Improving CT scan scheduling and queues with non-overtaking

Tjarko de Vree

February 2010

IBIS UvA
Instituut voor Bedrijfs- en Industriële Statistiek

Improving CT scan scheduling and queues with non-overtaking

Mathematical Sciences
Utrecht University and IBIS UvA
Tjarko de Vree
t.devree@uva.nl

Supervisors:
Drs. B.P.H. Kemper, IBIS UvA, Amsterdam, The Netherlands
Dr. E. Belitser, Utrecht University, Utrecht, The Netherlands
Prof. dr. R.J.M.M. Does, IBIS UvA, Amsterdam, The Netherlands

6th February 2010

Contents

List of Figures

Preface

You are about to read my Master's thesis about scheduling patients for a CT scan examination and the introduction of a new queuing theory property. In the first part of the thesis, an improvement project at Deventer Hospital at the CT scan area is extended with suggestions for a new appointment schedule. This study aims at the improvement of the appointment scheduling to facilitate more examinations while keeping performance very acceptable. This first research gave rise to a new queue property worth investigating in the second part of the thesis.

In the first year of my Masters, I followed the course Industrial Statistics at the University of Amsterdam. Given by the PhD student drs. Benjamin Kemper, this course became one of the courses I enjoyed the most during my time at Utrecht University. The course discussed the improvement methodology Six Sigma, a method which uses statistical ways to improve processes. For the first time, I was shown the possibilities of my mathematical knowledge in business. The interactive way of teaching was enlightening; in which other course did you fold paper helicopters to determine influence factors to derive an optimal design? Real life problems seemed to be completely different from the carefully fabricated ones used in other courses. I was therefore quite pleased that at the end of the course, Benjamin suggested a possible internship at IBIS UvA which facilitated this course. A year passed and after finishing my courses, there was no question left: I wanted to write my thesis at IBIS UvA.

Professor Ronald Does, the director of IBIS UvA, gave me the opportunity to write my thesis in Amsterdam for which I am very grateful. I asked dr. Eduard Belitser to be my supervisor in Utrecht. At first, it was not quite clear in which direction this thesis would go and I thank Eduard Belitser for giving me the freedom to follow my own path even though he would have loved to see a more statistical point of view. Benjamin Kemper became my supervisor at IBIS UvA and I especially want to thank him. His help, guidance and motivation made me possible to finish my Master.

There are more people I would like to thank. Jeroen de Mast, the one who did not mind to share his room with me, as far as I know anyhow. I enjoyed his sense of humor and his far reaching knowledge. He really opened my eyes about several topics and I am thankful to have learned a lot from him. I would also like to thank Marit Schoonhoven, Tashi Erdmann and Atie Buisman, from IBIS UvA, for welcoming me with open arms and making this time in Amsterdam very pleasant. I want to thank Yohan van der Bijl, Black Belt at Deventer Hospital, for allowing me to have a look at the radiology department and to gather the data needed for this thesis. I would like to thank my parents and Marianne for their helpful comments and support.

Amsterdam, January 2010, Tjarko de Vree.

Introduction

Hospitals are forced to make its processes more effective and efficient [28]. To provide the staff with the knowhow to undertake efficiency projects, Deventer Hospital in the Netherlands has launched a Lean Six Sigma program supported by IBIS UvA. During this program, employees are educated in quality improvement methodologies Six Sigma and Lean thinking. They also implement an improvement project submitted by the hospital. One of these projects investigates the routine at a CT scanner. It is this project that will be of great interest for this thesis.

Quality Improvement methodologies are used to improve the performance of processes. The CT scan project improved the CT scan process drastically which enabled the department to increase the examined number of patients per day. Sometimes, using only these methods, is not enough. Since the current appointment schedule (the planned appointment times for the patients) results in much idle time for the CT scanner, a new appointment schedule has to be made. How should this new appointment schedule look like? And how should we treat different patient types? In the thesis, queuing theory was used to capture the CT scan process. Unfortunately, the process was too complex to derive a sound model. Several characteristics seemed too difficult to incorporate in the queuing theory model. Also, extensive assumptions, which are not desired, are needed to make the model analytically doable. Therefore, a computer simulation method is chosen to calculate the performance measures for different schedules.

The thesis consists of two parts. In the first part of the thesis, a simulation study is started to find a better appointment schedule to schedule patients at the CT scanner. Several schedules are simulated to determine their performance.

While trying to capture the CT scan routine as a queuing system, an interesting property was found. A patient who arrives first in a dressing room is also examined first. For a next patient, it should not be possible to overtake this previous patient in the dressing room. However, overtaking in a multi server system with a first-come-first-served (FCFS) queue is possible. In the second part, the effect of the non-overtaking property for several multiple server systems is investigated.

The thesis can be described by Figure 1. In both parts of the thesis, a problem is formulated which may be solved by using queuing theory techniques. Since the problems are very complex, simulations are necessary to obtain solutions. For the CT scan case, these simulations are used to determine improvement actions. For the queuing theory property, performance measures are estimated by using simulations.

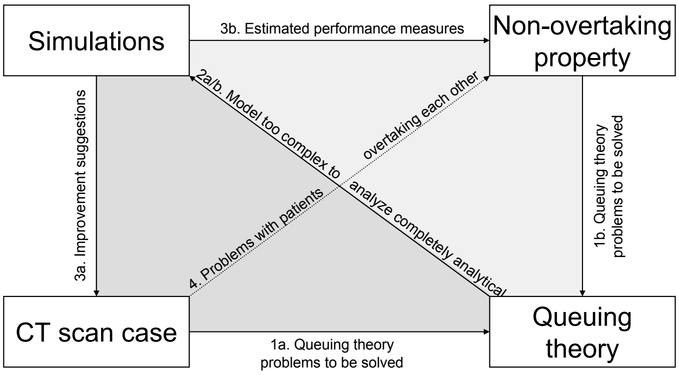

Simulations
3b. Estimated performance measures
Non-overtaking property
2a/b. Model too complex to analyze completely analytical
overtaking each other
4. Problems with patients
3a. Improvement suggestions
1b. Queuing theory problems to be solved
CT scan case
1a. Queuing theory problems to be solved
Queuing theory

Figure 1: Graphical description of reasoning behind the thesis

Part I

Improving patient flow in the CT scan routine

Introduction of Part I

In these times of economic decline and the announced reduction of 20% on government expenses on health care, efficiency starts to play a more profound role [18]. Improvements in efficiency, controllability of expenses, and income are the main focus in the following years. The CT scan area at the radiology department is typically of interest for improvement projects because of its high expenses and often its lack of efficiency.

The radiology department at Deventer Hospital has invested 1.227 million euros in 2008 in the CT scan area [38]. Capital intensive processes such as the CT scan, are found to be the bottleneck in many health care processes (from patient arrival to discharge) resulting in long access times. Therefore one of the projects at Deventer Hospital focused on the radiology department. Currently the radiology department does more than 7700 CT scans a year. The number of patients being scanned on a day, however, can be improved by using a quality improvement methodology. Being able to perform more examinations increases revenue.
Using discrete event simulations for this process, adjustments in procedures and appointment schedules can be simulated. *The main interest is finding an improved appointment schedule to be able to examine more patients on a day while maintaining appropriate performance.*

Three types of patients are distinguished in the simulation study (namely out-patients, in-patients, and urgent patients). Patients who have an appointment time scheduled in advance and are not hospitalized are called *out-patients.* Hospitalized patients who are also scheduled in advance are called *in-patients.* These are patients come from other departments. *Urgent patients* are patients who arrive unscheduled. The appointment schedule has to be designed in such a way that arriving urgent patients are examined within at most two hours after their arrival.

In the as-is process, scheduled patients arrive at the check-in desk and take place in a waiting room (WR). When the previous patient has almost finished the procedures in the scan room (SR), the next patient is forwarded to the dressing room (DR). Undressed, the patient proceeds to the SR. An intravenous (IV) access line is installed on the patient in the SR when contrast fluids (also abbreviated with contrast further on) are part of the examination. When arrived in the SR, the patient takes place on the scanner. A scout (prescan) is made to determine the area to scan. This is needed since any patient is positioned differently in the scanner. The scout is used to focus on the part of the body needed to be scanned. When the patient needs contrast (IV-patient), this is admitted after the scout by a pump. The next step is the actual scan. A next scans and a direct diagnosis follows if necessary (this direct diagnosis is now only done for some in-patients because a specialist needs to be called to give his or her judgment).
When contrast is part of the patients examination, a small recovery time is necessary to remove the IV access line. After completing this scanning routine, the

patient returns to his[1] DR. When dressed, he leaves the radiology department.

With the use of the Lean Six Sigma methodology, the process was improved and the need for a new appointment schedule became clear. With this process knowledge, a simulation study is started to determine improvement actions. In chapter 1, a literature review is given on CT scan process improvement articles and the use of simulations to model health care processes.

With the simulation program developed for the thesis, proposed schedules can be simulated to estimate their performance. The assumption document (containing the information for the simulation project) is given in chapter 2. Two schedules were suggested by Deventer Hospital, both seemed to be able to examine much more patients and to keep waiting times within reasonable bounds. With the use of literature about appointment scheduling, the advantages and disadvantages of several appointment methods are discussed in chapter 3. The results of the simulations are shown in chapter 4. The proposed schedules could be improved further more. Simulating several 'what if' scenarios (more urgent patients, longer scan times) other possible improvements can be found and performance in different situations can be estimated. In chapter 5, the results are discussed and improvements actions are suggested. In chapter 6, the quality improvement methodology Lean Six Sigma, as used by Deventer Hospital, is explained. Besides another methodology, Theory of Constraints, is introduced, as well as its additional suggested improvements on the CT scan process.

[1] From now on, when his or he is used, it is also possible to read respectively her or she.

Chapter 1

Literature overview

The CT scan area has been widely studied in improvement projects, see [3], [9], [32], and [34]. One common improvement action found is to remove certain procedures from the SR. Many examinations need to be conducted with IV-patients. Reallocating the insertion of the intravenous access line to a preparation room (PR), resulted in a decrease in access time from 21 days to less than 5 days at the Academic Medical Center (AMC) in Amsterdam, the Netherlands [9]. One substantial difference with the CT scan area at Deventer Hospital is the availability of three CT scanners where one is reserved for urgent patients. Elkhuizen et al. [9] stress the need to reduce the variability in the lead times. The time interval reserved for one appointment is determined by the mean examination time plus some slack time. This slack time is needed to compensate for the variability in examination times. Reducing the variability of an examination reduces the needed slack time and therefore smaller time intervals are applicable. Although this project resulted in a tremendous reduction in access time for scheduled patients, it did not make use of industrial statistics. The improvement actions were suggested in brainstorm sessions and data was not gathered to support these ideas. Only the access time before and after the project was measured, and indeed the reallocating of installing the intravenous line resulted in a higher possible capacity. Waiting times and other performance measures for the improved process were hard to estimate in advance. Vermeulen et al. [40] started a new project to determine the improved schedules. Increasing the capacity will not be sufficient, since being able to treat more patients in a day is of no use if the patients are not scheduled accordingly. After the improvement project done by Elkhuizen, scheduling was still done manually in cooperation with medical experts and based on experience and future expectations. A new method for scheduling patients is needed to make full use of this increased capacity [40].

Rhea et al. in [34] have a more cost based focus on improving the capacity at the CT scan area. Since this article is dated from 1994 suggested improvement actions might not be relevant anymore. Where a head CT took about 27.1 minutes in total, today less than 10 minutes are needed. Literature before the beginning of this century might therefore not be applicable anymore. Also digital image archiving as is discussed in [32] is now commonly used. Although most of the improvement actions discussed in these two articles have already been implemented, they show one common type of improvement: remove process steps form the SR.

Reinus et al. [33] use queuing theory to model the CT scan routine. They are interested in the steady state (the situation after some time when start up effect have worn out) waiting times. In reality steady state will not be reached. A working day is too short, and simply not enough patients are examined to make sure steady state is reached. They do conclude that when multiple scanners are available, scanners should be allocated to certain examinations only. Also the more difficult examinations, with more variability and higher mean examination times should be planned later on the day.

Boland [3] gives an overview on ways to improve CT scan capacity. These suggestions were collected from earlier research. Boland suggests that capacity can be increased by two options: by scanning more patients per hour or by scanning for more hours. Before deciding to proceed with an improvement action, the process has to be made visible carefully. Using flowcharting, the workflow can be outlined from start to end. Then, an operation management team can evaluate areas of opportunity to increase CT capacity. Proposed directions to search for improvements are adding technologists, reframe the workflow process or make use of RIS (Radiology Information System) and PACS (Picture Archiving and Communication System). When there are multiple technologists available per CT scanner, the flow of a patient through the CT scan process has shown to be much faster in comparison with using one technologist only. Especially when it is possible to install the intravenous access line outside the SR, much time can be saved by using multiple technologists. Several suggestions are made concerning reframing the workflow. Boland stresses the availability of all information the technologist needs before the starting time of the examination. Delays due to improper information can easily be avoided. Also *reallocating preparatory procedures*, such as installing the intravenous access line outside then SR in a room close by, can decrease the time necessary in the SR significantly. Lateness of the patient or, especially, the medical staff can disrupt the flow of the process seriously. Procedures which can be done by other personnel after working hours have to be eliminated from the process. Restocking of material is for example one thing which is possible at times when the CT SR is not in use. There has to be spent time for professional development. It is commonly known that using a *multi skilled workforce* performs much better then using only specialists. Many hospitals already use *RIS* and *PACS*, but when this is not the case it is advised to invest in these information systems. RIS is critical to *measure and record productivity.* It is necessary to effectively schedule the patients. PACS made the change possible from printing the photos to *digital archiving.* When the process flow is already optimized, another way to increase capacity is to *extend the working hours.* Many more examinations can be done when the CT scan is also used in the evening or weekends. Boland stresses that the increased personnel costs will be marginally compared to the increased income due to more examinations.

Since life expectation is expected to increase and the population is aging rapidly, it is important to find how to cope with this high demand for medical services and to determine if current plans will still be feasible. Health care modeling is a field which studies the flow of an individual patient through the health care system. During these studies information is collected (such as patients arrival, departure, examination, and waiting times). With this information, departments can be compared, trends can be identified and bottlenecks found. With the use of simulation techniques, process adjustment options can be investigated to determine the impact and consequences. Ivatts and Millard discuss the benefits and several pitfalls of health care modeling in [19] and [20]. Determining the performance of the health care process is one of the difficulties and commonly used performance measures should be reconsidered precisely defined as well. For example, Ivatts and Millard explain the difficulties concerning bed occupancy. The number of beds in use was counted at midnight in numerous studies, but the bed occupancy during the afternoon could be much higher (some patients are discharged in the afternoon to spent the night at home). They regard the current health care management methods to support changes as 'black-box' models (you do not have a clue what really takes place). In these models, flawed performance measures are used to justify the changes. Using health care modeling with individual patient data helps to tackle this problem.

Simulation methods and queuing theory are used to make these decisions more sound. The number of beds a ward has is a performance measure of much interest. Using queuing theory, Green determines how many beds need to be availably to meet certain goals [15]. The simple queuing model is only applicable when service time and arrival times are close to the exponential distribution (in terms of coefficient of variation). In [47], Worthington applies queuing models to the hospital waiting list problem to compare waiting lists with or without feedback (patients are more likely to go to a hospital with a small waiting list). These models are useful to determine the effects of changes in demand and available resources. However, when these models become more complicated (emergency patients, priorities, dependence between service times, etc.) queuing theory is not sufficient. Unrealistic assumptions have to be made and the question arises to what extend the model still is an appropriate resemblance of reality. Another problem with queuing theory is calculating performance measures when steady state is not reached. In health care systems, steady state is hardly reached anytime. Methods to overcome this deficiency almost invariably make the model even more complicated [6].

Queuing theory has the advantage that it is able to produce expressions for performance measures. This way, the effect of the parameters (such as service and arrival rates) is exactly known. For many different systems, these measures are already calculated. To be able to derive these expressions, assumptions have to be made. When these assumptions are very unlikely, simulations can be used instead. Simulation methods have more flexibility. However, using simulations also comes with disadvantages. More time is needed to program, test and verify the model. Since a program is only capable in producing approximations, much computation time may be needed to obtain good estimates. The challenge in simulations is to model the reality in the least complex way, to keep data requirements at a minimum, but still be able to derive sound conclusions. A review on the advantages and disadvantages on these methods and several others to model patient flows in health care, is given in [6].

In [23], the basics of simulations are given. It introduces a framework to set up a simulation study which will be the main topic of the next chapter.

Chapter 2

Starting the simulation project

Simulation is a powerful tool when used correctly. Complex systems can be analyzed by determining performance of *what if* situations. When determining possible new appointment schedules, which is one of the goals of this part of the thesis, it is impossible to try many different schedules in practice. Time, money, and reputation will be wasted and possibly even patients safety will be at risk if schedules are tested in practice. Therefore, simulations are of vital importance to decision making.
The simulation study should of course be applicable in practice. Results from a time consuming simulation study are useless when not properly documented. Therefore, time and effort have to be put into making an Assumption Document. This document contains the information necessary to understand, implement and reproduce the simulation project.

2.1 Assumption document

The need of an assumption document is stressed in [23]. This document states all information needed before starting the actual programming stage. It should contain the following:

- Project goals and scope.
- Flow chart of the process to be modeled.
- Performance measures for evaluation.
- Detailed descriptions of the subsystems and their interactions.
- Clarification of the assumptions.
- Summaries of data which need to be used for model fitting.
- Limitations of the simulation model.

These items are defined for the CT scan process in this section.

2.1.1 Project goals and scope

To increase the number of patients examined on a day a new appointment schedule are introduced. This project tries to obtain the following goals:

1. Calculate performance measures for different appointment schedules.
2. Incorporate corresponding improvement actions to facilitate decision making.

With the obtained performance measures (which are defined in the Appendix) improvement actions are compared to make the CT scanning process more efficient. Schedules are tested to determine a better appointment schedule for scheduling patients.

2.1.2 SIPOC chart of the CT scan simulation procedure

A graphical description of the system helps to visualize the steps the process needs to take in order to get the desired results. The SIPOC chart, which is a macro-description of the process and is shown in Figure 2.1, describes the process in such a manner. This is a SIPOC of the simulation model and not the SIPOC of the process in reality.

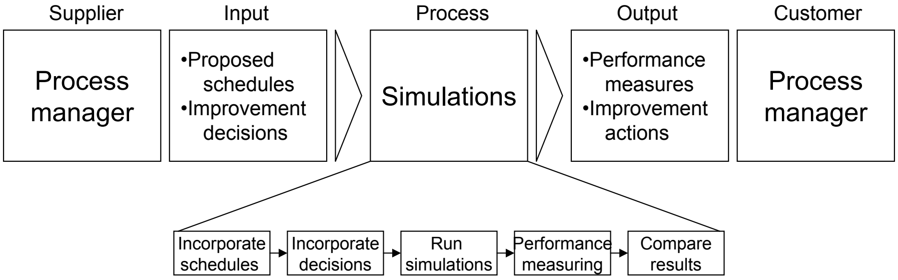


Figure 2.1: SIPOC-chart for CT scan simulation procedure

2.1.3 Patient attributes and performance measures

The system starts with a given appointment schedule. This contains the following information for the patients scheduled on a day (patient attributes):

- Patient's scheduled time (T_i).
- Patient type: out-patient or in-patient ($outp_i$).
- Type of CT scan examination: IVP, colon or other examination ($exam_i$).
- Whether contrast is injected (IV_i).
- Number of scans ($2ndscan_i$).
- Whether immediate diagnosis on the CT scan results is necessary or not ($diag_i$).

The subscript i denotes the attributes corresponding to patient i and are the input for a schedule.
To determine the influence of a certain adjustment on the performance of the system, several performance measures need to be calculated. The used notations of the measures and variables that are used are stated in the Appendix.

- Number of patients examined on a working day: $npday$.
  The number of patients examined on a working day consists of the scheduled patients (out- and in-patients) and the urgent patients. There will be tried to obtain schedules were this number of patients is as high as possible while keeping other measures at acceptable heights.

- Appointment lateness: $\sum_{i=1}^{npday} \max(0, aSR_i - T_i)$.
  In this measure aSR_i equals the actual entrance time of patient i in the SR. Whenever patient i is examined before his original appointment time, the appointment lateness will be 0. Appointment lateness is tried to keep at reasonable levels while increasing the number of patients on a working day. Minimizing the appointment lateness is not relevant since this would result in a schedule which accommodates far less patients. But disregarding this measure completely will of course result in high waiting times.

- Overtime: $\max(0, dSR_{npday} - t_{end})$.
  The day ends at t_{end}. If for some reason the schedule is delayed and patients have not finished their examinations, some overtime will be the result. It is also possible that an urgent patient arrives just before the end of the working day. This patient still has to be examined which causes overtime. Increasing the number of patients examined on a working day, overtime should also be taken into consideration. Allowing more overtime is a possibility to increase the number of patients drastically, but this is not advisable.

- Occupancy of the CT scanner: $\frac{\sum_{i=1}^{npday}(dSR_i - aSR_i)}{t_{end} - t_{start}} * 100\%$.
  A commonly used performance measure is the occupancy. This measure is the percentage of the time the SR is actually in use. An ideal occupancy equals 100%, but this value is highly unusual. Occupancy in health care is strongly affected by the inherent variability of arrival and service times. When increasing the number of patients examined on a working day, the occupancy increases when service times are kept equal. Low occupancy rates therefore also suggest that further improvements are possible.

- Possibility to take an uninterrupted break.
  At Deventer Hospital is was noted that the medical staff should jointly take a morning and a lunch break. This means that the SR will not be used for a quarter of an hour in the morning and for half an hour around noon. If for some reason the examinations are behind schedule (e.g. due to urgent patients, unexpected high service times or late arrivals) the breaks have to be compromised. To record this annoyance, the simulation model keeps track of the number of occasions in which it is possible to have the break.

- Waiting time urgent patients:
  $\sum_{i=1}^{npday} 1_{\{\text{patient i is an urgent patient}\}} * (\max(0, aSR_i - aWR_i))$.
  Urgent patients need to be examined within two hours. To determine if this is the case, the waiting times for the urgent patients are recorded. The waiting time is defined by the time between arrival and entrance in the SR. In some of the tested schedules, urgent patients are scheduled in emergency slots. Another possibility is to schedule urgent patients on their arrival time (taking the appointment time equal to the arrival time). Increasing the number of patients scheduled on a working day has influence on the expected waiting time of urgent patients. These waiting times need to be calculated to make sure urgent patients are treated in time. Also the probability is recorded that an urgent patient has to wait more than two hours.

Using the above performance measures, appointment schedules are simulated and performances compared.

2.1.4 Subsystems

A simulation projects consists of several subsystems.
The first step in the simulation is the alternation of the appointment schedule to the true arrival schedule. Data-analysis (see 2.1.6) was conducted to determine distributions which facilitate in calculating the true arrival times. Patients arrival times need to be simulated because they have high impact on performances.
In the following sections the notations described in Figure 2.2 will be used for the flow charts.

