart. *Simulation: the practice of model development and use*. Palgrave Macmillan, 2014.