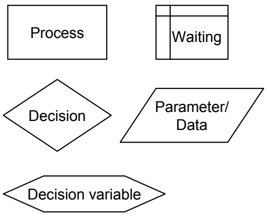


Figure 2.2: Flow-chart: Legenda

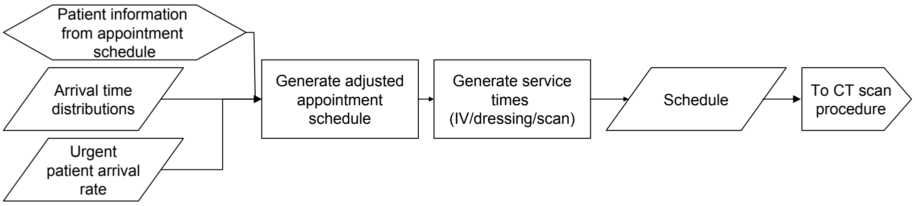

Patient information from appointment schedule
Arrival time distributions
Urgent patient arrival rate
Generate adjusted appointment schedule
Generate service times (IV/dressing/scan)
Schedule
To CT scan procedure

Figure 2.3: Flow-chart: Scheduling patients

The flow-diagram stated in 2.4 shows the steps a patient has to proceed to complete the CT scan procedure.

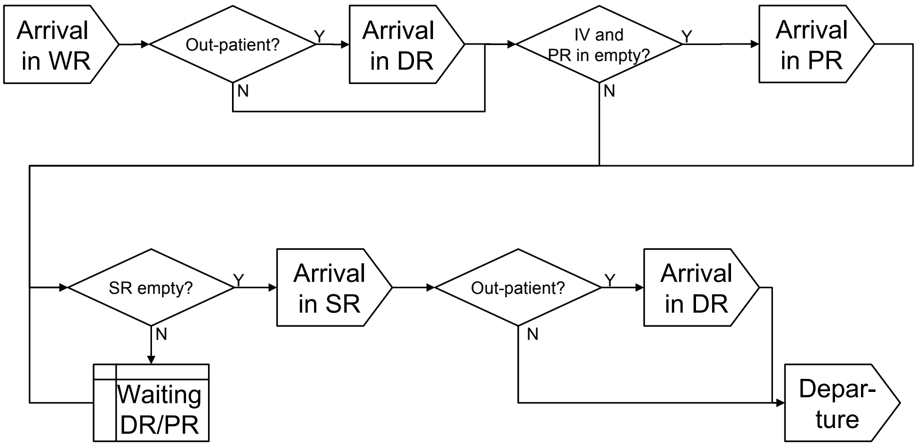


Figure 2.4: Flow-chart: Overall CT scan procedure

#DR
The number of dressing rooms (DRs) can be changed. In Deventer Hospital three of

the four DRs are currently used. The fourth acts as an exit for personnel. Therefore $\#DR = 1,\ 2,\ 3,$ or 4.

DR rules

During the DR routine, rules are used to make sure a patient does not have to wait an inappropriate amount of time. A patient in the WR is forwarded based on the residual service times of the patients in the remainder of the process. This estimate depends on the residual service time of a patient in the SR, the residual service time of a patient in the PR and the residual service times of the patients in the other DRs. If this amount of time does not exceed a predefined time length, the patient enters a DR.

PR rules

Similar to the DR rules are the rules in the PR. When a patient needs to receive contrast, the patient has the possibility to use the PR. If the SR is available, the injection takes place in the SR. Patients who do not need contrast skip this process step.

Subroutines

In Figures 2.5, 2.6, 2.7, and 2.8, the subroutines for an arrival in the WR, DR, PR, or SR, are represented by their flow charts.

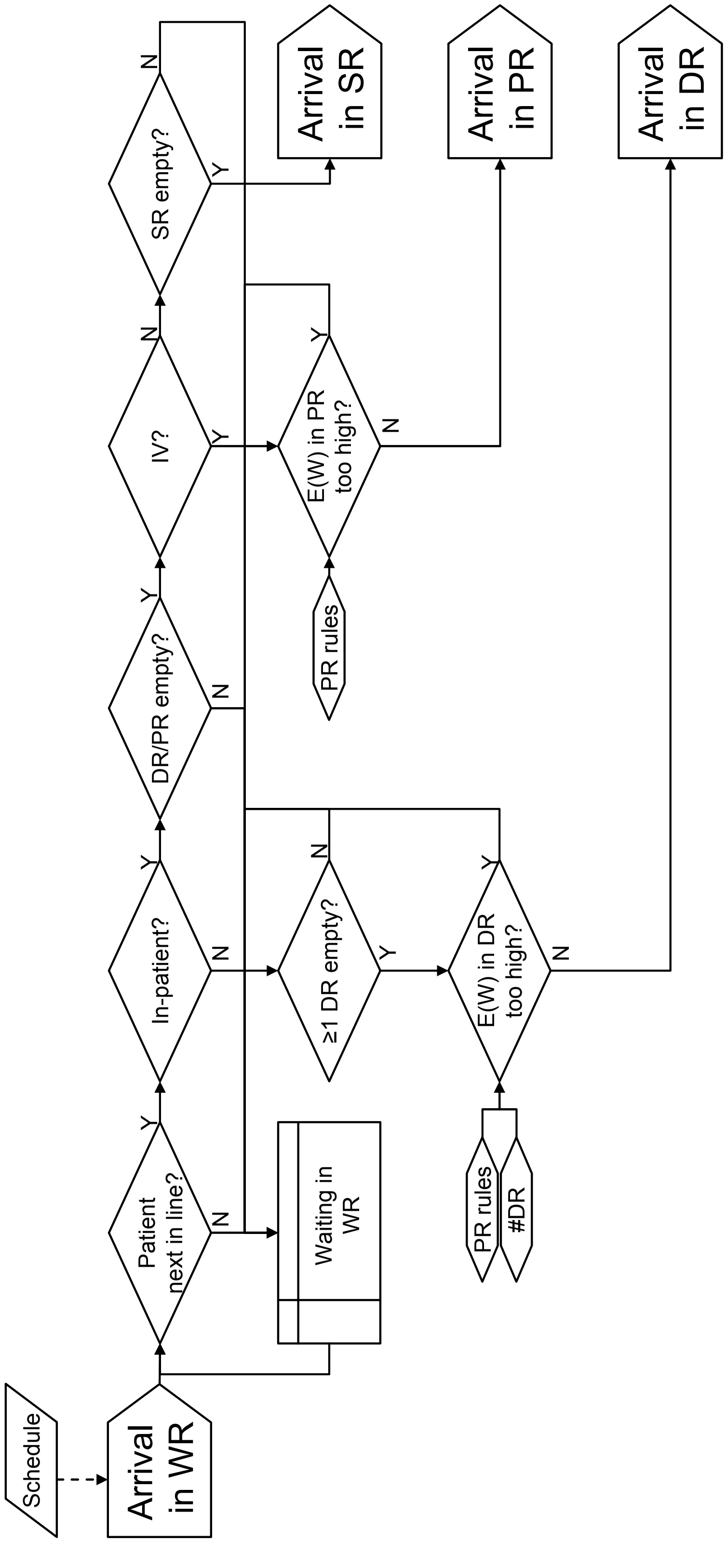


Figure 2.5: Flow-chart: WR routine

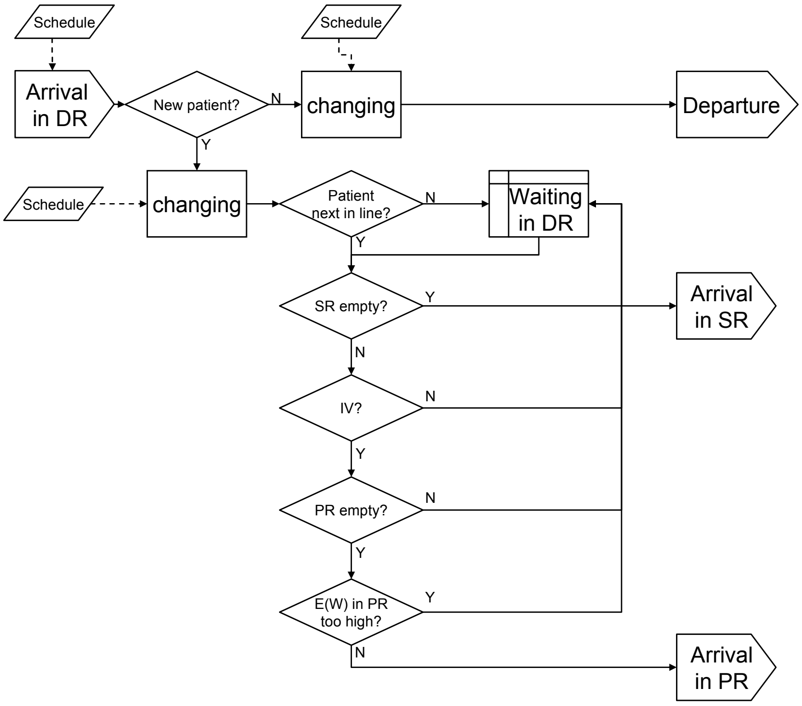


Figure 2.6: Flow-chart: DR routine

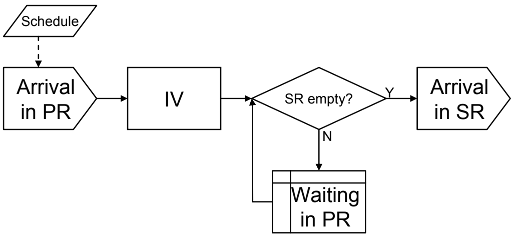


Figure 2.7: Flow-chart: PR routine

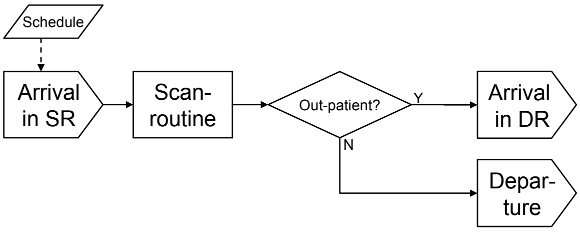


Figure 2.8: Flow-chart: SR routine

2.1.5 Assumptions

The following assumptions have to be made before proceeding with the data analysis.

- *The patient arrival rate does not fluctuate among days* (multiple runs). The drawback of this assumption will be explained in 2.1.8.

- *Residual service times can be estimated.* When a patient is being scanned, another patient can start undressing. This way, the scanner idle time decreases. Since the DRs are particularly small (1 by 1 meter) it is not appropriate to let a patient wait in the DR for a long time. Using the DR/PR rules, the medical staff estimated the waiting time in the DR/PR. By testing different distributions for this estimation, the effect is investigated.

- *Constant urgent patient arrival rate.* The system has to deal with urgent patients. With the use of data, an estimated arrival rate for the urgent patients was obtained. Since the arrivals were not recorded by time of entrance (but only by number per day) the assumption of random arrival times has to be made. Therefore the arrivals of emergent and urgent patients are considered to follow a Poisson process[1] with a predefined rate. Fluctuation in arrival rate of the urgent patients between days or months are not considered. One could argue that the arrival rate is influenced by seasons but this is not part of the simulation project. When this problem occurs in reality, the simulation can simply be adjusted.

- *Medical staff takes breaks if possible.* Adjustments to the breaks of the medical staff were not allowed. Sometimes it is possible that there is no time to let the two technicians take a break at the same time. The simulation program however only calculates the percentage where the break was possible.

- *Patients are examined according their arrival and appointment time.* Patients are served according to their planned arrival time, but there is a possibility that a later patient is served first due to earliness of this patient or lateness of the preceding patient. Due to this overtaking, an earlier planned patient receives a longer waiting time (when a later patient arrives much too early). In reality the medical staff has the possibility to change the order of examinations or to deliberately let a patient wait to maintain the flow in the system.

- *Dressing time equals undressing time.* The time an out-patient needs to get dressed after the examination equals his undressing time.

- *Personnel is on time.* Personnel is assumed to arrive several minutes before the actual opening of the SR so patients can enter the DR to undress.

- *No break downs of equipment.* The scanner is assumed not to break down during a day and also enough personnel is assumed to be available during the working hours.

- *The process is not influenced by different employees.* Personnel is not assumed to influence any performance measures. Different technicians are not considered in this simulation project.

- *Infinite WR size is possible.* The size of the WR is of no interest and therefore assumed to be infinite.

2.1.6 Data-analysis

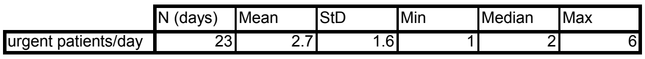
The data needed for this simulation project is acquired in Deventer Hospital. During 5 working days, the necessary times were recorded at the radiology department.

[1] Arrivals are random with exponentially distributed inter arrival times.

A total of 93 examinations were measured. The data analysis gives the distributions needed to simulate the CT scan procedure. In the Appendix, a graphical review of the data analysis is given which is performed using Minitab 15 [27].

Patient type

The data consists of 63 out-patients, 22 in-patients, 8 urgent patients. The number of urgent patients was believed to result in an underestimation for the average number of arrivals of urgent patients. An additional data inquiry was done to determine a better estimate for the urgent patients arrival rate. During a extra 23 days, a total of 62 urgent were examined at the CT scan. This resulted in a mean of 2.7 urgent patients per day. For the simulation, the arrivals of these urgent patients are assumed to follow a Poisson process with mean 2.7. This also includes emergent patients. In Figure 2.9 a statistical overview can be seen.

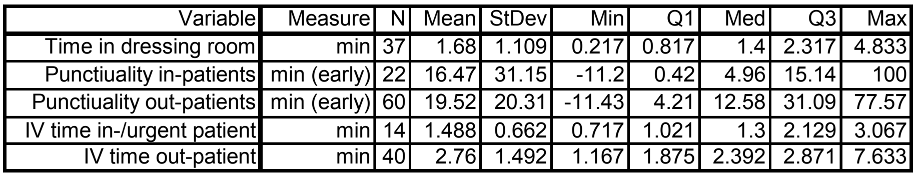


	N (days)	Mean	StD	Min	Median	Max
urgent patients/day	23	2.7	1.6	1	2	6

Figure 2.9: Statistical overview urgent patients

Punctuality

In practice, patients do not arrive precisely at their scheduled appointment times. Patients are either too early or too late. Punctuality measures the time a patient arrives before his appointment time. A punctuality of -5 will correspond to a situation where a patient arrives 5 minutes after his appointment time. Punctuality influences the performance during the day. When a patient arrives at late the CT scan may become idle. To take this into account in the simulation, out-patients' punctuality is simulated using a 3-parameter lognormal distribution. In-patients' punctuality was fit using the normal distribution. Since in-patients come from other departments in the hospital, it was possible to examine the patient at an idle period far before the scheduled appointment time. From the 22 in-patient observations, only 19 were applicable in determining the punctuality. Due to missing values for out-patients 59 observations were usable. The basic statistics of punctuality for both patient types can be found in Figure 2.10. A graphical summary of the analysis is shown by Figures 12.1 and 12.2 in the Appendix.



Variable	Measure	N	Mean	StDev	Min	Q1	Med	Q3	Max
Time in dressing room	min	37	1.68	1.109	0.217	0.817	1.4	2.317	4.833
Punctiuality in-patients	min (early)	22	16.47	31.15	-11.2	0.42	4.96	15.14	100
Punctiuality out-patients	min (early)	60	19.52	20.31	-11.43	4.21	12.58	31.09	77.57
IV time in-/urgent patient	min	14	1.488	0.662	0.717	1.021	1.3	2.129	3.067
IV time out-patient	min	40	2.76	1.492	1.167	1.875	2.392	2.871	7.633

Figure 2.10: Overview of individual patient data

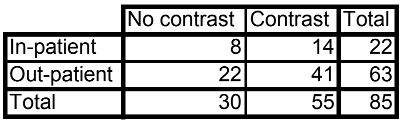
Changing in the DR

Out-patients need to undress prior to the examination. For in-patients and urgent patients this is not the case. During the measure phase, the entrance time of the out-patients in the DR was recorded. Subtracting this time from the entrance time in the SR results in an estimate for the time spent in the DR. This time is a combination of undressing and time to go from the DR to the SR. Since patients may wait in their DR because the previous patient is still being examined, several observations

were left out. From the 63 out-patients still 37 observations were usable. These were selected by comparing the departure time of the previous patient with the entrance time of the next patient in the DR.
In Figure 12.4 in the Appendix a probability plot shows an appropriate fit when using a Weibull distribution. The basic statistics of the time spent in the DR can be found in Figure 2.10.
After acquiring the scan, the patient returns to his DR to get dressed.

IV

Figure 2.11 shows the number of IV-patients.



	No contrast	Contrast	Total
In-patient	8	14	22
Out-patient	22	41	63
Total	30	55	85

Figure 2.11: Overview of contrast demand

Fisher's exact test finds a p-value of 0.258. Therefore the patient type has no significant influence on the percentage of patients needing an examination with contrast or not enough data was available to notice a significant difference. The number of observations is insufficient to properly estimating percentages. When scheduling random arrivals this flawed estimate of the true percentage could influence the simulation.
Prior to the actual admission of contrast, an IV access line had to be installed. The possible improvement action of installing the IV line in the preparation room could influence the time the installation takes. However, a Kruskal-Wallis test showed that the change from pricking in the SR to pricking in the PR did not influence the pricking time.
Using another Kruskal-Wallis test showed that patient type did influence the IV time. The number of in-patients in the analysis was quite small as well as the number of urgent IV-patients. Because it was not believed that the time it takes to install an intravenous access line on an in-patient, significantly differs from the time is takes on an urgent patient, the observations of the urgent patients and in-patients were combined. This resulted in 16 observations. The basic statistics can be found in Figure 2.10. For both patient groups, the lognormal distribution gave a good fit, see Figure 12.3 in the Appendix.
The data included 7 urgent patients, from which only 2 needed IV. Since the probability of an urgent patients needing contrast has to be known for the simulation, this is estimated at 2/7. Due to the very small sample size, this probability can be very different in reality. This will be part of the sensitivity investigation in section 4.2.

Usage of SR

Before the start of the improvement project, the installing of the IV line was done in the SR. Therefore most data for the SR includes this pricking time. In order to obtain a more reliable transfer function for the time spent in the SR, the pricking time for patients pricked in the SR was subtracted from the total time spent in the room. Using a transfer function to determine the SR usage time for all patients proved inappropriate. Treating the out-patients with contrast apart (still 29 patients) resulted in a nice fit for these patients as well as an appropriate transfer

function for the other patients (see Figures 12.4, 12.5 and 12.6).
The SR usage time of out-patients who only needed contrast seemed to follow a lognormal distribution. For the other patients the following transfer function was obtained:

$$\ln(\text{SR usage}) = 1.88 + 0.786\text{Recovery} + 0.331\text{Special exam} + 0.538\text{Diagnosis}$$

where dummy values were used to indicate when a patient needs recovery (and thus was pricked), a special examination or a consult from the specialist.

2.1.7 Improvement actions

In the preceding sections several improvement actions were stated. This section shortly summarizes the improvement actions or decisions possible in the simulation project.
Determining schedules includes making lots of decisions which clearly is an improvement action on its own. How can in-patients and out-patients be planned? Are they treated in blocks by examination type or are they planned at random? These decisions are treated in the following chapter on appointment scheduling.
Other improvement actions are shortly discussed in this section.

- The use of the PR is one of the proposed improvement actions and is incorporated in the simulation project.
- Patients are called in the DR and the PR after an estimation on the waiting time in these rooms by the technicians. These estimations can vary from best (the residual service time and dressing times are known exactly) to worse. Different parameters can be used to simulate the accuracy of the estimation.
- The number of DRs can vary from 1 to 4.

2.1.8 Limitations

The following limitations need to be known about the simulation.

- The simulation is considered for one day only. One day is simulated several times in order to get the desired performance measures. A change in arrival rate from either patient type can not be simulated. The drawback of this simulation is that when capacity raises the requests for the CT scanner probably increase [3]. The interaction between increased capacity and demand is not part of this simulation project.
- Rare events, like patients who need a large number of scans or have other complications that increases the time spent in the SR by a particular large amount, can not be incorporated. When known, several slots can be reserved for these examinations.
- Emergency patients are not treated separately since not enough data was available. Therefore these are treated as urgent patients. Although their number is assumed small, they seriously effect a schedule. Ad hoc changes by medical staff are possibly needed to cope with an emergent patient. This simulation project however is unable to determine the influences of these decisions.

With this assumption document, the simulation program can be made. As input, an appointment schedule is necessary. The next chapter gives an introduction into appointment scheduling, as well as the advantages and disadvantages of relevant schedules for the CT scan process.

Chapter 3

Appointment scheduling

A way to manage the waiting times of patients and idle times of the equipment is using appointment scheduling. This refers to a series of decisions which are of influence on the patients' scheduled time. In the larger part of the 20th century, most medical practitioners used an appointment system where the patient received just an appointment date. Patients were seen on a first-come, first-served basis which resulted in high waiting times, but also low idle times [5]. This trade-off between patients' waiting times and medical staff idle times is important. The true pioneers on this subject are Bailey and Welch who wrote a historical paper [44] in the Lancet. After this publication, many papers followed discussing different appointment schedules. The paper of Cayirli and Veral [5] gives a literature overview on appointment scheduling. The advantages and disadvantages of several appointment schedules are considered in this chapter and the applicability on the CT scan case is investigated.

When designing an appointment system, several variables have to be fixed. Cayirli and Veral call these variables the appointment variables. Combining them, results in an appointment rule which should fit the investigated situation. In basic, the appointment schedule is set up by three variables. The first variable determines the size of the slots. A working day is divided in slots; the time intervals in which patients are assumed to be served. This first variable determines how many patients are assigned an appointment time equal to the starting time of the slot. This number can be one, resulting in individually planned patients (which is most often used in medical environments nowadays). When the slot size is chosen greater than one, groups of patients (or jobs, customers) are assigned the same appointment time. This appointment variable is for example used in a blood bank where many people can be served at the same time. On a day, the slot-size can stay constant or is allowed to vary.

The next decision to be made is whether to include a special begin slot at the beginning of the day. Bailey was the first to consider this possibility for scheduling patients. Introducing a begin-slot where two patients are scheduled instead of one, results in a lower idle time for the medical staff while it has a limited effect on the patients' waiting time. Schedules without a special begin block normally result in more idle time at the beginning of a day [44].
The last appointment variable determines the interval between two successive appointment times (length of a slot). These intervals can be constant or variable.

Any combination of the three appointment variables gives a possible appointment rule. These rules have been studied extensively, see [5] for an overview. In the following section some relevant appointment rules and their characteristics are discussed.

3.1 Appointment rules

Single block

This rule schedules every patient at the same time at the beginning of the day. The advantage of this rule is that the probability of idle time for the medical staff is very low, because all the patients are present at the beginning of the day; there is always work. The major disadvantage of this rule is that waiting times can be huge. Since patients are served based on first-come, first-serve, patients might be tended to arrive very early. The WR will be on tremendous stress. Nevertheless this appointment rule is still used sometimes, also because it requires the least administrative effort [5].

Individual slots with a fixed interval time

A more advanced appointment rule is one which uses a slot of size 1 with a fixed time interval. This means that every patient is assigned a different arrival time and the time between appointments is always the same. By scheduling the patients at different times, part of the waiting time is eliminated. The idle time might however rise to undesirable heights. Especially in the beginning of the day there is some idle time (assuming the interval times are balanced with the desired demand) and at the end of the day this could result in overtime for the medical staff.

Individual slot with a variable interval time

To balance the load more equally over the day, the intervals can be adjusted. Making the intervals at the beginning of the day more tight and loosening them at a later time of the day could deal with the high idle times. The waiting times for the patients is distributed more evenly over the patients since before they were much higher for later patients. Wang derived in [42] an optimal appointment schedule for identical independently distributed service times and uniform waiting costs where the intervals increase towards the middle of the day and then decrease.

Individual slots with fixed interval time and a begin slot

The major drawback of fixed interval times is the higher idle time at the beginning of the day. To decrease this idle time, a possible two, three or perhaps more patients are scheduled at the starting of the working day. Literature shows that waiting times are kept within reasonable bounds while the idle time of the medical staff decreases substantially [44].

Individual slots with variable interval time and a begin slot

By allowing more patients to be scheduled at the first slot, there is a buffer to make sure the medicals staff has enough work to keep busy. Variable interval times are used to distribute the waiting times of the patients more evenly on the day. This appointment rule is the most flexible of the rules, the intensive administrative effort and workability should also be taken into consideration.

3.1.1 Adjustments

Patient attributes have a possible influence on the appointment system. Taking these differences into consideration can result in better appointment schedules. This way the time intervals between appointments are more balanced which is in favor of both the waiting and idle times.

Out-patients can be scheduled in advance while urgent patients arrive unannounced. An urgent patient also needs to be treated within a certain time interval. For out-patients this is not the case (however a very long waiting time is desirable). To deal with these different patient types decisions have to be made. Do we leave slots empty in the schedule to accommodate arrivals of non-scheduled patients? Are urgent and emergent patients put in front of the queue? Is a patient allowed to interrupt another patient's treatment or has the treatment to be concluded first? The standard ways to deal with these non-scheduled patients are reserving special slots on the day or increasing the time interval between appointments.

There is a possibility that patients do not turn up for their appointment and time is wasted by waiting. These patients are called *no-shows*. When the percentage of no-shows is substantial and other methods, like different reminder techniques and fees, have not reduced the number of no-shows enough, the appointment system has to be adjusted to deal with these no-shows. Increasing the slot size of certain slots (at times where the number of no-shows is large) or decreasing the time interval between the appointments, should reduce the effect of no-shows.

It might also be possible to have so called *walk-ins*. In some medical sections, it is possible for patients to arrive while they were not scheduled. For example, a patient has visited his doctor and an X-photo has to be taken but it is not urgent. The patient can go to the radiology department to see if there is of a free spot, avoiding the necessity to return at a later time. When this occurs often, the appointment schedule can be adjusted. Most often the occurrences of no-shows and walk-ins are not equally spread, therefore the probability that they cancel each other out is small.

Combining all these decisions should allow the scheduler to derive an appointment schedule for the precise situation.

3.1.2 Appointment scheduling for the CT scan

With the appointment schedules discussed above, different schedules were made for the CT scan case.
During the data analysis it became clear that dividing the patients over several groups would reduce the inner group variance drastically. Therefore, patients with different characteristics are scheduled differently [9].
before scheduling, the patients are grouped in the following groups:

- out-patients without a special exam
- patients needing an IVP or colon exam
- in-patients without a special exam

In the remaining part of this chapter, schedules for the CT routine are given. These schedules are also graphically represented in the Appendix.

Current schedule (S_C)

First the currently used schedule is described. The working day starts with a patient scheduled at 8:15. Before the lunch break, only out-patients are scheduled (most of the time, unless there are no out-patients scheduled due to low demand and an in-patient is helped instead). Every morning a patient is given 15 minutes (the interval time between two appointment times is 15 minutes with the exception of a morning break from 10:15 to 10:30. and a block reserved for urgent patients at 9:00). There is no distinction in different examinations. Also Colon and IVP examinations are given 15 minutes. The lunch break is from 12:30 till 13:00. As already noted, both technicians take a break at the same time so scanning during these breaks is impossible. After the lunch break, the out-patients are treated which could not be scheduled in the morning, followed by the in-patients. For an in-patient a 20 minutes interval is reserved. On a working day, the average number of patients helped was 15.6 out-patients (including urgent patients) and 6.1 in-patients.

Several points are remarkable about this type of scheduling. By treating every out-patient the same way, high idle times are possible because of the high variability in service times. Treating the out-patients with special exams differently could result in a better appointment schedule.

Since not enough slots are available for urgent patients, there are high waiting times for urgent patients. Increasing the number of emergency slots and distributing them more evenly over the day should improve the system, especially for urgent patients[1]. By using these emergency slots, deviations from the expected service times can be diluted at an early time without delaying the whole schedule.

The first proposed schedule (S_{P1})

In the improvement phase of the CT scan project, the team working on the project proposed a new schedule. During the analysis, several improvement points became clear and were incorporated into a new appointment scheduling system. The installation of the intravenous access line was moved to the PR. The day was divided in two parts, one before the lunch and one after the lunch. From 8:15 till 12:00 there are 24 slots of 10 minutes each (and a morning break of 15 minutes), from which 4 are reserved for urgent patients. After the break two slots for special exams are reserved of 20 minutes each. At the end of a working day, 8 slots are reserved for in-patients, also 20 minutes each. This way, the hospital is able to help 20 out-patients, 2 patients who need a special exam and 8 in-patients, which results in a total of 30 scheduled patients.

Introducing this new appointment schedule would increase the capacity from 21.7 (including urgent patients) to 30 (without urgent patients).

Schedule S_{P1} *noE* is a slightly adjusted version of schedule S_{P1} with no slots reserved for urgent patients. Instead of 5 slots and for urgent patients in the morning, schedule S_{P1} *noE* has four blocks of 5 slots each. The slots in a block are of the following consecutive lengths: 10, 15, 10, 15 and 10 minutes.

The second proposed schedule (S_{P2})

Another proposed schedule, for which a pilot-run is implemented at Deventer Hospital, is a schedule with 20 out-patients, 6 in-patients and 3 special exam slots. In the morning the out-patients are scheduled and before the lunch break 3 special exams

[1] High access times for urgent patients was noted also in another improvement project at Deventer Hospital.

are scheduled. The number of blocks reserved for urgent patients in the morning is 4 and in the afternoon 2. Immediately after the lunch break, 6 blocks of 10 minutes each are reserved for out-patients needing no special exam. At the beginning of the day, after each emergency block and after each break, an out-patient who does not need contrast is scheduled.

Reversed proposed schedule (S_{R-P})

In the proposed schedules, the out-patients (patients with a relatively short service time and variance) are treated at the beginning of the day. The in-patients and special exams are scheduled after the lunch break. Reversing this schedule results in a schedule where out-patients are examined at the end of the day, while in-patients are scheduled earlier.

Two blocks schedule (S_{2B})

As a reference, a schedule will be calculated consisting of two blocks, one in the morning and one after the lunch break. Every patient is scheduled at the beginning of a block. Of course, this is a very simplistic schedule and good performance is not expected (for patients), but as a reference it might be informative to simulate this schedule.

Variable interval time schedule (S_V)

The next schedule to be simulated will be using a variable interval size. When patients are punctual and the appointment times are fixed, it is intuitive to arrange patient arrivals in order of their service time variances [42]. It was shown that the inter arrival times follow a dome shaped function. This shape is loosely incorporated in a schedule to determine its performance.

Multi-block schedule ($S_{Multi-B}$)

An effective way to minimize the idle time of important equipment is scheduling multiple patients at the beginning of the block. The drawback of this method is an increasing waiting time for the patients. The morning is divided into four blocks. In each block, the patients are scheduled at the beginning. Urgent patients arriving in a block are scheduled at the end of the block or treated as ordinary arrivals.

Urgent patient slots

The schedules discussed above are combined with two types of methods to handle emergency patients. The first method schedules the urgent patients upon arrival in the nearest emergency block. These blocks are reserved for emergent and urgent patients, but can also help to compensate for extensive overtime from other patients. By doing this, the scheduled patients are less likely delayed by urgent patients, but one has to consider the extra waiting time for urgent patients. By choosing enough emergency slots in the schedule, an urgent patient is almost always helped within 2 hours. Shifting with these blocks can be used to determine a more suitable schedule.

Another method schedules urgent patients at their arrival time. When the call for an examination of an urgent patient arrives, the urgent patient can be regarded as a patient for which the appointment time is equal to the arrival time. When there are regular patients present who were scheduled before the urgent patient's arrival time, these patients are served first. Regular patients who are present in the WR with an appointment time later than the urgent patient, have to wait until the urgent patient is examined. This method results in lower waiting times for the urgent patients. The expected waiting time of the regular patients however is assumed to increase.

The schedules discussed in this chapter were simulated. Next chapter introduces some specific modifications on these schedules and the results of the simulations.

Chapter 4

Simulation results

The schedules as discussed in the previous chapter have been simulated for 1000 runs which resembles 1000 working days. Averaging the obtained measures from a day (a run) results in an estimation for the actual performance measure.
Every proposed schedule resulted in a series of performance measures which are already discussed in 2.1.3. Figure 4.1 presents results for several relevant schedules. Several additional schedules have been simulated as well; these results are shown in Figure 12.18 in the Appendix. The additional schedules were obtained by some adjustments on the schedules which were previously discussed. The notation in Figures 4.1 and 12.18 is as follows:

- *noE* represents schedules without emergency slots. For example S_{P1} *noE* is schedule S_{P1} without emergency slots. Instead of a block with 5 slots of 10 minutes each and an emergency slot of 10 minutes in the morning, this schedule has blocks with slot lengths 10, 15, 10, 15 en 10 minutes. Also for the second proposed schedule, an adjusted schedule was made with no emergency slots. Two different schedules were constructed, one with individual slots and one with slots of two appointments.

- Schedules with *Random* in the name use random scheduling of the patients. For S_{P2} there are two options: *Random A* and *Random B*. Since it was proposed to schedule an out-patient which does not need contrast after the breaks or emergency blocks, there has to be a distinction between keeping incorporating this assumption or not. Random A schedules do schedule an out-patient without contrast in those slots, while Random B schedules do not. The probability for the other slots to be filled with an out-patient who does not need contrast will decrease in Random A schedules. Only Random (no A or B) means the same as with Random B (every slot is filled at random with the appropriate patient attributes).

- The $S_{P2}-1$ *outp* schedules are the same as the schedule S_{P2} except it schedules one out-patient less. The out-patient removed from schedule S_{P2} was the last out-patient scheduled in the afternoon. The slots for in-patients were adjusted to fill the gap.

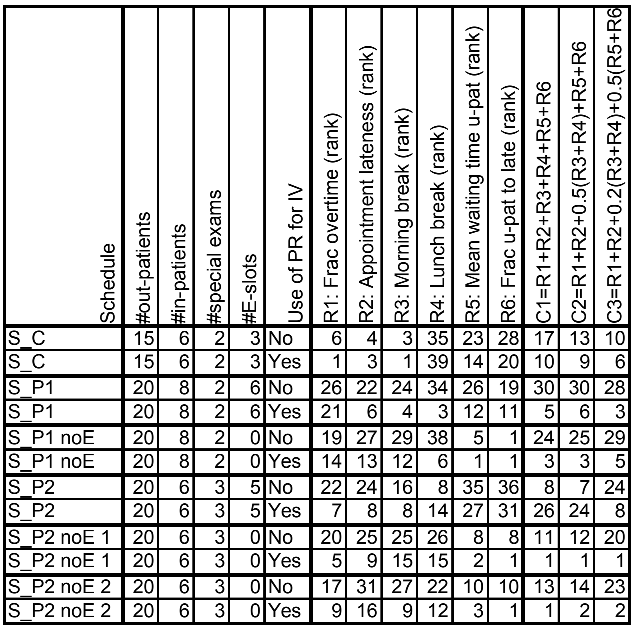
In Figure 4.1 the performance measures are shown for the most relevant schedules. The results for the other schedules are shown in Figure 12.18 in the Appendix. For the appointment lateness and the waiting of urgent patients the mean, variance, and quantiles are given. With these quantiles, the mean and the variance of the different schedules can be better investigated than by only using the mean and variance. A mean appointment lateness of 10 minutes might sound appropriate, but when 80% has an appointment lateness of 0 minutes and 20% of 50 minutes, the schedule might not be suitable after all. For urgent patients it was necessary to be examined within 120 minutes after arrival. The percentage of urgent patients who have to wait for more than two hours is given.

							Occupancy		Overtime (min)			Appointment lateness (min)				Break		Waiting time urgent patients (min)				
Schedule	#out-patients	#in-patients	#special exams	#E-slots	Use of PR	Average #urgent patients	Mean	Var	Frac	Mean	Var	Mean	Var	50%-Quantile	80%-Quantile	Morning	Lunch	Mean	Var	50%-Quantile	80%Quantile	%>120
S_C	15	6	2	3	No	2.799	0.68	4.0E-03	0.13	11.55	1.21E+02	5.64	7.12E+01	0.71	11.37	0.98	0.64	39.36	1.78E+03	25.68	72.65	6.29
S_C	15	6	2	3	Yes	2.678	0.63	3.6E-03	0.09	13.29	1.21E+02	4.63	5.73E+01	0.00	9.32	0.99	0.60	29.26	1.16E+03	17.03	54.81	2.24
S_P1	20	8	2	6	No	2.69	0.84	3.6E-03	0.34	16.21	2.93E+02	10.96	1.60E+02	7.78	19.42	0.84	0.67	42.43	1.01E+03	39.39	69.02	1.64
S_P1	20	8	2	6	Yes	2.737	0.77	3.2E-03	0.26	12.78	1.44E+02	6.36	7.82E+01	2.51	12.36	0.96	0.87	28.68	6.19E+02	23.30	49.30	0.26
S_P1 noE	20	8	2	0	No	2.678	0.84	3.4E-03	0.24	16.04	1.98E+02	11.62	1.94E+02	7.43	20.51	0.79	0.61	20.23	2.57E+02	17.11	31.70	0.00
S_P1 noE	20	8	2	0	Yes	2.677	0.78	3.3E-03	0.20	13.64	1.78E+02	7.22	1.09E+02	2.49	13.75	0.92	0.85	15.72	1.79E+02	13.35	24.33	0.00
S_P2	20	6	3	5	No	2.673	0.83	3.4E-03	0.26	18.26	2.47E+02	11.31	1.96E+02	7.24	20.06	0.90	0.85	66.77	2.76E+03	54.59	115.13	18.48
S_P2	20	6	3	5	Yes	2.73	0.76	3.6E-03	0.14	13.99	1.53E+02	6.95	9.33E+01	2.58	13.40	0.95	0.82	43.93	1.80E+03	31.10	78.29	7.36
S_P2 noE 1	20	6	3	0	No	2.648	0.83	3.3E-03	0.25	19.39	3.55E+02	11.51	2.54E+02	5.76	20.28	0.84	0.77	22.15	3.77E+02	17.49	34.35	0.11
S_P2 noE 1	20	6	3	0	Yes	2.641	0.77	3.5E-03	0.13	13.09	1.44E+02	6.96	1.14E+02	1.57	13.20	0.91	0.81	16.00	1.93E+02	13.47	24.42	0.00
S_P2 noE 2	20	6	3	0	No	2.692	0.83	3.4E-03	0.24	16.05	2.82E+02	12.50	2.65E+02	7.25	21.88	0.84	0.78	23.15	3.92E+02	18.60	35.55	0.19
S_P2 noE 2	20	6	3	0	Yes	2.683	0.76	3.3E-03	0.15	13.01	1.71E+02	7.64	1.20E+02	2.53	14.58	0.94	0.84	16.29	2.06E+02	13.73	25.75	0.00

Figure 4.1: Simulation results of several relevant schedules

Figure 4.1 in the Appendix and Figure 12.18 show that assigning patients at random to slots results in worse performance. This is obviously because the work can not be balanced. The resulting performance measure can be seen as an upper bound for the actual situation when using the corresponding schedule.

Further analysis is carried out with the best schedules found in the simulations. To determine the schedules which perform best relative to the other schedules, the performance measures of Overtime, Appointment lateness, Morning and Lunch break, Waiting time for urgent patients and the Percentage of urgent patients who have to wait too long, are transformed to their respective rank. For each of the performance measures, the rank gives its corresponding top 40 place. For example, the appointment lateness for the current schedule with random arrivals was the lowest compared with the other schedules, resulting in a rank of 1. For some performance measures it is of course more appropriate to give a higher value a higher rank (such as with morning break). Giving the ranks different weights for different performance measures it is possible to give more importance to Appointment lateness than to the Morning break. Since in practice the breaks are more flexible (easier to take a break at a later time, separate breaks or shorter breaks), the performance measure concerning these breaks is given a lower weight than the performance measure Waiting time for urgent patients. In Figure 4.2 and Figure 12.19 in the Appendix, the ranks for several performance measures are given. Using different weight functions the different schedules can be compared.
Overall the schedules S_{P2} *noE* 1 and 2 perform best, followed by S_{P1} *noE* and S_{P1} depending on which weights were chosen. Remarkable is the observation that the current schedule is not the best. It has the best ranks in Appointment lateness and Overtime, but when taking the breaks and urgent patients into consideration, the current schedule performs worse.



Schedule	#out-patients	#in-patients	#special exams	#E-slots	Use of PR for IV	R1: Frac overtime (rank)	R2: Appointment lateness (rank)	R3: Morning break (rank)	R4: Lunch break (rank)	R5: Mean waiting time u-pat (rank)	R6: Frac u-pat to late (rank)	C1=R1+R2+R3+R4+R5+R6	C2=R1+R2+0.5(R3+R4)+R5+R6	C3=R1+R2+0.2(R3+R4)+0.5(R5+R6
S_C	15	6	2	3	No	6	4	3	35	23	28	17	13	10
S_C	15	6	2	3	Yes	1	3	1	39	14	20	10	9	6
S_P1	20	8	2	6	No	26	22	24	34	26	19	30	30	28
S_P1	20	8	2	6	Yes	21	6	4	3	12	11	5	6	3
S_P1 noE	20	8	2	0	No	19	27	29	38	5	1	24	25	29
S_P1 noE	20	8	2	0	Yes	14	13	12	6	1	1	3	3	5
S_P2	20	6	3	5	No	22	24	16	8	35	36	8	7	24
S_P2	20	6	3	5	Yes	7	8	8	14	27	31	26	24	8
S_P2 noE 1	20	6	3	0	No	20	25	25	26	8	8	11	12	20
S_P2 noE 1	20	6	3	0	Yes	5	9	15	15	2	1	1	1	1
S_P2 noE 2	20	6	3	0	No	17	31	27	22	10	10	13	14	23
S_P2 noE 2	20	6	3	0	Yes	9	16	9	12	3	1	1	2	2

Figure 4.2: Simulation results in ranks of several relevant schedules

4.1 Adjustments to CT scan area

Using the results obtained in the previous section, several high potential schedules are investigated more closely in this section. The schedules S_{P2} *noE* 1 and S_{P2} *noE* 2 perform particularly well and S_{P2} *Random A* and S_{P1} *Random* are the proposed schedules so these are investigated with adjustments also. Since reallocating installing the intravenous access line to a PR proves to significantly increase the system's performance, only simulations are run with this property. Several possible improvement actions are simulated to determine their impact. Results can be found in Figure 12.20 in the Appendix.

Re-arranging the E-slots

In schedule S_{P2} *Random A* the emergency slot from 15:20 is moved to 14:00 (schedule S_{P2} *Random A* (2)) to find out whether the performances for especially the urgent patients improve. The effect of this adjustment is a decreases in the probability of waiting too long for urgent patients. Appointment lateness as well as the probability of overtime however slightly increases.

Adjusting the number of DRs

When the DRs are the bottleneck of this process, increasing the number of DRs to four can result in better performance. It can also be possible to close a DR when this does not influence the performance significantly. This DR could then be used for other purposes.

Removing increased the appointment lateness. Whenever two patients are in service (for example one in the SR and another getting dressed) it is not possible for a third to enter a DR. This results in additional idle time for the SR. Adding a DR has practically no influence.

Improving the estimation on residual examination time

Before a patient goes to the DR or the PR, the medical staff makes an estimate on the waiting time in the relevant room. Improving the capability of the medical staff to determine this waiting time might result in patients being sent to the relevant room at a more appropriate time. This could result in less idle time of the scanner. The simulations done previously used $\mathcal{N}(0,1)$ to simulate the medical staffs estimate of the residual service time. This distribution is altered in the constant 0 and $\mathcal{N}(0,4)$. Being able to accurately estimate the residual service times showed to influence the performance.

4.2 Sensitivity analysis

Assumptions are needed to model the process in simulations. The effect of a change in these assumptions has to be tested. For example it is possible that the system is highly sensitive for the arrival rate of urgent patients. If this is the case, adjustments have to be made to sustain equal performance. Sensitivity analysis is also important for finding bottlenecks. A minor change in service times resulted in significantly different performance which probably means that the SR acts like a bottleneck in the system. Knowing this, additional adjustments might be possible to improve the system. The schedules S_{P2} *noE* 1, S_{P2} *noE* 2, S_{P2} *Random A*, S_{P2} *Random A*2 and S_{P1} *Random* resulted in the most promising results, therefore these schedules are chosen for the sensitivity investigation. Results can be found in Figure 12.20 in the Appendix.

Adjusting the service times

By adding another random variable to the SR usage time, the effect a slight increase or decrease in the SR usage time has on the overall performance can be investigated. For the schedules, the SR usage time is adjusted by the following distributions:

- $\mathcal{N}(-1,0.5)$
- $\mathcal{N}(1,0.5)$

Only a slight adjustment in SR usage time showed to severely affect the performance of the system. This suggests that the SR is still the bottleneck of the system. Continuous improvements are necessary to achieve better performance.

Adjusting patients' punctuality

Patients' punctuality is believed to have a high impact on the system [3]. To simulate the situation where patients arrive almost always before the appointment time, patients' punctuality can be taken equal to, for example, 10 minutes. This also gives enough time to get undressed before being scanned. Taking this punctuality results in a better performing system, it might be recommended that more effort is being put to make sure patients arrive on time. As a reference, the schedules where the patients arrive precisely on time are also simulated. Comparing a punctuality of 0 (arrival on appointment time) and 10 (arrival 10 minutes before appointment time) showed a big difference. When patients arrive precisely on the appointment time, they still have to undress (in the out-patient case), which results in appointment lateness.

Adjusting arrival rate of urgent patients

The arrival rate of the urgent and emergent patients was assumed to follow a Poisson process with mean 2.7. In reality, sometimes the arrival rate is higher or lower. By varying this arrival rate the effect of more or less unscheduled arrivals can be simulated. Adjusting this arrival rate seemed to have no major influence on the schedule. This is relevant since the system is quite able to handle fluctuating arrival rates of urgent patients. Also, the expected waiting time for these urgent patients did not show a remarkable difference.

Adjustments in percentage of contrast needing patients

The percentage of IV-patients was estimated by the arrival rate of contrast needing patients and is adjusted by plus or minus 20%. Especially for appiontment lateness, an increase in contrast needing patients showed to affect the performance negatively. This has a close relation with adjusting the time spent in the SR. Contrast needing patients also need more time in the SR (for recovery).

The results are summarized in Figure 4.3. The average change in terms of percentage of the performance measures mean appointment lateness, mean overtime, morning and lunch break, and mean waiting time urgent patients is given for the five investigated schedules.

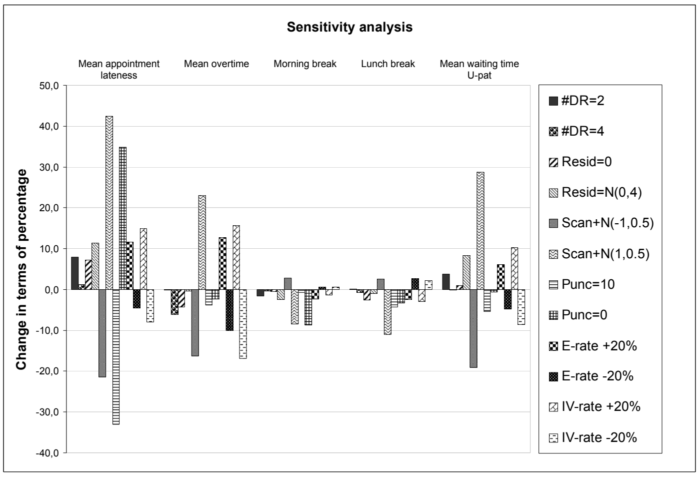


Figure 4.3: Sensitivity analysis overview

Chapter 5

Discussion of simulation results

In the above, different schedules have been simulated. What stands out is the impact of reallocating the installing of the intravenous access line procedure to a PR. Also, there is a difference between predefined schedules and the random schedules. This can have three possible reasons:

- The fixed schedules do not represent the used attribute distributions for patients in a good way.
- The arrivals are not balanced (random requests for scheduling).
- The patients are assumed to arrive at random, with high probability the number of more difficult patients is relative large.

The first situation was not the case since the same probabilities were used to construct the fixed schedules. The second and third situation can be compensated during scheduling patients. In practice it is for example possible to schedule an IV-patient on another day if the first day already has a lot of these slightly more difficult patients. Sensitivity analysis also showed significant different performances when arrival rate of contrast needing patients increases. Therefore these random schedules should really be treated as a way to estimate upper bounds for the performance measures.

The proposed schedules performed particularly well compared to the current schedule. Only a slight increase in appointment time and overtime seemed to be the result. For urgent patients the performance improved.
Overall the system seems capable of examining 29 scheduled patients on a working day. Currently an average of 21.7 patients (including urgent and emergent patients) are examined on a day. Subtracting the expected 2.7 patients from the emergency department, results in an average of 19 scheduled patients. Incorporating several improvement actions and using a new schedule increases capacity with about 50%.

The sensitivity analysis showed the effect of changes in the assumptions. The effect of changes in SR usage time are particularly of influence on the performance. This suggest the SR is the bottleneck in this process. Adding one minute to the average time spent in the SR, resulted in an average change of 42% in the appointment lateness. An average decrease of 1 minuted could lower the appointment lateness with 21%.

Changes in patients' punctuality resulted in changes of -33% and 35% in appointment lateness when patients arrived resp. 10 minutes or 0 minutes before their appointment time. Making sure patients arrive on time is therefore of great interest.
An increase of 20% in urgent patients arrival rate resulted in an average increase of 12% for the appointment lateness. Since the appointment lateness when the urgent patients arrival rate is at its normal level is about 7 minutes, an increase of 12% is not very much.

In reality it is also possible for the medical staff to take the breaks separately. Working with only one technologist decreases performance at that time, but when delays are foreseen, it might restore the flow in the process. Currently this adjustment is not an option, but when the need is high there is no choice to be more flexible.

In the next chapter, the quality improvement methodologies Lean Six Sigma and Theory of Constraints are shortly clarified. Since the CT scan project was conducted using the Lean Six Sigma methodology, Theory of Constraints is used to search for additional improvement actions.

Chapter 6

Improving the CT scan area with LSS and TOC

At Deventer Hospital the Lean Six Sigma (LSS) methodology was adopted to improve capacity. LSS is a combination of the methodologies Lean and Six Sigma. Six Sigma is an organizational and methodological framework originally developed by Motorola to derive continuous improvements in a organization. It uses a structured approach to improve business processes (which can be anything from manufacturing cell phones to handling insurance requests) with the use of improvement projects. Following the DMAIC steps (Define Measure Analyze Improve Control), the Green or Black belt (employee who has been educated in Six Sigma) tries to locate influence factors which prevent the process to function at its highest standards. By controlling these influence factors or limiting their impact, the variability in the process is minimized and causes of defects are removed. During the project several quality management tools were used, for example Project Charter, SIPOC, CTQ Flowdown, Gauge R&R, Pareto Chart, ANOVA, DOE and the Control Chart. These tools help the Green or Black belt to visualize the search for improvements. More information about Six Sigma can be found in the literature; [7] provides a Dutch introduction into this methodology and [17] gives the basics in English.

Lean or also called Lean Manufacturing is a slight adaption of the Toyota Production System. Toyota designed a production system based on low inventory, speed and flexibility. Processes have to be fast and disturbances should be removed. The customer is the main focus, non-value-adding work has to be eliminated. Other forms of waste like transportation, waiting, defects and overproduction should be removed. A Value Stream Map (VSM) is a tool to analyze the flow of for example raw materials or orders to the customer. By using a VSM waste can be visualized. Solutions like 5S (Sorting, Straighting, Sweeping, Standardizing and Sustaining), Poka-yoke (mistake proofing, preventing mistakes to occur) and JIT (Just-In-Time) are used to improve quality, eliminate waste, reduce cycle times and, the desired consequence, reduce costs.

Lean Six Sigma is the combination of both methodologies. It inherits the analyze and diagnostic approach as well as the organizational framework for continuous improvements from Six Sigma. The spirit of Lean Manufacturing is incorporated to get the mind set on eliminating waste as well as the sound tools to handle process flows. The reader is referred to the Dutch [8] and the English [12] for more information. In [26] the authors explain Lean Six Sigma in particular for service and health care.

During the Lean Six Sigma project at the radiology department, the different phases (DMAIC) were followed. By using the VSM it was evident that installing the intravenous access line could be removed from the SR. This resulted in an instant decrease of the time needed in the SR. It also became clear that the process was functioning under less stress than it could handle. The data showed that many more patients could be examined. Proving this by using data analysis is vital for a project to succeed.

Another quality improvement methodology is Theory of Constraints (TOC) [13]. Introduced by Dr. E.M. Goldratt, this theory suggests that in any system the flow is limited by a small number of constraints. These constraints are called the bottlenecks of the process. A process can be seen as several tubes through which a liquid flows. The smallest tube determines the overall speed and thus is the one limiting the throughput. The first step to improve the process is to locate these bottlenecks. Changes are made to make sure the bottleneck is removed (the tunnel is broaden). Then the next bottleneck is part of investigation, repeating this several times to improve the process.
In the CT scan case, the bottleneck was clearly the SR. This phase was the slowest (compared with checking-in, undressing, getting dressed) and therefore determined the throughput. A closer look at the uise of the SR results in several possible improvement actions. As already described, at first a scout is made to determine the position of the scan. One could argue that this is unnecessary when in some way it is possible to position the patient or just scan a wider area (but this has unwanted consequences). When contrast needs to be administered, this is pumped through the intravenous access line after the scout. Perhaps this is not necessary anymore when the contrast is taken for example orally before proceeding to the SR. For several in-patients it was necessary to get an opinion of a specialist if another scan needed to be made. If somehow this procedure can be diverted, just making the extra scan or invest in extra training for the technologists to make this decision by themselves, valuable time can be saved. Close proximity of a specialist and clear regulations can otherwise perhaps decrease the necessary time. Using multiple technologists might reduce the time needed in the SR because multiple tasks can be performed in parallel. For example while one technologist helps the previous patient back to his DR, the other prepares the scanner for the next patient.
It was shown that with minor adjustments the CT scan is perfectly capable to handle the demand. The way of planning the patients however determines also the throughput. Thus increasing the number of patients scheduled on a day should also be part of the investigation. TOC does not have the tools to handle this properly, it merely guides into the direction of improvements. Even when a schedule is obtained which is capable to let the SR run optimally, it is still possible that the demand is the bottleneck. If simply not enough patients need to be examined, the throughput can not be increased. Demand has to be increased by ways outside the scope of this thesis. But increased capacity probably also increases the demand for examinations [3].

The discussed improvement methods help to find improvement actions. Many processes can not be improved by queuing theory or simulations only. When the optimal layout of the system is known, queuing theory and simulations help the process manager to make the right decisions. Optimizing an already sub-optimal process is a waste of time, but much time has to be put in generating an appropriate model. Therefore it is advised to combine these calculation intensive techniques with the quality management methodologies to obtain a more efficient process while maintaining or improving quality.

Part II

Queues with a restriction on overtaking

Introduction of Part II

During the analysis of the CT scan routine as described in Part I of the thesis, an interesting property was found: patients are not allowed to overtake each other in the dressing room. Until now, queuing models allow overtaking and although a situation where overtaking in a CT scan routine could occur is unlikely, a model with overtaking could be seriously biased if overtaking is not desirable. The CT scan routine is not the only situation where this property is of interest. In data lines where packages are sent over several different lines, it is possible that a follow-up server needs to receive the packages in the same order as they were sent.

Queuing theory is a part of mathematics which analyzes queues. For example jobs, customers, orders, patients, or phone calls arrive in a system waiting to be served. For the remaining chapters of the thesis, the objects in the queuing systems are referred to as jobs. With the use of queuing theory one is able to calculate waiting times, job's time spent in the system and the optimal number of servers to meet certain requirements. These systems can be small, comprised of only one server like a small kiosk with only one cashier. The system can also be more complex. An example is checking in at an airport where several stages have to be proceeded (checking in, security check, boarding) with multiple servers and group arrivals.

Prior to analyzing queues where overtaking is prohibited, some basic queuing principles are presented in chapter 7. Then, a literature review is used to establish credibility for this research in chapter 8. Queuing systems with one or two stages and a non-overtaking property, are introduced in chapter 9. In the next chapter, the steady state probabilities are determined in a less complex case, for exponentially distributed arrival times and service times. Simulation methods are used to derive more general conclusions in chapter 10.

Chapter 7

Queuing theory framework

The very beginning of queuing theory is said to be an article by Erlang published in 1909 [10]. At this time Erlang worked for the telephone company in Copenhagen. His most important publication was one in 1917 [11], where he presents a formula to determine the probability a job finding a saturated system upon its arrival (Erlang's loss formula) and a formula to calculate the expected waiting time.
During the first years of queuing theory there was no uniform way to characterize a system. In 1953 Kendall introduced a notation to solve this problem [22]. The notation consists of three positions $A/B/C$.

- Position A: the arrival distribution. These arrivals follow a Poisson process ($A = M$), which represents random arrivals. The M stands for Markovian, referring to the underlying Markov process (a process in which the next situation depends only on the current situation). Other possibilities for the first entry in Kendall's notation are G (denoting a general arrival distribution), D (a deterministic arrival distribution, arrivals at fixed times) and E_k (an Erlang distribution with parameter k, which is used to model jobs consisting of multiple tasks).

- Position B: the service distribution. When the service times are exponentially distributed, this second entry will be an M, when deterministic a D and when generally distributed a G.

- Position C: the number of servers in the system. For example, $C = 2$ resembles a system with two servers in parallel.

Properties such as a finite waiting room, the population size (population where arrivals come from, which can affect the arrival rate) and queue discipline are mostly denoted by expanding Kendall's notation with additional parameters. For example, an M/M/2/3 system resembles a system with Poisson arrivals, exponentially distributed service times, two parallel servers and a finite waiting room with space for at most 3 jobs. Jobs arriving when 3 jobs are already waiting leave the system upon arrival (these jobs are not served).

In [24], Little introduced a formula describing a relation between the arrival rate (average number of jobs arriving during a specified time period, denoted with λ), average number of jobs in the system ($\mathbb{E}[L]$) and the average total time spent in the system ($\mathbb{E}[S]$). This relation is given by $\mathbb{E}[L] = \lambda\mathbb{E}[S]$.
Wolff popularized and gave the first rigorous prove that Poisson arrivals see time averages (PASTA). This property states that in an $M/./.$ system the fraction of time the system is in a specific situation (called a state), is equal to the fraction of

jobs finding the system in that situation. With the PASTA property performance measures can be obtained.

This chapter introduces notations and theorems for queuing theory which are needed to investigate more complex queuing systems. The standard tandem queue is introduced in the next section and is used to show some basic calculations.

7.1 The tandem queue

A type of queue one often encounters in practice is the tandem queue. This queue is a representation of a system with multiple servers placed in series. In the tandem queue, as depicted in 7.1, jobs arrive according to a certain distribution. A job is served in stage 1 by a server S_1. If on arrival, this server is busy and thus the job can't be served directly, it takes place in the first queue (Q_1). The job waits for server S_1 to finish the service of all jobs which arrived earlier. A total waiting time of W_1 (≥ 0) is needed before the job can be served. The service time in the first stage is defined by B_1, which is distributed by a given non-negative distribution. After completing its service in stage 1, the job proceeds to the next stage. If the server in stage 2 is empty it can be served immediately, otherwise it joins the queue Q_2 where it stays a waiting time of W_2. The service time B_2 is needed to finish the service of the job after which it leaves the system. The total time the job is in the system: $W_1 + B_1 + W_2 + B_2$, is denoted by S and is called the sojourn time (or throughput time). For the standard case, the service times are assumed to be independent.

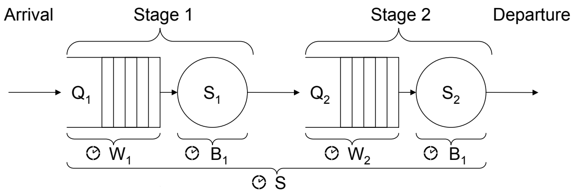


Figure 7.1: The standard tandem queue with two stages

To follow the notation used by Pinedo and Wolff [31], this system can be described by M/M/1→M/1. Stages are separated by arrows and the service situation in each stage is characterized by Kendall's notation. The second stage has only two positions since the arrival distribution is fixed by the departure distribution of the previous stage. Some calculations for the case where the arrivals are a Poisson process with arrival rate λ and the service times at both servers are exponential distributed with mean $\frac{1}{\mu}$ will be shown.

The system can be seen as a collection of states (situations) between which the system alternates over time. In this tandem queue case, a state consists of two entries (n_1, n_2); n_1 denotes the number of jobs in stage 1 and n_2 the number of jobs in stage 2. State (3,1) is therefore the situation in which there are three jobs in stage 1 (two in the queue Q_1 and one job in service at server S_1) and one in stage 2 (in service at server S_2). The state space can be described by $N = \{(n_1, n_2) | n_1 = 0, 1, \ldots, \infty,\ n_2 = 0, 1, \ldots, \infty\}$.
In queuing theory, exact calculations or simulations are used to determine the limiting behavior of a system, since even in the simple M/M/1 case the time-dependent behavior leads to very difficult state probabilities (see [1] p.101). This suggest that explicit solutions for more general situations (that is, time dependent) are often

impossible to obtain.
In order to introduce Markov characteristics in popular queuing models, some theorems from Markov chains theory are introduced in the next section.

7.2 Markov theory

Let the state space, defined by N, be an infinite or countable set containing the possible states. Consider a continuous time stochastic process $\{X(t),\ t \geq 0\}$ which takes values form N. This stochastic process is called a continuous time Markov chain if for all $t, s \geq 0$ and states $(i, j) \in N$ it has the following equality:

$$P(X(t+s) = j|X(s) = i, X(u) = x(u), 0 \leq u < s) = P(X(t+s) = j|X(s) = i).$$

This equality is called the Markovian property and it states that the conditional distribution of the future state at time $t + s$, given the present state (at time s) and the states before time s, depends only on the present state. In the special case where the distribution is also independent of s, the Markov chain is said to have stationary homogeneous transition probabilities. In that case, the probability of going from one state i to a state j does not change over time. This type of Markov chains will be considered further on.
The states in the state space can have several characteristics. States i and j are said to *communicate* if it is possible to reach the other state (possibly by passing states other than i and j). If all the states communicate with each other, the Markov chain is called *irreducible*. A state is called *transient* if the probability of ever returning to this state equals one. If this is not the case, the state is called *recurrent*. If the expected time of returning to state i is finite, state i is *positive recurrent*. If this is not the case the state is said to be *null recurrent*.

The transition behavior in a continuous Markov chain is captured in the matrix $\mathbf{Q}$. The entry q_{ij} in $\mathbf{Q}$, denotes the transition rate from departing state i to state j. In state i a transition takes place after an exponential time with parameter $\sum_{j \neq i} q_{ij}$. The system makes a transition from i to j with probability $p_{ij} = q_{ij} / \sum_{k \neq i} q_{ik}$. By defining $q_{ii} = -\sum_{j \neq i} q_{ij}$, the rate out of i, the matrix $\mathbf{Q}$ is called the generator of the continuous time Markov chain.

Under certain conditions, the generator $\mathbf{Q}$ can be used to derive a limiting distribution (stationary). This distribution determines the average time the system is in a specific situation. The limiting distribution is used to calculate the performance measures. An important theorem from Markov chain theory proves the existence of a unique stationary distribution.

Theorem 7.2.1 *If the following conditions are met*

- *the Markov chain is irreducible*
- *the Markov chain is positive recurrent,*

there exists a unique stationary distribution π.

Proof See p175-177 in [35].

The probability that the system is in state i converges to π_i when $t \to \infty$ and the conditions of Theorem 7.2.1 are fulfilled. These limiting probabilities can be calculated using the balance equations. Balance equations balance the number of transitions into a state and out of a state. The transitions from state i to state j occur at rate q_{ij} and therefore the number of transitions from i to j per time unit equals $\pi_i q_{ij}$. The number of transitions out of state i per time unit equals $\pi_i \sum_{j \neq i} q_{ij}$. The number of transitions per time unit into state i equals $\sum_{j \neq i} \pi_j q_{ji}$ for $i, j \in I$. These equations can be simplified to the equality $0 = \pi \mathbf{Q}$. Using normalization to acquire the limiting probabilities results in a unique stationary distribution. More information about Markov chains can be found in the literature [1], [30] and [35].

Exponentially distributed arrival times and service times are needed to have constant transition rates. Since the exponential distribution has the memoryless property, the rate of going from state i to state j does not change overtime. If this rate would change over time (which occurs with other arrival time and service time distributions), it is impossible to derive equations which balance the flow into and out states (the balance equations). In equilibrium (steady state), the flow out of a state equals the flow into the state. For some distribution, a change in state space description is possible to obtain constant transition rates. For the Erlang distribution it is possible to divide the service time in several phases. Instead of using the number of jobs in the system, the state space uses the number of phases (a job consists of several phases) in the system. Because the Erlang distribution is the sum of several exponential distributions, the service time of a phase is exponentially distributed and constant transition rates are obtained. Unfortunately, for almost every other distribution, such a change in state space is impossible.
The next section shows how to use the balance equations to derive a unique stationary distribution for the tandem queue.

7.3 Steady state probabilities

In order to obtain the balance equations, the transition rates between the states need to be derived. Assuming the system is in state (0,0), the only possible transition is an arrival of a job (since there are no jobs in the system it is impossible for the next event to be a service completion). These arrivals follow a Poisson process with rate λ and therefore $q_{(0,0)(1,0)} = \lambda$. The only way the system can return to state (0,0) is by going through state (0,1). The rate at which this transition occurs is equal to μ (the service rate). For state (0,0) the balance equation

$$\lambda p_{(0,0)} = \mu p_{(0,1)}$$

is acquired. For the other states balance equations are obtained in a similar way. Resulting in the following system of equations:

$$\begin{aligned}
\lambda p_{(0,0)} &= \mu p_{(0,1)} \\
(\lambda + \mu) p_{(n_1,0)} &= \lambda p_{(n_1-1,0)} + \mu p_{(n_1,1)} \quad n_1 > 0 \\
(\lambda + \mu) p_{(0,n_2)} &= \mu p_{(1,n_2-1)} + \mu p_{(0,n_2+1)} \quad n_2 > 0 \\
(\lambda + \mu + \mu) p_{(n_1,n_2)} &= \lambda p_{(n_1-1,n_2)} + \mu p_{(n_1+1,n_2)} + \mu p_{(n_1,n_2+1)} \quad n_1, n_2 > 0 \\
\sum_{n_1 \leq 0, n_2 \leq 0} p_{(n_1,n_2)} &= 1.
\end{aligned}$$

Since the corresponding Markov chain is irreducible (every state can be reached from any other) and positive recurrent, assuming the occupancy is less than one (demand is less than, otherwise the number of patients in the system will go to infinity, theorem 7.2.1 proves the existence of a unique solution to these equations.

Solving these equations eventually leads to the following formula:

$$p_{(n_1,n_2)} = \left(\frac{\lambda}{\mu}\right)^{n_1+n_2} \left(1 - \frac{\lambda}{\mu}\right)^2 .$$

With the steady state probabilities, the mean performance characteristics can easily be found. The arrivals follow a Poisson process, the PASTA property applies. The stationary distribution gives therefore the probability of a job arriving in a specific state. In the next section, several performance measures are determined for the standard tandem queue.

7.4 Steady state performance measures

The number of jobs in the system can be obtained from the steady state probabilities when there is a clear correspondence between the states and the number of jobs in the system. The expected number of jobs in the system can then be calculated by multiplying the number of jobs in the system with its corresponding probability. In case of the standard tandem queue, this is just $\mathbb{E}[L] = \sum_{n_1,n_2=0}^{\infty} \left(1 - \frac{\lambda}{\mu}\right)^2 \left(\frac{\lambda}{\mu}\right)^{n_1+n_2} = \frac{2\lambda}{\mu-\lambda}$.

Using Little's formula ($\lambda\mathbb{E}[S] = \mathbb{E}[L]$), the expected waiting time is calculated. For the standard tandem queue, the expected time in the system equals $\frac{2}{\mu-\lambda}$. When the sojourn time in the queue is known, the expected waiting time is calculated easily. The total time spent in the system equals the sum of the expected waiting time in the queue and the expected service time. For the standard tandem queue this is just $\frac{1}{\mu}$, so the expected waiting time in the queue equals $\frac{\mu+\lambda}{\mu(\mu-\lambda)}$. Applying Little's formula in the queue (the waiting line) the expected number of customers in the queue can be calculated ($\mathbb{E}[L_q] = \lambda\mathbb{E}[W]$). Other performance measures are derived in similar ways.

The following chapters focus on a specific situation in which jobs are not allowed to overtake each other in the system. A non-overtaking property is introduced and its effect is investigated.

Chapter 8

Non-overtaking queues in literature

In literature, overtaking in queues has been investigated [4], [41], [45]. Overtaking is said to occur when a job A arrives before a job B, but job B departs first [45]. This property is only treated in queuing networks (such as the network shown in 8.1). In a single server queue with the FCFS discipline, overtaking is impossible. A queuing network consists of several queues with the FCFS discipline. If each sub-system (a queue and a server) has the FCFS discipline, exponentially distributed service times, Poisson arrivals, possible job departures after each service (when a job is served, with a certain probability it leaves the system, otherwise it joins another sub-system), and the occupancy is below 1, the network is said to be a Jackson network [21]. In these networks it might be possible to overtake.

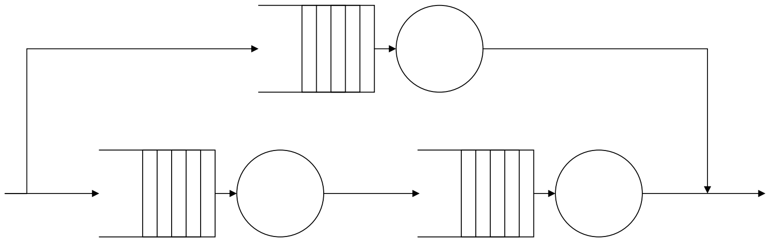


Figure 8.1: Example of a queuing network where overtaking is possible

It has been shown that the more overtaking takes place, the more correlated the sojourn times are [45]. Walrand and Varaiya [41] show that in any open Jacksonian network, the sojourn times of jobs at the various nodes of a non-overtaking path are all mutually independent. The sojourn time distribution at these nodes is known and therefore it is easy to calculate the sojourn time of jobs on these non-overtaking paths.

Literature which introduces non-overtaking as a queue property has not been found. This property is introduced in the next chapter.

Chapter 9

Queues with overtaking restrictions

Before investigating queuing systems with overtaking restrictions, a formal definition to determine non-overtaking is introduced.

Definition Let T_i be the time job i commences service and let B_i be its service time. If $\exists j \in \mathbb{N}_{>0}$ such that $T_i + B_i > T_{i+j} + B_{i+j}$, job $i+j$ blocks the server until jobs $i, \ldots, i+j-1$ have finished service, the system is said to have the *non-overtaking property.*

Note that in this definition, every single server queuing system has the non-overtaking property. For systems with multiple servers (or processor sharing servers) this is interesting.
Whenever a job has to wait in a stage (due to no room in the next stage or the non-overtaking property) is gets delayed.

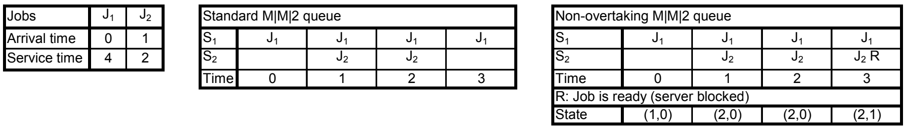
Definition Let T_i be the time job i commences service, B_i the service time and Y_i the time job i departs from the stage. The *delay* that job i (D_i) obtains is given by: $D_i = \max(0, Y_i - T_i - B_i)$.

The definition of overtaking is different from the definition used in queuing networks. It can be said that the overtaking definition proposed in this thesis is a inner-queue property, while the definition used in queuing networks is a inner-network property. Since jobs can not overtake other jobs by taking a different route (which was possible in the Jackson network, the dynamics will probably be quite different. The property is investigated for a wide variety of queuing systems, starting with a single stage queue in the next section.

9.1 Single stage queues

Single stage queues can consist of multiple servers in parallel. Normally, it is possible to overtake a previous job in parallel systems. Figure 9.1 gives an example of two queuing systems, one with the possibility of overtaking and one without this possibility.
Two jobs are considered, job 1 arrives at time 0 and has a service time of 4, job 2 arrives at time 1 and has a service time of 1. In the standard M/M/2 queue, job 1 is served upon arrival at time 0, job 2 starts service at the second server at time 1. At time 3, job 2 has finished its service and leaves the system while job 1

is still in service at the other server. Job 1 departs at time 4 (after completing its service requirement of 4 time units). In the non-overtaking M/M/2 queue, when job 2 has completed its service at time 3, it is not allowed to leave the system. The non-overtaking property makes sure that job 1 leaves the system before job 2. Therefore, job 2 has to occupy the second server for one time unit job 1 has completed its service. Both jobs leave the system at time 4.



Jobs	J_1	J_2
Arrival time	0	1
Service time	4	2

Standard M|M|2 queue				
S_1	J_1	J_1	J_1	J_1
S_2		J_2	J_2	
Time	0	1	2	3

Non-overtaking M|M|2 queue				
S_1	J_1	J_1	J_1	J_1
S_2		J_2	J_2	J_2 R
Time	0	1	2	3
R: Job is ready (server blocked)				
State	(1,0)	(2,0)	(2,0)	(2,1)

Figure 9.1: Example of a system where overtaking is allowed and one with the non-overtaking property

The first non-overtaking system of interest is one with two servers placed in parallel and no queue in front of them (see Figure 9.2). Jobs which arrive while the system is fully occupied are blocked. Starting with this system gives more insight in the performance of non-overtaking systems.
Removing the queue results in a system which has a finite state space. Otherwise it is possible to have an arbitrary large number of jobs in the queue. Limitations on the queue size results in less complex calculations for the steady state probabilities. When investigating these systems with a different maximum number of jobs waiting (different queue sizes), the performance of the queuing system with an infinite waiting room can be approximated.

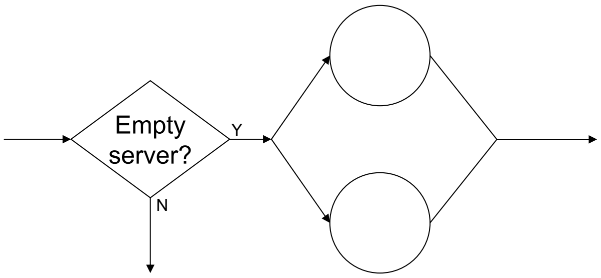


Figure 9.2: Single stage, two server queue with blocking

In this situation, the order of arrivals needs to be incorporated. A job can only depart from a system when all previous jobs departed before it. The state space of this system can be defined by $N = \{(n,r)|n = \{0,1,2\}\ r = \{0,1\}\}$, where n equals the number of jobs in the system (0, 1 or 2 because more jobs are impossible due to blocking) and r equals 1 if the last job is being delayed (and 0 when this is not the case). The balance equations result in linear equations which can be solved easily:

$$\begin{aligned}
\lambda p_{0,0} &= \mu p_{1,0} + \mu p_{2,r} \\
(\lambda + \mu) p_{1,0} &= \lambda p_{0,0} + \mu p_{2,0} \\
2\mu p_{2,0} &= \lambda p_{1,0} \\
\mu p_{2,1} &= \mu p_{2,0}.
\end{aligned}$$

Solving these equations results in the following steady state probabilities:

$$\begin{aligned}
p_{0,0} &= \frac{\mu(\lambda+2\mu)}{3\lambda\mu+2\mu^2+2\lambda^2}\\
p_{1,0} &= \frac{2\lambda\mu}{3\lambda\mu+2\mu^2+2\lambda^2}\\
p_{2,0} &= \frac{\lambda^2}{3\lambda\mu+2\mu^2+2\lambda^2}\\
p_{2,1} &= \frac{\lambda^2}{3\lambda\mu+2\mu^2+2\lambda^2}.
\end{aligned}$$

Given the steady state probabilities above, the performance measures can be obtained. In equilibrium the fraction of time the system is in state $(0,0)$, equals the corresponding steady state probability (PASTA). Taking the arrival rate arbitrary equal to one[1], the following formulas for the performance characteristics are found:

$$\begin{aligned}
\mathbb{P}(\text{job blocked on arrival}) = \mathbb{P}(\text{arrival in } p_{2,0} \text{ or } p_{2,1}) &= \frac{2}{2\mu^2+3\mu+2}\\
\lambda_{adjusted} = \lambda(1-\mathbb{P}(\text{job blocked on arrival})) &= \frac{\mu\,(3+2\,\mu)}{2\mu^2+3\mu+2}\\
\mathbb{E}[L] &= \frac{2(2\mu+3)}{2\mu^2+3\mu+2}\\
\mathbb{E}[S] &= \frac{2(\mu+2)}{\mu(3+2\mu)}.
\end{aligned}$$

The probability for an arbitrary job being blocked equals the probability of a job-arrival in a situation where it can not be served directly. This situation corresponds to the states $(2,0)$ and $(2,1)$. By using the PASTA property, the probability of being blocked (and thus leaving the system upon arrival) equals $p_{(2,0)} + p_{(2,0)}$.
Since some jobs are blocked, these do not effect the system and servers do not see these jobs arriving. Therefore, the arrival rate of jobs which are served ($\lambda_{adjusted}$) is less than the actual arrival rate and can be calculated by multiplying the actual arrival rate (λ) with the probability of entering the system ($1-\mathbb{P}$(job blocked on arrival)). The expected number of jobs in the system is calculated as in section 7.4. Applying Little's formula with the adjusted arrival rate and expected number of jobs in the system, results in the expected sojourn time.
In the standard M/M/2/0 queue (the zero notation suggest the absence of a queue) jobs never have to wait. Whenever a job arrives and the system is fully occupied (both servers are non-idle), the job leaves immediately. Therefore the expected sojourn time is simply equal to the expected service time, $\frac{1}{\mu}$. In the system with the restriction on overtaking the expected sojourn time is $\frac{2\mu+4}{2\mu+3}$ times larger then the standard case.

The same calculations have been done for systems with room for of 2, 4, 10 or 20 waiting jobs. When the number of jobs allowed to wait increases, the expected sojourn time seems to converge, see Figure 9.3.

[1] When the actual arrival rate equals 60 jobs per hour and the service rate equals 90 jobs per hour, a change in timescale from hours to minutes is used to obtain an arrival rate of 1 and a service rate of 1.5 per minute. By adjusting the time scale, it is always possible to obtain an arrival rate of 1.

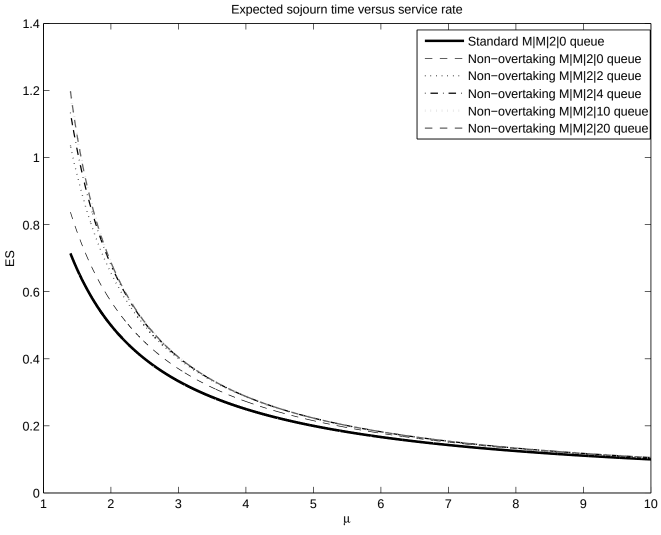


Figure 9.3: Expected sojourn time for different queue sizes.

In the next section, the multistage queue with a possible infinite queue length is estimated. Other performance measures are obtained from that queuing system.

9.2 Two server parallel queue without overtaking

When the blocking property is removed from the parallel server queue investigated in the previous section, the state space can be written as $N = \{(n,r)|n = \{0,1,2\ldots\},\ r = \{0,1\}\}$ with n the number of jobs in the system and $r = 0$ when no job has to wait on a previous job due to the non-overtaking property. If a job is delayed, r will equal 1. The balance equations are stated below.

$$\begin{aligned}
\lambda p_{0,0} &= \mu p_{1,0} + \mu p_{2,1} \\
(\lambda+\mu) p_{1,0} &= \lambda p_{0,0} + \mu p_{2,0} + \mu p_{3,1} \\
(\lambda+2\mu) p_{i,0} &= \lambda p_{i-1,0} + \mu p_{i+1,0} + \mu p_{i+2,1}\ (i \geq 2) \\
(\lambda+\mu) p_{2,1} &= \mu p_{2,0} \\
(\lambda+\mu) p_{i,1} &= \lambda p_{i-1,1} + \mu p_{i,0}\ (i \geq 3)
\end{aligned}$$

At first, these equations seem not to be very complex. But the interaction between the two recursive balance equations makes sure that to calculate steady state probability the previous steady state probabilities need to be known. Using substitution seemed useless since no terms cancel out. In order to solve the balance equations, an iterative method (the Gauss-Seidel method) is used as in [16].

The Gauss-Seidel method is used to approximate the steady state probabilities. This method is an iterative method to solve linear systems of equations and is a special case of the successive over-relaxation iterative method [39]. the steady state probabilities can be calculated quite accurately, despite the fact that the Gauss-Seidel method solves a finite set of linear equations, by allowing a huge number of jobs waiting in the queue. To use the Gauss-Seidel method, each steady state equation was divided by the coefficients (in terms of λ and μ) on the left hand side. The probability on the left hand side is called p^{new}. At the beginning, starting values for the probabilities are chosen (not equal to zero). At each iteration, the new steady state probabilities are calculated. When using only the old probabilities, the iterative method is called the Jacobi method. When calculating the new probabilities with the previous calculated probabilities and the old probabilities (for those states of which no new probability has been calculated yet), the method is called the Gauss-Seidel method. Using the new probabilities to calculate other new probabilities, the method will converge faster to the steady state probabilities. The intuitive explanation for this improvement in convergence rate, is the fact that the new probabilities are closer to the steady state probabilities than the old probabilities. A convergence criterion is used to determine whether approximated probabilities have converged sufficiently. In Procedure 1, the algorithm is presented.

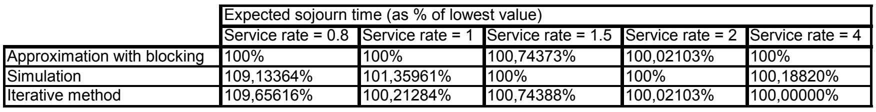
Procedure 1 : Approximation of the steady state probabilities

```
Step 1: Assigning values to $p_{i,j}^{old}$

        $SETp_{0,0}^{old} = 0.5$
           $p_{2,1}^{old} = 0.5$
        $SET\ maxN = 10000$ (max number of jobs in system)
        $FOR\ 1 \leq j \leq maxN$
        $SET\ p_{j,0} = 0.5 * p_{j-1,0}$
        $FOR\ 3 \leq j \leq maxN$
        $SET\ p_{j,1} = 0.5 * p_{j-1,1}$
        $SET\ \epsilon = 10^{-8}$ tolerance value
        $SET\ tol1 = 10$
        $SET\ tol2 = 1$

Step 2: Calculate $p_{i,j}^{new}$ using the most recently calculated probabilities

        $WHILE(tol1 > tol2)$
        $p_{0,0}^{new} = (\mu p_{1,0}^{old} + \mu p_{2,1}^{old})/\lambda$
        $p_{1,0}^{new} = (\lambda p_{0,0}^{new} + \mu p_{2,0}^{old} + \mu p_{3,1}^{old})/(\lambda + \mu)$
        $FOR\ 4 \leq j \leq maxN - 1$
        $p_{j,0}^{new} = (\lambda p_{j-1,0}^{new} + \mu p_{j+1,0}^{old} + \mu p_{j,1}^{old})/(\lambda + 2\mu)$
        $p_{2,1}^{new} = (\mu p_{2,0}^{new})/(\lambda + \mu)$
        $FOR\ 3 \leq j \leq maxN - 2$
        $p_{j,1}^{new} = (\lambda p_{j-1,1}^{new} + \mu p_{j+2,0}^{old})/(\lambda + \mu)$

Step 3: Check stopping criterion

        $tol1 = \sum_{i=0,j=2}^{i=maxN,j=1} |p_{i,j}^{new} - p_{i,j}^{old}|$
        $tol2 = \epsilon * \sum_{i=0,j=2}^{i=maxN,j=1} |p_{i,j}^{new}|$
        $IF\ tol1 \leq tol2$
        $STOP\ WHILE\ LOOP$
        $ELSE\ p_{i,j}^{old} = p_{i,j}^{new}$
```

Step 4: Normalize the p^{new} values

$$p^{new}_{i,j} = \frac{p^{new}_{i,j}}{\sum_{i=0,j=0}^{i=maxN,j=1} p^{new}_{i,j}}$$

The arrival rate is set arbitrary to one. With the iterative method, the steady state probabilities are obtained for different service rates. In Figure 9.4, the expected sojourn time which were obtained by investigating the system with blocking, simulations and the Gauss-Seidel iterative method are shown. The expected sojourn times are represented as percentages of the lowest value (this value is set to 100%). Percentual differences are of more importance than the actual values when comparing estimated performance measueres for different methods. For different service rates, the expected sojourn time is estimated using the blocked queuing system, simulations, and the Gauss-Seidel method (iterative method). At low service rates (corresponding with high occupancy), the approximations from the simulation and iterative method are about 10% higher than the approximation from the blocked queuing system. This is probabily caused by a higher probability of being blocked, which results in less jobs being served. The expected sojourn time is therefore underestimated. When the service rate increases, the differences decrease. This suggest that for low occupancy systems, the blocked queuing system with atmost 20 jobs in the queue, results in reliable approximation. For this system with blocking, the performance characteristics can be calculated exactly, however, these are too complex to be usable or to be presented in the thesis.



	Expected sojourn time (as % of lowest value)				
	Service rate = 0.8	Service rate = 1	Service rate = 1.5	Service rate = 2	Service rate = 4
Approximation with blocking	100%	100%	100,74373%	100,02103%	100%
Simulation	109,13364%	101,35961%	100%	100%	100,18820%
Iterative method	109,65616%	100,21284%	100,74388%	100,02103%	100,00000%

Figure 9.4: Estimation of the expected sojourn time using different methods

The approximation of the blocked queuing system is close too the other methods and this suggests that using the calculated formulas of the performance measures in the case of at most 20 waiting jobs, would be a good approximation when the occupancy is not too high. These formulas however are too complex to be usable or to be presented in the thesis.

For different service rates, the expected sojourn time is estimated. Since the arrival rate was chosen equal to one, the expected sojourn time equals the expected number of jobs in the system (Little's law).

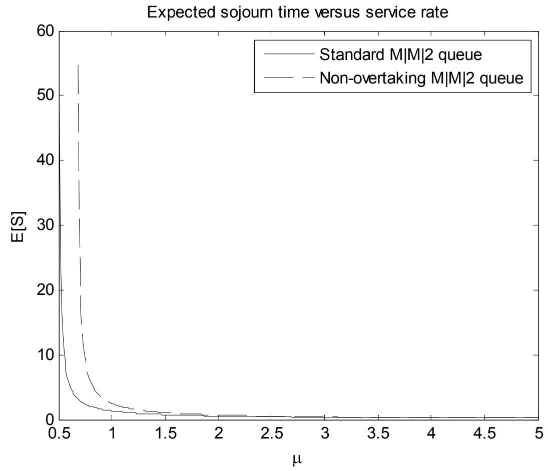


Figure 9.5: Expected sojourn time versus service rate (M/M/2 queuing systems)

In Figure 9.5, the difference is shown between the normal two server queue and the non-overtaking two server queue. For high service rates (μ), corresponding to a low occupancy, the expected sojourn times for the different queues do not deviate much. In fact, when the occupancy decreases, the expected sojourn time of the non-overtaking queue converges to the expected sojourn time of the standard queue. This convergence is due to the decrease in possible delay. Jobs are are less likely to arrive in a system which is already serving another job. Therefore, the probability a job is delayed is smaller. When the service rates increase, the probability for a job to be delayed due to the non-overtaking property decrease. Therefore, the expected delay for a job decreases to zero, resulting in equal sojourn times for both queues.

Figure 9.5 shows a much higher sojourn time at low service rates for the non-overtaking queue. When the probability a job finds another job in the system increases, the probability of delay also increases. This result in a higher expected sojourn time, especially at low service rates (high occupancy). The graph might suggest the same relation between occupancy and expected sojourn time for both systems, this would mean that when the occupancies are equal, the expected sojourn times would be equal. However, graph 9.6 shows that this is not the case.

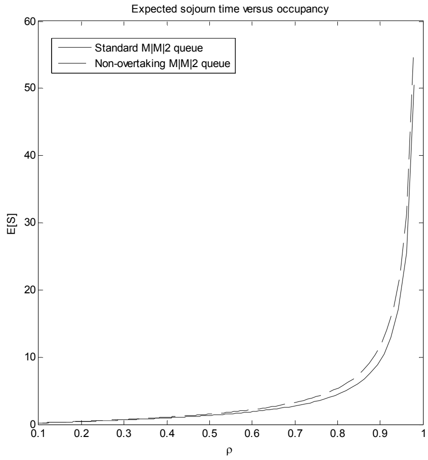


Figure 9.6: Expected sojourn time versus occupancy (M/M/2 queuing systems)

The occupancy can be calculated when the expected service time, the number of servers and the service rate are known. The expected sojourn time and the expected waiting time are obtained using the steady state probabilities by applying Little's formula on the expected number of jobs in the system and queue length. The time a job spends in service equals its actual service requirement plus a possible delay. This expected service time equals the expected sojourn time minus the expected waiting time. Since the arrival rate is known, the occupancy of the non-overtaking system equals $\lambda(\mathbb{E}[S] - \mathbb{E}[W])/2$. The occupancy in the standard M/M/2 queue is equal to $\lambda/2\mu(= \lambda\mathbb{E}[B]/2)$. Figure 9.7 shows the occupancy of the systems for different service rates.

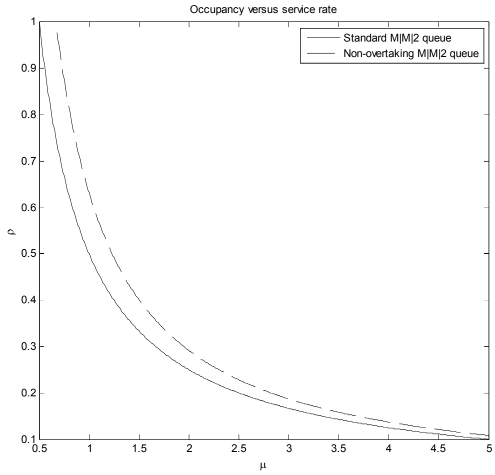


Figure 9.7: Occupancy versus service rate (M/M/2 queuing systems)

The probability a job needs to wait upon arrival is a commonly used performance measure. In call centers for example, the waiting probability is used to find the appropriate number of servers if a condition like '80% of the calls need to be answered immediately' needs to be satisfied. By calculating this probability for systems with different numbers of servers, the optimal number can be found. The waiting probability equals the probabilty of a job arriving in a state in which it has to wait.

$$\Pi_W = 1 - p_{(0,0)} - p_{(1,0)}$$

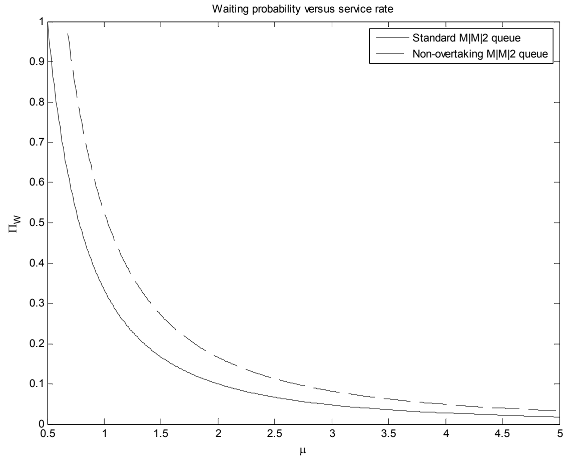


Figure 9.8: Waiting probability (M/M/2 queuing systems)

Figure 9.8 shows the waiting probability for different μ's for both systems.
The expected delay an arbitrary job obtains can be calculated by subtracting the service requirement $(1/\mu)$ and the expected waiting time from the expected sojourn time. Figure 9.9 shows the expected delay for different service rates. It is seen that in a system with two servers, both serving with rate 1 job per hour and an arrival rate of also 1 job per hour, the expected delay of a job is about 15 minutes. In the standard M/M/2 queue, this would not occur. Therefore, in systems with the overtaking restriction, the impact of this restriction should not be underestimated.

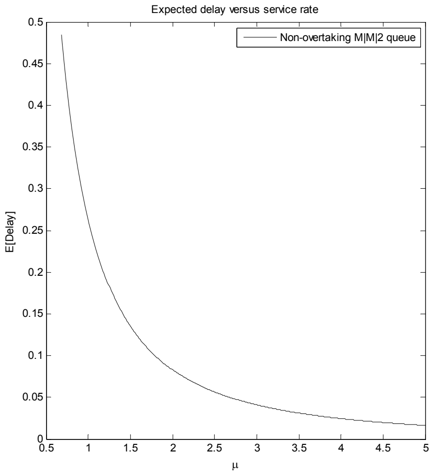


Figure 9.9: Expected delay versus service rate (M/M/2 non-overtaking queuing system)

In this section, the non-overtaking property has been analyzed for single stage queuing systems. In the next section, a special tandem queue is also investigated with the non-overtaking property.

9.3 Tandem queue with non-overtaking and blocking

The tandem queue with two identical servers in the first stage, only one server in the second stage and no queues in front of the servers (blocking) is extended with the non-overtaking restriction (see Figure 9.10 for a graphical representation of the queue). Jobs are still served according the FCFS principle and overtaking is prohibited. Arrivals are distributed according to a Poisson process with rate λ and the service times are exponentially distributed with parameter μ.

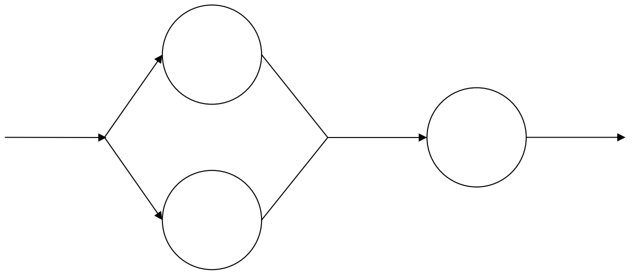


Figure 9.10: Graphical representation of the tandem queue

The state space can be defined by $N = \{(i,j,k)|i = \{0,1,r\},\ j = \{0,1,r\},\ k = \{0,1\}\}$. Where i and j resemble the situation at the first server and second server in stage one. The third entry, k, resembles the situation at the server in the second stage. If i, j or k equals 0, the relevant server is idle and if i, j or k equals 1, the server is busy. If i equals r, a job in the first stage is delayed because the second server is still busy. If j equals r, a job in the first stage is delayed due to another job in the first stage (a job arriving first in stage 1 is still in service). This state space uses a different notation as the state space used in the single stage queue. It is impossible to use a description for the number of jobs in the first stage (n_1), second stage (n_2) and jobs being delayed (r). For example, in state $(n_1, r, n_2) = (2, 1, 1)$ there are three jobs, one in stage 2 (job A) and two in stage 1 (B and C). One of the jobs in stage 1 has already finished service, but this state description can not distinguish which job (B or C) started service first. This is of interest, because when job B arrived before job C, job B needs to start service at the second stage before job C (due to the non-overtaking restriction).

This system has the advantage that it has a finite state space. The states clearly communicate with each other (it is possible to reach any state in a number of steps from any other state), the system is irreducible. Since the expected time to return to a state is clearly finite, the system is positive recurrent. This results in the existence of a unique stationary distribution by Theorem 7.2.1.
Writing out the balance equations and normalizing eventually leads to the following steady-state probabilities:

$$\begin{aligned}
p_{(0,0,0)} &= \frac{8\mu^3}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}\\
p_{(1,0,0)} &= \frac{2\,(3\,\lambda + 4\,\mu)\,\lambda\mu^2}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}\\
p_{(1,1,0)} &= \frac{\lambda^2\left(\lambda^2 + 4\,\lambda\mu + 4\,\mu^2\right)}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}\\
p_{(0,0,1)} &= \frac{8\lambda\mu^2}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}\\
p_{(1,0,1)} &= \frac{6\lambda^2\mu}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}\\
p_{(1,1,1)} &= \frac{2\lambda^3}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}\\
p_{(1,r,0)} &= \frac{\lambda^2\left(2\,\lambda^2 + 5\,\lambda\mu + 4\,\mu^2\right)}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}\\
p_{(r,0,1)} &= \frac{2\lambda^2\mu\,(4\,\lambda + 5\,\mu)}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}\\
p_{(r,1,1)} &= \frac{\lambda^3\,(5\,\lambda + 6\,\mu)}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}\\
p_{(1,r,1)} &= \frac{\lambda^3}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}\\
p_{(r,r,1)} &= \frac{\lambda^3\,(6\,\lambda + 7\,\mu)}{(17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3)\,(\lambda + \mu)}.
\end{aligned}$$

The probability a job is blocked equals according to PASTA, the probability that it arrives when the system is in state (1,1,0), (1,1,1), (1,r,0), (r,1,1), (1,r,1) or (r,r,1). This equals

$$P(\text{job is blocked}) = \frac{(17\,\lambda + 8\,\mu)\,\lambda^2}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}.$$

The arrival rate has to be adjusted for the possibility of blocking jobs, the servers see less jobs than there really were. The overall arrival rate equals λ but with $\mathbb{P}$(job is blocked) an arriving job is blocked [39]. Therefore the arrival rate the system sees equals

$$\lambda_{adj} = \lambda * (1 - P(\text{job is blocked})) = \frac{2\lambda\mu\left(7\,\lambda^2 + 8\,\lambda\mu + 4\,\mu^2\right)}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}.$$

The number of jobs in the system is calculated by multiplying a number of jobs in the system and its steady state probability.

$$\begin{aligned}
\mathbb{E}(L) = \quad & 0 * p_{(0,0,0)} + 1 * (p_{(1,0,0)} + p_{(0,0,1)}) +\\
& 2 * (p_{(1,1,0)} + p_{(1,0,1))} + p_{(1,r,0)} + p_{r,0,1)}) +\\
& 3 * (p_{(1,1,1)} + p_{(r,1,1)} + p_{(1,r,1)} + p_{(r,r,1)})
\end{aligned}$$

$$\mathbb{E}(L) = \frac{2\left(24\,\lambda^2 + 23\,\lambda\mu + 8\,\mu^2\right)\lambda}{17\,\lambda^3 + 22\,\lambda^2\mu + 16\,\lambda\mu^2 + 8\,\mu^3}$$

Using Little's formula, $\mathbb{E}(S)$ is obtained:

$$\mathbb{E}(S) = \frac{1}{\lambda_{adj}}\mathbb{E}(L) = \frac{24\,\lambda^2 + 23\,\lambda\mu + 8\,\mu^2}{\mu\,(7\,\lambda^2 + 8\,\lambda\mu + 4\,\mu^2)}.$$

In this system it is impossible for a job to have a waiting time since jobs leave on arrival when they are not directly served. Therefore, the sojourn time of a job consists of its service times in the two stages and its delay (D) between stage 1 and 2. The expected service time of a job equals $2 * 1/\mu$, subtracting this from the sojourn time results in the expected delay.

$$\begin{aligned} \mathbb{E}(D) &= \mathbb{E}(S - B_1 - B_2) = \mathbb{E}(S) - \mathbb{E}(B_1) - \mathbb{E}(B_2) \\ \mathbb{E}(D) &= \frac{\lambda\,(10\,\lambda + 7\,\mu)}{\mu\,(7\,\lambda^2 + 8\,\lambda\mu + 4\,\mu^2)} \end{aligned}$$

In the next section, the blocking property is removed. This allows jobs to wait when the first stage is fully occupied. Computationally, this situation is more complex. Other methods are necessary to estimate the system's performance.

9.4 Tandem queue with non-overtaking

Tandem queues with multiple servers in a stage or different paths for jobs can be subjected to overtaking. The next queue is a tandem queue with two identical servers in parallel in stage one, with one queue with no restriction on the number of jobs in the queue. The second stage contains an individual server with no prior queue. A graphical representation as seen in most queuing theory related articles and textbooks is shown in Figure 9.11. The possibility of overtaking is not allowed and therefore earlier arrivals lead to earlier departures.

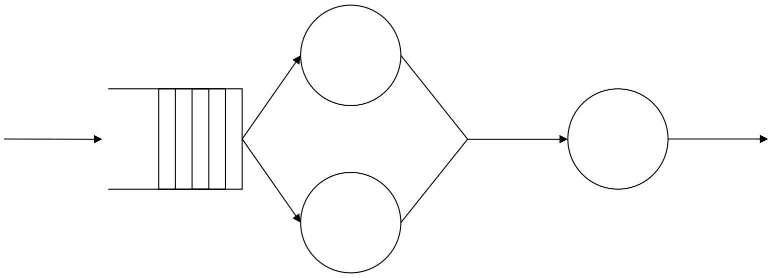


Figure 9.11: Graphical representation of the tandem queue

Let B_1^i denote the service time of job i in stage 1 and B_2^i the service time of job i in stage 2. Define the total time spent in service T_i of job i by the time the job stays in the stages 1 and 2. This includes the possible delay which occurs if a job has finished the service needed in stage 1 but a previous job still occupies the server in stage 2. Let D^i be the delay for job i in this situation. The total time spent in service can be written as:

$$T^i = B_1^i + D^i + B_2^i.$$

The service times in every server are exponentially distributed with parameter μ (resulting in a mean service time at a server of $1/\mu$) and the arrivals follow a Poisson process with arrival rate λ. A job which initiates a busy period finds an empty system and therefore will not be interrupted by another job. The total service time is

Erlang-2(μ) distributed because it passes two stages with exponentially distributed service times and no possibility of being delayed (overtaking is prohibited).
Other jobs however will be subjected to a differently distributed total service time due to the possibility of delay.
Writing the system in Kendall's adjusted notation gives us the queue $M/G/2 \rightarrow M/1/0$ where the G stands for an arbitrary distribution because of the adaptation of the exponential distribution due to delays.

The state space can be defined by $N = \{(i, j, k, l)|i = \{0, 1, r\},\ j = \{0, 1, r\},\ k = \{0, 1\},\ l = \{0, 1, 2, \ldots\}\}$. The state space is identical to the state space which was used in the case with blocking, with the extension of a fourth variable indicating the number of jobs in the queue.
The corresponding balance equations are stated below.

$$\begin{aligned}
\lambda p_{(0,0,0,0)} &= \mu p_{(0,0,1,0)} \\
(\lambda+\mu)p_{(1,0,0,0)} &= \lambda p_{(0,0,0,0)} + \mu p_{(1,0,1,0)} \\
(\lambda+2\mu)p_{(1,1,0,0)} &= \lambda p_{(1,0,0,0)} + \mu p_{(1,1,1,0)} \\
(\lambda+\mu)p_{(0,0,1,0)} &= \mu p_{(1,0,0,0)} + \mu p_{(r,0,1,0)} \\
(\lambda+2\mu)p_{(1,0,1,0)} &= \lambda p_{(0,0,1,0)} + \mu p_{(1,1,0,0)} + \mu p_{(r,1,1,0)} \\
(\lambda+3\mu)p_{(1,1,1,0)} &= \lambda p_{(1,0,1,0)} + \mu p_{(1,1,0,1)} + \mu p_{(r,1,1,1)} \\
(\lambda+\mu)p_{(1,r,0,0)} &= \mu p_{(1,1,0,0)} + \mu p_{(1,r,1,0)} \\
(\lambda+\mu)p_{(r,0,1,0)} &= \mu p_{(1,0,1,0)} + \mu p_{(1,r,0,0)} + \mu p_{(r,r,1,0)} \\
(\lambda+2\mu)p_{(r,1,1,0)} &= \mu p_{(1,1,1,0)} + \mu p_{(1,r,0,1)} + \lambda p_{(r,0,1,0)} + \mu p_{(r,r,1,1)} \\
(\lambda+2\mu)p_{(1,r,1,0)} &= \mu p_{(1,1,1,0)} \\
(\lambda+\mu)p_{(r,r,1,0)} &= \mu p_{(r,1,1,0)} + \mu p_{(1,r,1,0)} \\
\text{and for } i \geq 1 : & \\
(\lambda+2\mu)p_{(1,1,0,i)} &= \lambda p_{(1,1,0,i-1)} + \mu p_{(1,1,1,i)} \\
(\lambda+3\mu)p_{(1,1,1,i)} &= \lambda p_{(1,1,1,i-1)} + \mu p_{(1,1,0,i+1)} + \mu p_{(r,1,1,i+1)} \\
(\lambda+\mu)p_{(1,r,0,i)} &= \lambda p_{(1,r,0,i-1)} + \mu p_{(1,1,0,i)} + \mu p_{(1,r,1,i)} \\
(\lambda+2\mu)p_{(r,1,1,i)} &= \lambda p_{(r,1,1,i-1)} + \mu p_{(1,r,0,i+1)} + \mu p_{(r,r,1,i+1)} + \mu p_{(1,1,1,i)} \\
(\lambda+2\mu)p_{(1,r,1,i)} &= \lambda p_{(1,r,1,i-1)} + \mu p_{(1,1,1,i)} \\
(\lambda+\mu)p_{(r,r,1,i)} &= \lambda p_{(r,r,1,i-1)} + \mu p_{(r,1,1,i)} + \mu p_{(1,r,1,i)}
\end{aligned}$$

The iterative equations can be written as follows:

$$\begin{aligned}
p_{(1,1,0,i+1)} &= \left(\frac{\lambda}{\lambda+2\mu}\right)^{i+1} p_{(1,1,0,0)} + \frac{\mu}{\lambda}\sum_{k=1}^{i+1}\left(\frac{\lambda}{\lambda+2\mu}\right)^{i+2-k} p_{(1,1,1,k)} \\
p_{(1,1,1,i+1)} &= \left(\frac{\lambda}{\lambda+3\mu}\right)^{i+1} p_{(1,1,1,0)} + \frac{\mu}{\lambda}\sum_{k=2}^{i+2}\left(\frac{\lambda}{\lambda+3\mu}\right)^{i+3-k} (p_{(1,1,0,k)} + p_{(r,1,1,k)}) \\
p_{(1,r,0,i+1)} &= \left(\frac{\lambda}{\lambda+\mu}\right)^{i+1} p_{(1,r,0,0)} + \frac{\mu}{\lambda}\sum_{k=1}^{i+1}\left(\frac{\lambda}{\lambda+\mu}\right)^{i+2-k} (p_{(1,1,0,k)} + p_{(1,r,1,k)}) \\
p_{(r,1,1,i+1)} &= \left(\frac{\lambda}{\lambda+2\mu}\right)^{i+1} p_{(r,1,1,0)} + \frac{\mu}{\lambda}\sum_{k=1}^{i+1}\left(\frac{\lambda}{\lambda+2\mu}\right)^{i+2-k} p_{(1,1,1,k)} \\
&\quad + \frac{\mu}{\lambda}\sum_{k=2}^{i+2}\left(\frac{\lambda}{\lambda+2\mu}\right)^{i+3-k} (p_{(1,r,0,k)} + p_{(r,r,1,k)}) \\
p_{(1,r,1,i+1)} &= \left(\frac{\lambda}{\lambda+2\mu}\right)^{i+1} p_{(1,r,1,0)} + \frac{\mu}{\lambda}\sum_{k=1}^{i+1}\left(\frac{\lambda}{\lambda+2\mu}\right)^{i+2-k} p_{(1,1,1,k)} \\
p_{(r,r,1,i+1)} &= \left(\frac{\lambda}{\lambda+\mu}\right)^{i+1} p_{(r,r,1,0)} + \frac{\mu}{\lambda}\sum_{k=1}^{i+1}\left(\frac{\lambda}{\lambda+\mu}\right)^{i+2-k} (p_{(r,1,1,k)} + p_{(1,r,1,k)}).
\end{aligned}$$

Solving these equations analytically seemed to be too complex. Again, the iterative Gauss-Seidel method (introduced in section 9.2) is used to approximate the steady state probabilities. The program follows the same steps as the program used for the single stage queue and is therefore omitted from the thesis.
The expected waiting time, sojourn time and delay are calculated using the steady state probabilities. Figure 9.12 shows these performance measures for this queuing system.

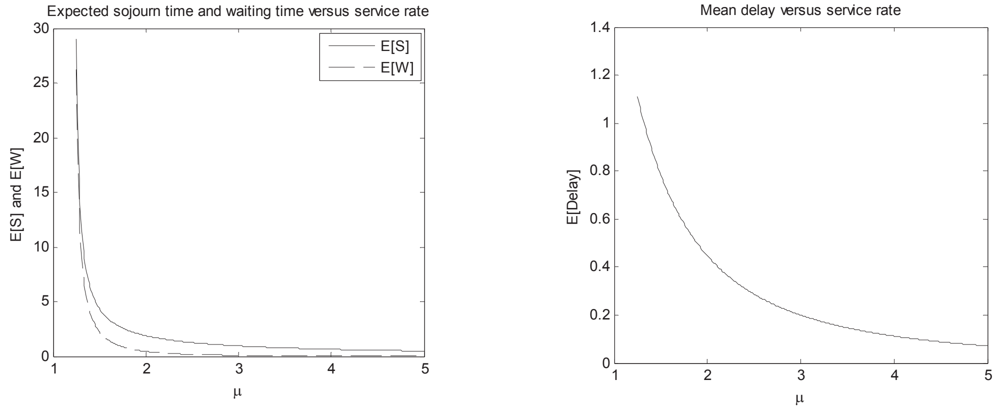


Figure 9.12: Performance measures for the two stage tandem queue without overtaking

The restriction on overtaking and no queue in front of the second stage results in some jobs being delayed. The possibility for a job to be delayed can be calculated by summing all steady state probabilities corresponding to a state containing a delayed job (PASTA). Figure 9.13 shows this probability for different service rates. For low service rates, the probability of delay can be very high.

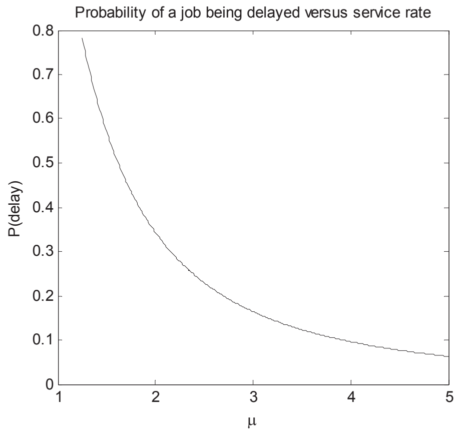


Figure 9.13: Probability of delay for the two stage tandem queue without overtaking

The amount of delay a job obtains in different situations can be calculated when conditioning on the situation in which the system is when the job arrives. The first job which initiates a busy period is not subjected to delay. Jobs which arrive during a busy period however, can arrive in a situation in which they have to wait on a job in front of them. There are four situations in which a job can start service when it is not the first in a busy period.

- Situation 1: $(1,0,0,k)$, a job starts service when only one server in stage 1 is serving another job.
- Situation 2: $(0,0,1,0)$, a job starts service when one job is in service in stage 2.
- Situation 3: $(1,0,1,k)$, a job starts service when one job is in service in stage 1 and one job is in service in stage 2.
- Situation 4: $(r,0,1,k)$, a job starts service when one job is ready in server 1 (but blocks the server) and one job is in service in stage 2.

The job starting service is the job in front of the queue. Situation 2 can not have a queue since the two servers in stage 1 are both idle and any job in the queue would already be in service (the system would then be in situation 1, 3 or 4). Delay of a job arriving in situation 1:

$$D^i = \max(0, R_1^{i-1} + B_2^{i-1} - B_1^i) \overset{d}{=} \max(0, B_1^{i-1} + B_2^{i-1} - B_1^i).$$

Where R_1^{i-1} is the residual service time of job $i-1$ at stage 1 and the equality is valid because of the memoryless property of the exponential distributed service times.
The delay of a job arriving in situation 2:

$$D^i = \max(0, R_2^{i-1} - B_1^i) \overset{d}{=} \max(0, B_2^{i-1} - B_1^i).$$

The delay of a job arriving in situation 3:

$$D^i = \max(0, R_1^{i-1} + D^{i-1} + B_2^{i-1} - B_1^i) = \max(0, R_1^{i-1} + \max(0, R_2^{i-2} - R_1^{i-1}) + B_2^{i-1} - B_1^i) \overset{d}{=}$$
$$\max(0, B_1^{i-1} + \max(0, B_2^{i-2} - B_1^{i-1}) + B_2^{i-1} - B_1^i).$$

The delay of a job arriving in situation 4:

$$D^i = \max(0, R_2^{i-2} + B_2^{i-1} - B_1^i) \overset{d}{=} \max(0, B_2^{i-2} + B_2^{i-1} - B_1^i).$$

Conditioning on the service times and using the law of total probability results in closed forms for the expected delays.

$$\mathbb{E}[D^i|\text{arrival in situation 1}] = \mathbb{E}[\mathbb{E}[\mathbb{E}[\max(0, B_1^{i-1}{+}B_2^{i-1}{-}B_1^i)|B_1^i = x]|B_1^{i-1} = y]] =$$

$$\int_0^\infty \int_0^\infty \int_0^{B_2^{i-1}+y} (B_2^{i-1} + y - x)\mu e^{-\mu x}\ dx\ \mu e^{-\mu y}\ dy\ \mu e^{-\mu B_2^{i-1}}\ dB_2^{i-1} =$$

$$\int_0^\infty \int_0^\infty \left((B_2^{i-1} + y) - \frac{1}{\mu}\left(1 - e^{-\mu(B_2^{i-1}+y)}\right)\right) \mu e^{-\mu y}\ dy\ \mu e^{-\mu B_2^{i-1}}\ dB_2^{i-1} =$$

$$\int_0^\infty \left(B_2^{i-1} - \frac{1}{\mu} + \frac{1}{\mu} + \frac{e^{-\mu B_2^{i-1}}}{2\mu}\right) \mu e^{-\mu B_2^{i-1}}\ dB_2^{i-1} = \frac{5}{4\mu}$$

Where integration by parts is used several times.

$$\mathbb{E}[D^i|\text{arrival in situation 2}] = \mathbb{E}[\mathbb{E}[\max(0, B_2^{i-1} - B_1^i)|B_1^i = x]] =$$

$$\int_0^\infty \int_0^{B_2^{i-1}} (B_2^{i-1} - x)\mu e^{-\mu x} dx \mu e^{-\mu B_2^{i-1}} dB_2^{i-1} =$$

$$\int_0^\infty \left(B_2^{i-1} - \frac{1}{\mu} + \frac{e^{-\mu B_2^{i-1}}}{\mu}\right) \mu e^{-\mu B_2^{i-1}} dB_2^{i-1} = \frac{1}{2\mu}$$

Similar calculations result in the expected delay for a job arriving in situation 3.

$$\mathbb{E}[D^i|\text{arrival in situation 3}] =$$

$$\mathbb{E}[\mathbb{E}[\mathbb{E}[\mathbb{E}[\max(0, B_1^{i-1}{+}\max(0, B_2^{i-2}{-}B_1^{i-1}){+}B_{i-1}^2{-}B_1^i)|B_1^i = x]|B_1^{i-1} = y]|B_{i-1}^2 = z]] =$$

$$\int_0^\infty \int_0^\infty \int_0^{B_2^{i-2}} \int_0^{y+\max(0,B_2^{i-2}-y)+z} \big(y + \max(0, B_2^{i-2} - y) + z - x)\big)$$

$$\mu e^{-\mu x}\ dx\ \mu e^{-\mu y}\ dy\ \mu e^{-\mu z}\ dz\ \mu e^{-\mu B_2^{i-2}} dB_2^{i-2} =$$

$$\int_0^\infty \int_0^\infty \left((B_2^{i-2} + z) - \frac{1}{\mu}(1 - e^{-\mu(B_2^{i-2}+z)})\right) (1{-}e^{-\mu B_2^{i-2}})\ \mu e^{-\mu z}\ dz\, \mu e^{-\mu B_2^{i-2}} dB_2^{i-2}+$$

$$\int_0^\infty \int_0^\infty \left((B_2^{i-2} + z)e^{-\mu B_2^{i-2}} + \frac{e^{-\mu(z+2B_2^{i-2})}}{2\mu}\right)\ \mu e^{-\mu z}\ dz\ \mu e^{-\mu B_2^{i-2}} dB_2^{i-2} =$$

$$\int_0^\infty \left((B_2^{i-2} - \frac{1}{\mu} + \frac{1}{\mu} + \frac{e^{-\mu B_2^{i-2}}}{2\mu})(1 - e^{-\mu B_2^{i-2}})\right)\ \mu e^{-\mu B_2^{i-2}} dB_2^{i-2}+$$

$$\int_0^\infty \left(\frac{e^{-\mu B_2^{i-2}}}{\mu} + B_2^{i-2} e^{-\mu B_2^{i-2}} + \frac{e^{-2\mu B_2^{i-2}}}{4\mu}\right)\ \mu e^{-\mu B_2^{i-2}} dB_2^{i-2}$$

$$= \frac{5}{6\mu} + \frac{5}{6\mu} = \frac{5}{3\mu}$$

A job arriving in situation 4 has the same expected delay as a job arriving in situation 1, because not only the service times at servers in stage 1 and the server in stage 2 are equally distributed, but also the other job in stage 1 has already completed its service and can proceed immediately to server 2 if this server becomes empty. Therefore the residual delay of the job in stage 1 equals the residual service time of the job in server 2.

$$\mathbb{E}[D^i|\text{arrival in situation 4}] = \frac{5}{4\mu}$$

With these expressions for the expected delay in certain situations, the expected service time for these jobs can be derived very easily. The expected total time in service ($\mathbb{E}[T]$) consists of the service time in stage 1, the delay in stage 1 and the service time in stage 2.

$$\begin{aligned}
\mathbb{E}[T^i|\text{arrival in situation 1}] &= \frac{1}{\mu} + \frac{5}{4\mu} + \frac{1}{\mu} = \frac{13}{4\mu} \\
\mathbb{E}[T^i|\text{arrival in situation 2}] &= \frac{5}{2\mu} \\
\mathbb{E}[T^i|\text{arrival in situation 3}] &= \frac{11}{3\mu} \\
\mathbb{E}[T^i|\text{arrival in situation 4}] &= \frac{13}{4\mu}
\end{aligned}$$

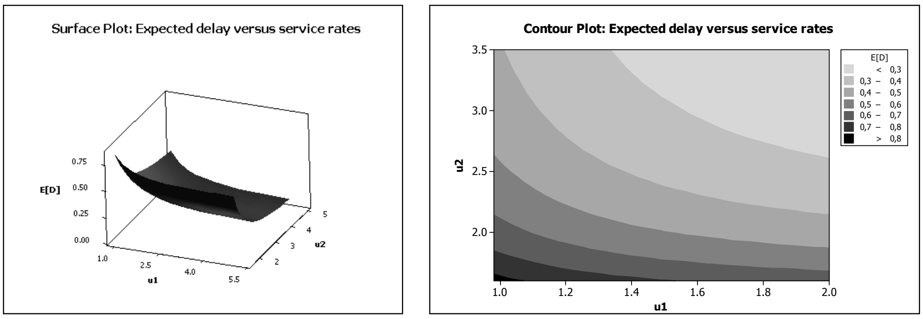
A job finding one job in stage 1 and one in stage 2 (situation 3) has therefore an expected total service time of $11/3\mu$ of which only $2/\mu$ is its actual service time.

For jobs starting service in the system, the expected time in service and the expected delay were calculated. For arbitrary jobs, the expected delay and time spent in the system were approximated using the Gauss-Seidel iterative method. In the next section, these expressions are calculated for a server with different service rates between the stages.

9.4.1 Different service times between stages

In practice the assumption of the same mean service time is one which often can not be justified (nor are the Poisson process or exponentially distributed service times but to make it analytical doable one needs to make concessions). Therefore the same set up is considered as in the previous section, but with different service rates between the two stages. The service rate of the servers in stage 1 is μ_1 and the service rate in stage 2 is μ_2.
The resulting balance equations are more complicated. By adjusting the previous iterative program to calculate the steady state probabilities, the performance measures are calculated.

Figure 9.14 shows the expected delay of an arbitrary job for different service rates. A decrease in service rate will increase the expected delay. Graphs for the expected sojourn time, expected waiting time and probability of delay are given in Figure 12.21 in the Appendix.



Surface Plot: Expected delay versus service rates

Contour Plot: Expected delay versus service rates

Figure 9.14: Expected delay of an arbitrary job for different service rates

Following the same steps as in section 9.4, the formulas for the expected delays in the given situations (1, 2, 3 or 4 as stated in the previous section) are obtained.

$$\begin{aligned}
\mathbb{E}[D^i|\text{arrival in situation 1}] &= \frac{2\mu_1(\mu_1+\mu_2)+\mu_2^2}{2\mu_1\mu_2(\mu_1+\mu_2)} \\
\mathbb{E}[D^i|\text{arrival in situation 2}] &= \frac{\mu_1^2}{\mu_1\mu_2(\mu_1+\mu_2)} \\
\mathbb{E}[D^i|\text{arrival in situation 3}] &= \frac{\mu_1(8\mu_1(\mu_1+\mu_2)+3\mu_2^2)+\mu_2^3}{2\mu 1\mu 2(\mu_1+\mu_2)(2\mu_1+\mu_2)} \\
\mathbb{E}[D^i|\text{arrival in situation 4}] &= \frac{\mu_1(2\mu_1+3\mu_2)}{\mu_2(\mu_1+\mu_2)^2}
\end{aligned}$$

Taking the same value for μ_1 and μ_2 results in the previously obtained formulas. Since the service times in the two stages are not equally distributed anymore, the expected time a job is in service also changes.

$$\begin{aligned}
\mathbb{E}[T^i|\text{arrival in situation 1}] &= \frac{2\mu_1(2\mu_1+3\mu_2)+3\mu_2^2}{2\mu_1\mu_2\left(\mu_1+\mu_2\right)} \\
\mathbb{E}[T^i|\text{arrival in situation 2}] &= \frac{\mu_1(3\mu_1+4\mu_2)+2\mu_2^2}{2\mu_1\mu_2\left(\mu_1+\mu_2\right)} \\
\mathbb{E}[T^i|\text{arrival in situation 3}] &= \frac{\mu_1(6\mu_1(2\mu_1+3\mu_2)+11\mu_2^2)+3\mu_2^3}{2\mu 1\mu 2(\mu_1+\mu_2)(2\mu_1+\mu_2)} \\
\mathbb{E}[T^i|\text{arrival in situation 4}] &= \frac{3\mu_1(\mu_1(\mu_1+2\mu_2)+\mu_2)+\mu_2^3}{\mu_2\left(\mu_1+\mu_2\right)^2\mu_1}
\end{aligned}$$

In the next chapter, more complex systems are investigated using discrete event simulation.

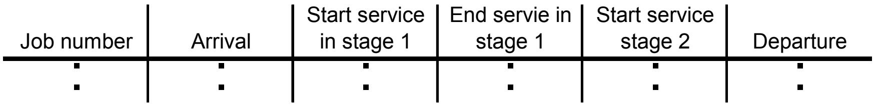
Chapter 10

Discrete event simulation

In this chapter, the single stage queue and the tandem queue investigated in the previous chapter are investigated more thoroughly. Performance characteristics are obtained for cases with different service time distributions and different numbers of servers in the stages. The restriction of only two stages is still maintained, but more stages could of course be incorporated in the design of the program.
For a variety of distributions, bounds are needed to make the theory applicable in practice. The goal is to acquire an overview of the characteristics of these particular queuing systems in order to facilitate in approximating expected delays, waiting times, throughput times and delay probabilities.

10.1 Model description

A model to investigate the particular queue has been made in Matlab 7. At the beginning the model is initialized. An end time and a start time are defined, the distributions for the service times and arrival times as well as the number of servers in stage 1 and 2 are chosen. The program works roughly as follows: after each event, the times for all possible next events are calculated and the minimum is taken to be the next event (an arrival or service completion).
As a result, the program is able to generate a matrix which is as stated in Figure 10.1.



Job number	Arrival	Start service in stage 1	End servie in stage 1	Start service stage 2	Departure
⋮	⋮	⋮	⋮	⋮	⋮

Figure 10.1: Output matrix from discrete event simulation.

From this matrix the performance of the queue can be easily derived since it contains all relevant data. The program for the single stage tandem queue with the non-overtaking property is obtained by removing one stage from the tandem queue program.

10.2 Warm-up period in discrete event simulations

When using discrete event simulation, the problem of reaching steady state needs to be noticed. A simulation program normally does not start in steady state, but in an empty state or some other specific state. However when one is interested in steady state behavior, which is most often the case in simulating theoretical queues, this can seriously disrupt estimating performance characteristics. The problem which arises is called initialization bias, warm-up period, start up or initial transient.

A commonly used method to determine the length of the warm-up period is using the Marginal Standard Error Rule (MSER) [25]. The MSER rule calculates the variance of the estimated characteristic for different subsets of the whole simulation. It starts with the complete set and removes the first observation, the variance of the remaining observations is then calculated. This procedure repeats itself for about half the total number of observations. The point at which this variance is minimal is taken as the truncation point. The idea behind this procedure is that this will result in tighter confidence intervals for the estimated performance measure since the variance is minimal. After determining the truncation point, the performance measure (in this case the mean sojourn time) is estimated.

10.3 Single stage queue

The first queue which is examined is a single stage queue consisting of n servers in parallel and with the property that overtaking is prohibited (note that when n equals 1, the ordinary M/G/1 queue is obtained which is not influenced by the restriction on overtaking). The time a job is in the system, the sojourn time, is the main interest for further research. Knowing this performance measure is one most relevant in practice. For most jobs the time in the system is the property one usually wants to minimize, since the longer a job stays in the system, the higher for example the costs are.

Many problems emerge in the analysis of real-life systems. As shown in the previous chapters, it is often difficult to derive exact formulas for the performance characteristics. When this problem occurs, an appropriate solution could be using bounds instead. The system of interest could be bounded by two systems for which exact solutions are known.

The expected sojourn time in the non-overtaking queue consists of a waiting time before entering service, a service time and an expected delay due to the restriction on overtaking. In the normal M/G/n queue without the non-overtaking property, the expected service time is the same but the expected delay equals zero because overtaking in a stage is allowed and a job leaves the system when finishing its service. A decrease in service occupancy also results in a lower expected waiting time for the normal M/G/n queue. Therefore, the M/G/n queue and its lower occupancy, can be used to calculate a lower bound for the expected sojourn time.

To obtain a rough upper bound, the servers in the first stage are put in series. By letting every job go through the same number of servers in series as there are servers in the first stage, the worst case scenario for a job is obtained. It resembles the situation in which an arriving job finds a system with in the first stage all servers busy except for one. All jobs in front of the arriving job are still in service and complete their services in consecutive order (first job 1 completes service, then 2, etc.). If the arriving job finishes service first, it has to wait for $n-1$ service completions. Including its own service requirement, a total of n service completions are needed before being able to depart from the system. This resulting series system behaves as an M/G/1 queue where the service distribution is the convolution (sum of distributions) of n times the service distribution of the non-overtaking queue.

Using the Pollaczek-Khinchin formula an estimate on the sojourn time is obtained [39].

$$\begin{aligned}
\mathbb{E}(S_U) = \mathbb{E}(B_U) + \mathbb{E}(W) &= \mathbb{E}(B_U) + \frac{\lambda * \mathbb{E}[B_U^2]}{2(1-\rho_U)} \\
\mathbb{E}S_U &= (1 + \frac{1+scv}{2}\frac{\rho_U}{1-\rho_U}) * \mathbb{E}(B_U)
\end{aligned}$$

10.3.1 The M/M/n non-overtaking queue

The case where arrival and service times are homogeneous distributed according to an exponential distribution is simulated. For different numbers of servers, the performance measures are estimated. A single simulation run ends when 10.000 jobs have past the system. Repeating the simulation run 10 times, more accurate estimates for the performance measures are obtained. Every estimation will therefore use the information of 100.000 jobs minus the jobs discarded due to truncation (to determine steady state performance).

As stated before, the M/M/n queue is used to derive a lower bound on the expected sojourn time. the formula $\mathbb{E}(S_L) = \frac{\Pi_{W_L} + n(1-\rho_L)}{n\mu(1-\rho_L)}$ can be used to calculate a lower bound [39]:

$$\begin{aligned}
\mathbb{E}(S_L) = \mathbb{E}(W_L) + \mathbb{E}(B) &= \Pi_{W_L}\frac{1}{n\mu} + \mathbb{E}(L_L^q)\frac{1}{n\mu} + \mathbb{E}(B) \\
\text{where } \Pi_{W_L} &= \frac{(n\rho_L)^n}{n!}\left((1-\rho_L)\sum_{i=0}^{n-1}\frac{(n\rho_L)^i}{i!} + \frac{(n\rho_L)^n}{n!}\right)^{-1} \\
\mathbb{E}(L_L^q) &= \Pi_{W_L}\frac{\rho_L}{1-\rho_L} \\
\mathbb{E}(B) &= \frac{1}{\mu} \\
\text{thus } \mathbb{E}(S_L) &= \frac{\Pi_{W_L} + n(1-\rho_L)}{n\mu(1-\rho_L)}
\end{aligned}$$

Taking the load (ρ_L) equal to $1/n\mu$, which is smaller than the load in the non-overtaking queue with the same arrival and service rate (due to delays), a lower bound of $(\mu(\Pi_{W_L} + 1) - 1)/(\mu(\mu - 1))$ is calculated. The Poisson distribution is used to calculate the waiting probability. This can be done easily since many software packages are capable of calculating Poisson probabilities.
The service time distribution needed for the upper bound queue is given by the n-convolution of the exponential distribution. This results in an Erlang-n distribution with mean n/μ and variance n/μ^2. Substituting these in the formula for the mean sojourn time given above results in an upper bound. In obtaining an upper bound, the occupancy has to be estimated. Using the true occupancy of the system is a possibility, but since this information is not a priori known, also this measure has to be estimated.
The analysis is carried out for three servers. Other systems are discussed only briefly before results are given.

The occupancy is tried to be estimated using only the service rate. The occupancy of the system is regarded as one of the key performance indicators. The estimated occupancy can be used to determine an upper bound for the expected sojourn time.

In figure 10.2, the fit of a third degree polynomial is presented. Instead of plotting the occupancy versus the service rate, 10.2 shows the occupancy versus the inversed service rate. This is done because in the transfer function the $1/\mu$ is used (since it equals the expected service time) instead of μ.

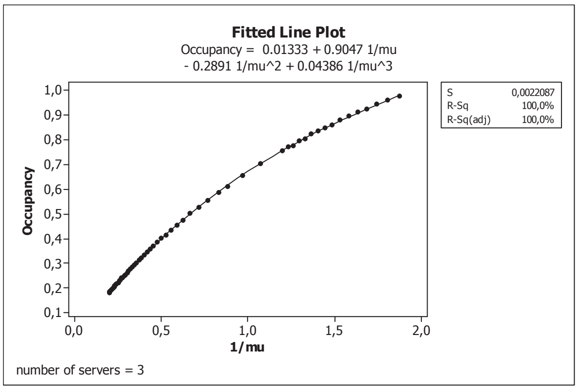
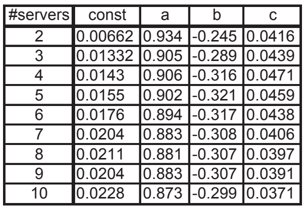


Figure 10.2: A polynomial of degree three fit for the occupancy versus the inverse service rate

It is advised to fit polynomials of the smallest degree which still represent the data in a fair way. A polynomial of higher order will give a more precise fit, but for practical reasons this is not recommended. Note that when using transfer functions (the function that resembles the correspondence between the independent variable, the x, and the dependent variable, the y as in y=f(x) for some function f) only estimations are allowed within the investigated range. Extrapolating is not advised since the behavior outside the investigated region is not known.
For systems with different numbers of servers the transfer functions are given in Figure 10.3.



#servers	const	a	b	c
2	0.00662	0.934	-0.245	0.0416
3	0.01332	0.905	-0.289	0.0439
4	0.0143	0.906	-0.316	0.0471
5	0.0155	0.902	-0.321	0.0459
6	0.0176	0.894	-0.317	0.0438
7	0.0204	0.883	-0.308	0.0406
8	0.0211	0.881	-0.307	0.0397
9	0.0204	0.883	-0.307	0.0391
10	0.0228	0.873	-0.299	0.0371

Figure 10.3: Occupancy = const+a/μ+b/μ^2+c/μ^3

In Figure 10.4, the simulated expected sojourn time for the single stage queue with 3 servers without overtaking and the calculated lower- and upper bounds are shown for different values of μ. The lower bound seems to be a good estimate for the expected sojourn time when the service rate increases. But when service rates are low (corresponding with a higher occupancy) the expected sojourn time differs largely from the lower bound as well as from the two proposed upper bounds. When dealing with high load systems this might cause problems. But despite the bad performance of the lower bound as an estimate for the mean sojourn time in high load systems, it still performs better than the upper bound.

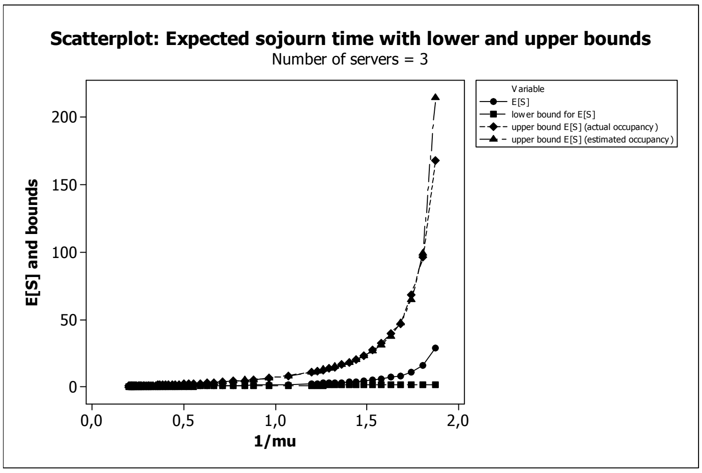


Figure 10.4: Expected sojourn time and calculated bounds

In Figure 10.5, the influence of the number of servers on the difference between the expected sojourn time and its bounds can be seen. Where the lower bound performs worse when the number of servers is low, the upper bounds perform worse when the number of servers is high.

To obtain a transfer function for the expected mean sojourn time one can use these bounds. As stated before, in highly occupied systems, both bounds fail to give an applicable estimate on the mean sojourn time. Therefore a transfer function with a bound as well as $1/\mu$ might be suitable. The option of taking an upper or lower bound presents itself. Since the lower bound is more difficult to calculate, one might prefer using the upper bound, which depends on the system's occupancy, the number of servers and the service rate. The actual system's occupancy can be taken equal to the simulated occupancy, but as stated before it is more realistic to estimate this measure first. Figure 10.6 shows the transfer function for the expected sojourn time for three servers. In Figure 10.7 the coefficients of the transfer function $\mathbb{E}[S] = \text{const} + a * \frac{1}{\mu} + b * \mathbb{E}[S_U]$ are given.

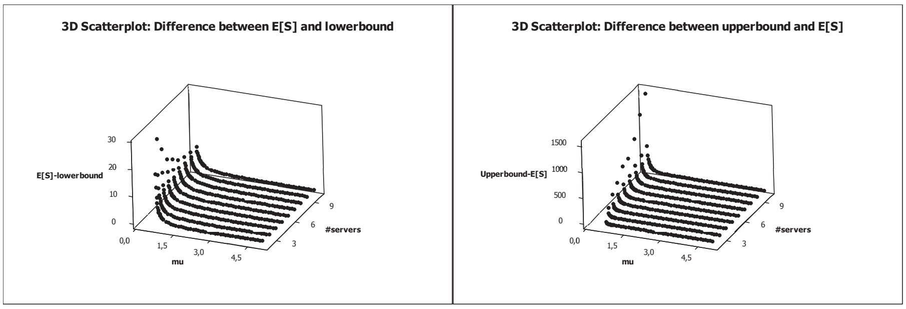


Figure 10.5: Differences between simulated expected sojourn time and bounds

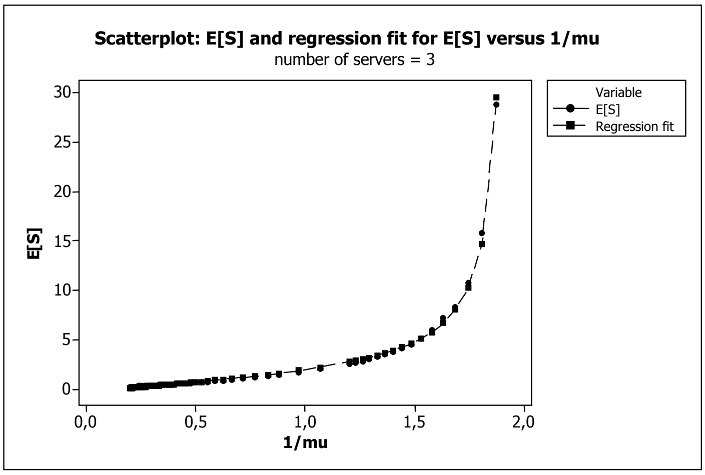
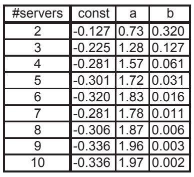


Figure 10.6: Expected sojourn time and its regression fit for a system with 3 servers



#servers	const	a	b
2	-0.127	0.73	0.320
3	-0.225	1.28	0.127
4	-0.281	1.57	0.061
5	-0.301	1.72	0.031
6	-0.320	1.83	0.016
7	-0.281	1.78	0.011
8	-0.306	1.87	0.006
9	-0.336	1.96	0.003
10	-0.336	1.97	0.002

Figure 10.7: $\mathbb{E}[S] = \text{const} + a * \frac{1}{\mu} + b * \mathbb{E}[S_U]$

The expected delay seems to be equal to $1/\mu$ for the case of two servers, see Figure 10.8. When the number of servers increases, the average delay tends to deviate more from a linear relation with $1/\mu$.

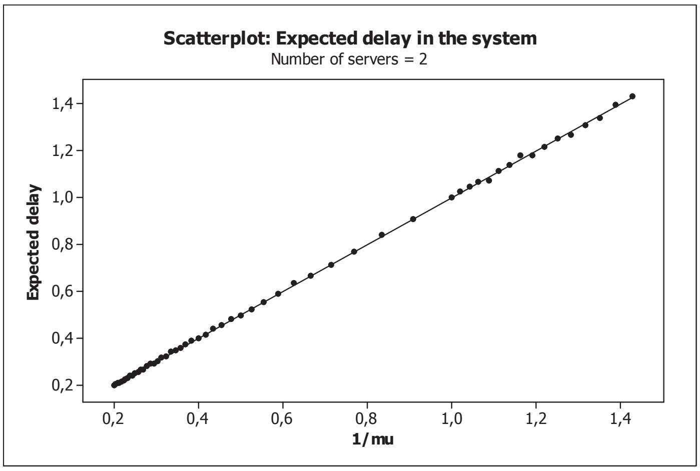


Figure 10.8: Scatterplot of the delay versus $1/\mu$ in a system of 2 servers

For the system with two parallel servers the expected delay can be calculated analytically. By defining W_2 as the waiting time of a job to start service while the first server is busy, R_i as the residual service time of the job in service and B_{i+1}

as the service time of the commencing job, the equality $W_2 = \max(0, R_i - B_{i+1})$ is valid. Using the same integration techniques as before, the expected value of W_2 can be shown to be equal to $1/2\mu$. This expectation however also considers jobs which do not have to wait. When looking at the jobs which actually get delayed (this percentage is 50% of all jobs arriving in a busy period due to the memoryless property), their expected delay is twice as big which results in $1/\mu$.

Transfer functions of expected delay and the probability of delay are presented for the different systems in Figure 10.9. The probability of delay is fitted using a polynomial of degree three ($P(\text{Delay}) = \text{const} + a * \frac{1}{\mu} + b * \frac{1}{\mu^2} + c * \frac{1}{\mu^3}$) and the expected delay a job obtains (if it is delayed) is fitted using a polynomial of degree 2 ($\mathbb{E}(\text{Delay}) = \text{const} + a * \frac{1}{\mu} + b * \frac{1}{\mu^2}$). Figure 10.10 shows the polynomial fit for the probability of delay when the system has three servers.

Prob(Delay)

#servers	const	a	b	c
2	0.0124	0.385	-0.172	0.0388
3	0.0069	0.461	-0.168	0.0292
4	0.00255	0.487	-0.161	0.0243
5	0.00404	0.479	-0.142	0.0188
6	0.00389	0.478	-0.135	0.0167
7	0.00524	0.472	-0.126	0.0144
8	0.00472	0.474	-0.126	0.0144
9	0.00388	0.479	-0.129	0.0146
10	0.00559	0.471	-0.121	0.0129

E[Delay]

#servers	const	a	b
2	0	1	0
3	-0.0064	1.04	0.060
4	-0.0104	1.05	0.094
5	-0.0063	1.03	0.125
6	0	1.00	0.151
7	0	1.01	0.154
8	0	1.01	0.155
9	0	0.99	0.161
10	0	1.02	0.155

Figure 10.9: Transfer functions for the average probability of delay and expected delay

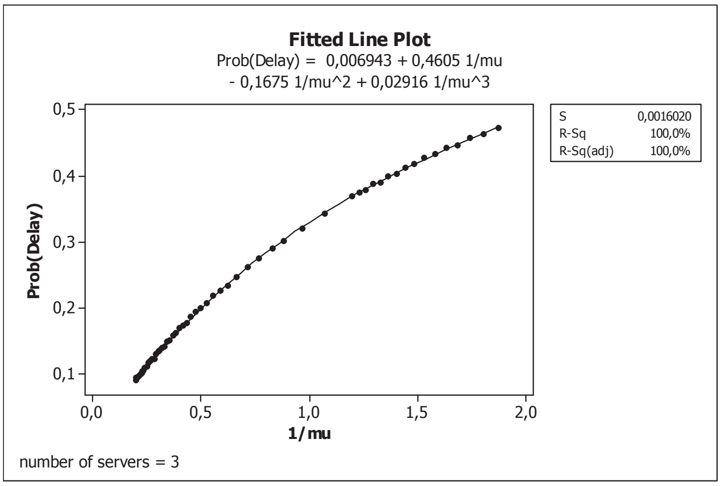


Figure 10.10: Probability of being delayed fitted by a polynomial of degree 3

In the next section, the queue with three parallel servers is subjected to differently distributed service and arrival times.

10.3.2 The G/G/3 non-overtaking queue

In the previous section, the inter arrival times and service times were homogeneous exponentially distributed. This section investigates what happens when the arrival and service times have other homogeneous distributions.

A measure to determine the dispersion of a distribution is the squared coefficient of variation (*scv*).

$$\begin{aligned} scv &= \frac{\sigma^2}{\mu^2} \\ \sigma &= \text{standard deviation} \\ \mu &= \text{mean} \end{aligned}$$

For the exponential distribution, the *scv* equals 1 (mean and variance and standard deviation are the same). A phase-type distribution is used to obtain the service time distribution with different $scv's$. Phase-type distributions are dense in the field of all positive-valued distributions, this means that any distributions which can only obtain positive values can be represented by a phase-type distribution. For distributions with an $scv < 1$, an Erlang distribution is used, if $scv > 1$ a hyper-exponential distribution is used.

In this section only systems are investigated consisting of three servers with service and arrival distributions with *scv* larger then 1.
Overall can be seen that increasing the squared coefficient of variation results in worse performance of the system (assuming longer sojourn times and higher delays are not desirable). Figure 10.11 shows the expected sojourn time and expected delay, other similar graphs are omitted.

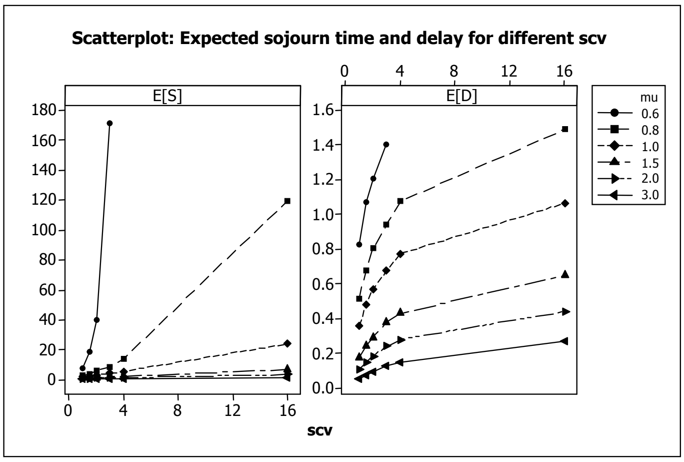


Figure 10.11: Simulation procedure

In the following sections, discrete event simulation is used to investigate the performance of the tandem queue when overtaking is prohibited.

10.4 Two stage queue, homogeneous case

The mean sojourn time is the point of interest under different occupancy rates and $scv's$. The arrival distribution is an exponential distribution with parameter 1 to resemble random arrivals with a standardized arrival rate. The difficulty which arises in this queue is that distributional parameters of the service times in the queue are not determined completely by the occupancy rate. Therefore, the occupancy in the second stage is being fixed for every systems occupancy. The parameter for the second stage for distributions with different $scv's$ is then chosen in such a way that

for every distribution the load in the second stage is identical. With the occupancy in the second stage being equal to a given value, the parameter for the first stage can be estimated using trial runs.
The steps necessary to complete one simulation run are graphically presented in flow diagram 10.12.

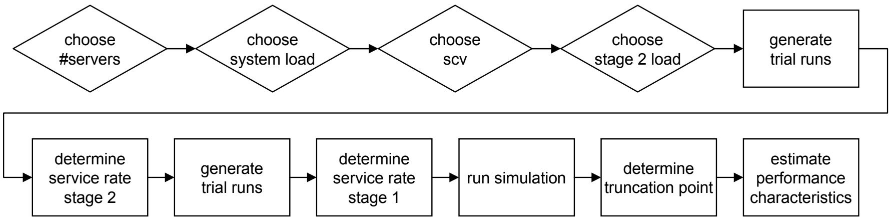
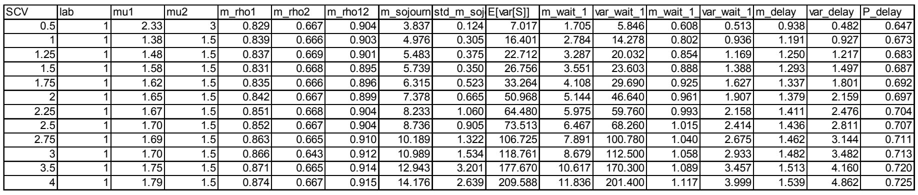

choose #servers
choose system load
choose scv
choose stage 2 load
generate trial runs
determine service rate stage 2
generate trial runs
determine service rate stage 1
run simulation
determine truncation point
estimate performance characteristics

Figure 10.12: Simulation procedure

10.4.1 Tandem queue with occupancy = 0.9

The first queue which is analyzed, is the queue comprised of two stages with in the first stage 2 servers and in the second stage 1 server (see figure 9.11). Choosing 2/3 for the occupancy in the second stage, the service rates were determined and eventually for every *scv* 50 simulation runs are performed. Each time the simulation is stopped when the system successfully served 20000 jobs. The results are given in Figure (10.13).



SCV	lab	mu1	mu2	m_rho1	m_rho2	m_rho12	m_sojourn	std_m_soj	E[var[S]]	m_wait_1	var_wait_1	m_wait_1_	var_wait_1	m_delay	var_delay	P_delay
0.5	1	2.33	3	0.829	0.667	0.904	3.837	0.124	7.017	1.705	5.846	0.608	0.513	0.938	0.482	0.647
1	1	1.38	1.5	0.839	0.666	0.903	4.976	0.305	16.401	2.784	14.278	0.802	0.936	1.191	0.927	0.673
1.25	1	1.48	1.5	0.837	0.669	0.901	5.483	0.375	22.712	3.287	20.032	0.854	1.169	1.250	1.217	0.683
1.5	1	1.58	1.5	0.831	0.668	0.895	5.739	0.350	26.756	3.551	23.603	0.888	1.388	1.293	1.497	0.687
1.75	1	1.62	1.5	0.835	0.666	0.896	6.315	0.523	33.264	4.108	29.690	0.925	1.627	1.337	1.801	0.692
2	1	1.65	1.5	0.842	0.667	0.899	7.378	0.665	50.968	5.144	46.640	0.961	1.907	1.379	2.159	0.697
2.25	1	1.67	1.5	0.851	0.668	0.904	8.233	1.060	64.480	5.975	59.760	0.993	2.158	1.411	2.476	0.704
2.5	1	1.70	1.5	0.852	0.667	0.904	8.736	0.905	73.513	6.467	68.260	1.015	2.414	1.436	2.811	0.707
2.75	1	1.69	1.5	0.863	0.665	0.910	10.189	1.322	106.725	7.891	100.780	1.040	2.675	1.462	3.144	0.711
3	1	1.70	1.5	0.866	0.643	0.912	10.989	1.534	118.761	8.679	112.500	1.058	2.933	1.482	3.482	0.713
3.5	1	1.75	1.5	0.871	0.665	0.914	12.943	3.201	177.670	10.617	170.300	1.089	3.457	1.513	4.160	0.720
4	1	1.79	1.5	0.874	0.667	0.915	14.176	2.639	209.588	11.836	201.400	1.117	3.999	1.539	4.862	0.725

Figure 10.13: Simulation results ($n = 2$, $m = 1$, $\rho = 0.9$, $\rho_2 \approx 2/3$)

When plotting the expected sojourn time against the standard coefficient of variance, a relation can be seen (see figure 10.14). Using regression techniques, a quadratic model is used to fit the expected sojourn time. The transfer function is given by $E[S] = 2.79 + 1.70scv + 0.313scv^2$.

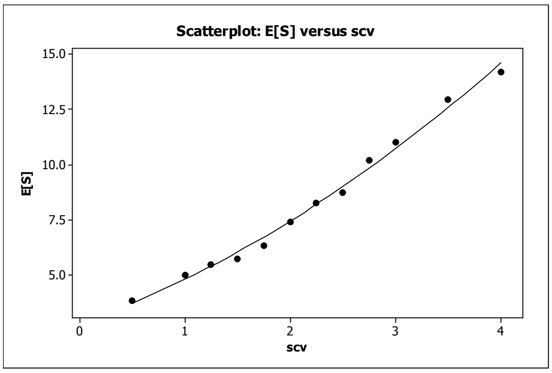


Figure 10.14: Scatterplot E[S] versus *scv* ($n = 2$, $m = 1$, $\rho = 0.9$, $\rho_2 \approx 2/3$)

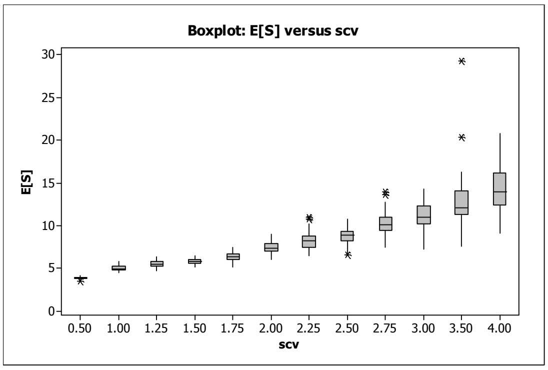
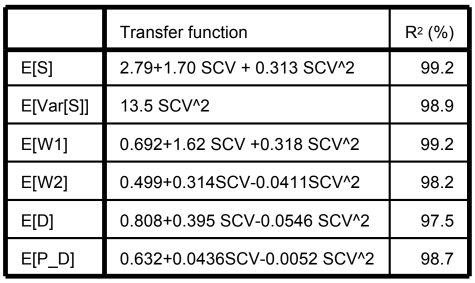


Figure 10.15: Boxplot of E[S] for different values of *scv*

During the analysis of the simulation data, it became clear that the influence of the *scv* was considerable on the expected sojourn time's standard deviation (see Figure 10.15). Analysis made clear that the 50 observations of the expected sojourn time were normally distributed (which is evidently by the central limit theorem) with a standard deviation increasing with the *scv*.
For other performance characteristics transfer functions were made too (see table 10.16).

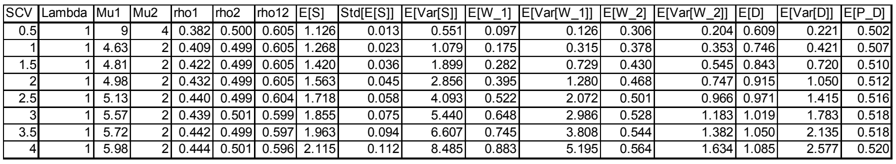


	Transfer function	R2 (%)
E[S]	2.79+1.70 SCV + 0.313 SCV^2	99.2
E[Var[S]]	13.5 SCV^2	98.9
E[W1]	0.692+1.62 SCV +0.318 SCV^2	99.2
E[W2]	0.499+0.314SCV-0.0411SCV^2	98.2
E[D]	0.808+0.395 SCV-0.0546 SCV^2	97.5
E[P_D]	0.632+0.0436SCV-0.0052 SCV^2	98.7

Figure 10.16: Transfer functions for performance characteristics ($n = 2$, $m = 1$, $\rho = 0.9$, $\rho_2 \approx 2/3$)

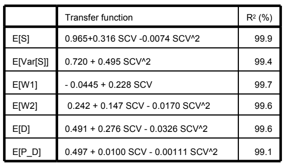
10.4.2 Tandem queue with occupancy = 0.6

Previous simulations were conducted using an occupancy of 0.9. This occupancy rate is however quite high, especially in environments other than industry. Therefore the same analysis is done with an average occupancy rate of 0.6.
Again the system is simulated for different service time distributions. The occupancy of the server in the second stage is chosen in such a way that the occupancy of the second server is as close to 0.5 as possible. By adjusting the service rate of the first server an overall system's occupancy of around 0.6 can be obtained. Again a total of 50 runs are used to estimate the appropriate performance measures where a single run is completed when 20000 customers are served.
The results are presented in the same way as before in Figure 10.17 and 10.18.



SCV	Lambda	Mu1	Mu2	rho1	rho2	rho12	E[S]	Std[E[S]]	E[Var[S]]	E[W_1]	E[Var[W_1]]	E[W_2]	E[Var[W_2]]	E[D]	E[Var[D]]	E[P_D]
0.5	1	9	4	0.382	0.500	0.605	1.126	0.013	0.551	0.097	0.126	0.306	0.204	0.609	0.221	0.502
1	1	4.63	2	0.409	0.499	0.605	1.268	0.023	1.079	0.175	0.315	0.378	0.353	0.746	0.421	0.507
1.5	1	4.81	2	0.422	0.499	0.605	1.420	0.036	1.899	0.282	0.729	0.430	0.545	0.843	0.720	0.510
2	1	4.98	2	0.432	0.499	0.605	1.563	0.045	2.856	0.395	1.280	0.468	0.747	0.915	1.050	0.512
2.5	1	5.13	2	0.440	0.499	0.604	1.718	0.058	4.093	0.522	2.072	0.501	0.966	0.971	1.415	0.516
3	1	5.57	2	0.439	0.501	0.599	1.855	0.075	5.440	0.648	2.986	0.528	1.183	1.019	1.783	0.518
3.5	1	5.72	2	0.442	0.499	0.597	1.963	0.094	6.607	0.745	3.808	0.544	1.382	1.050	2.135	0.518
4	1	5.98	2	0.444	0.501	0.596	2.115	0.112	8.485	0.883	5.195	0.564	1.634	1.085	2.577	0.520

Figure 10.17: Simulation results ($n = 2$, $m = 1$, $\rho = 0.6$, $\rho_2 \approx 1/2$)



	Transfer function	R^2 (%)
E[S]	0.965+0.316 SCV -0.0074 SCV^2	99.9
E[Var[S]]	0.720 + 0.495 SCV^2	99.4
E[W1]	- 0.0445 + 0.228 SCV	99.7
E[W2]	0.242 + 0.147 SCV - 0.0170 SCV^2	99.6
E[D]	0.491 + 0.276 SCV - 0.0326 SCV^2	99.6
E[P_D]	0.497 + 0.0100 SCV - 0.00111 SCV^2	99.1

Figure 10.18: Transfer functions for performance characteristics ($n = 2$, $m = 1$, $\rho = 0.6$, $\rho_2 \approx 1/2$)

Several remarks have to be made about this analysis. First of all, the transfer functions are based on a small number of observations only. This means that it results in rough estimates. For practical means however, it does show the correspondence between the *scv* and the performance measures. Since the fits were quite reasonable, it is expected that the value of the performance measures for intermediate *scv*'s can be estimated by these functions. This suggests the second remark, the transfer functions are only applicable (as usual) for values in the investigated range. When one wants to obtain an estimate on a performance measure when the *scv* is far out of the investigated range, the result can be highly biased.
It is obvious that the mean sojourn time in these less occupied systems is lower than the expected sojourn times earlier found.

Chapter 11

Discussion and conclusion on non-overtaking queues

In this second part of the thesis, the effect of the restriction of overtaking has been studied for parallel multiple server queues and a special type of tandem queues. The restriction on overtaking showed an expected increase in sojourn time. Especially for high occupancies, the effect of the non-overtaking property is substantial. Delays obtained due to the restriction have two negative effects.

- Waiting times: Jobs are being delayed due to the non-overtaking restriction. This means waiting time at a server for a delayed job and since the system has a higher occupancy, jobs receive longer waiting times before entering service.
- Idle times: When a job is delayed, it occupies a server without actually being served. This is idle time for the server.

The M/M/2 queue with the non-overtaking property has been investigated extensively, with the help of three methods (blocking, an iterative method, and simulations). Performance measures are obtained to determine the impact of overtaking. Queuing systems with blocking are used to derive performance measures when the number of jobs allowed to wait is low. For low occupancy systems, these measures are believed to approximate the performance well.
For an arbitrary large number of jobs which are allowed to wait, an iterative method (the Gauss-Siedel method) was used to solve the system of balance equations.
To determine the effect of multiple servers in a system, discrete event simulations was used. Also, using this method, a two stage tandem queue has been investigated more thoroughly. Regression analysis was done for various performance measures to estimate performance for other situations. An increase in variability (increasing *scv*) showed worse performance.

Problems with overtaking occur in practice. The CT scan case was an example where queuing theory was unable to capture non-overtaking with the current tools. Even dough the non-overtaking property has not been found in literature, this property has a substantial impact on the performance of relevant systems.

Epilogue

In the thesis, two problems are investigated using simulation and queuing theory. Queuing theory was unable to provide solutions for both problems because of their complexity. Recalling Figure 1 from the introduction, the steps can now be clarified.

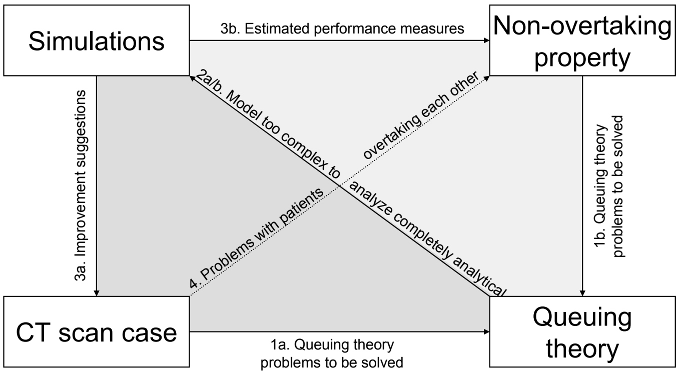

Simulations
3b. Estimated performance measures
Non-overtaking property
2a/b. Model too complex to analyze completely analytical
overtaking each other
4. Problems with patients
3a. Improvement suggestions
1b. Queuing theory problems to be solved
CT scan case
1a. Queuing theory problems to be solved
Queuing theory

Figure 1: Graphical description of reasoning behind the thesis

- 1a: The CT scan case is a scheduling problem for a stochastic system. Queuing theory can be used to model such systems. The assumptions necessary to use analytical methods only, however, are not suitable to find practical solutions.
- 2a: Discrete event simulation is much more flexible for calculating these models. Since every distribution is possible to use without making the program more complex, discrete event simulations are used to calculate the performance measures for the different schedules.
- 3a: The simulation output is used to determine a more appropriate schedule.
- 4: Patients are not allowed to overtake each other in the dressing room. When using simulations, this problem is easily avoided. Queuing theory is, however, unable to capture this behavior. The non-overtaking property is introduced in queuing theory.
- 1b: Simple systems with a very limited number of jobs allowed to wait are investigated with queuing theory techniques.
- 2b: When the models become more complex, other methods have to be sought. The Gauss-Seidel iterative method is used to estimate models with a very large number of jobs allowed to wait. Making this number of jobs arbitrary large, the performance measures are easily estimated. Discrete event simulations is used to estimate the same measures for systems when balance equations are impossible to obtain or too complex.

- 3b: The iterative method and simulations generate performance measures for queues with the non-overtaking property. The effects of this property are shown in graphs.

Bibliography

[1] Asmussen, S. (2004). *Applied Probability and Queues.* New York: Springer.

[2] Bailey, N. T. J. (1954). Queueing for medical care. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 3(3):137-145.

[3] Boland, G. W. L. (2008). Enhancing CT productivity: Strategies for increasing capacity. *American Journal of Roentgenology*, 191:3-10.

[4] Chang, C. S., Chen, Y. T., Cheng, J., Lee, D. S. (2006). Multistage constructions of linear compressors, Non-overtaking delay lines, and flexible delay lines. *Proceedings of IEEE INFOCOM 2006*, 1:11.

[5] Cayirli, T., Veral, E. (2003). Outpatient scheduling in Health Care: A review of literature. *Production and Operations Management*, 12(4):519-549.

[6] Davies, R., Davies, H. T. O. (1994). Modelling patient flows and resource provision in health systems. *Omega, International Journal of Management Science*, 22(2):123-131.

[7] Does, R. J. M. M., De Koning, H. (2003). *Six Sigma Stap voor stap.* Alphen aan den Rijn: Beaumont Quality Publications.

[8] Does, R. J. M. M., De Koning, H., De Mast, J. (2008). *Lean Six Sigma Stap voor stap.* Alphen aan den Rijn: Beaumont Quality Publications.

[9] Elkhuizen, S. G., van Sambeek, J. R., Hans, E. W., Krabbendam, K. J., Bakker, P. J. (2007). Applying the variety reduction principle to management of ancillary services. *Health Care Management Review*, 32(1):37-45.

[10] Erlang, A. K. (1909). The theory of probabilities and telephone conversations. *Nyt. Tidsskrift for Matematik B*, 20:33.

[11] Erlang, A. K. (1917). Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. *Elektrotkeknikeren*, 13:138-155.

[12] George, M. L. (2003). *Lean Six Sigma for Service: How to Use Lean Speed and Six Sigma Quality to Improve Services and Transactions.* Blacklick: McGraw-Hill.

[13] Goldratt, E. M., Cox, J. (1984). *The Goal: A Process of Ongoing Improvement.* Great Barrington: North River Press.

[14] Gorunescu, F., McClean, N. T. J., Millard, N.T.J. (2002). A queueing model for bed-occupancy management and planning of hospitals. *The Journal of the Operational Research Society*, 53(1):19-24.

[15] Green, L. V. (2003). How many hospital beds? *Inquiry*, 39:400-412.

[16] Griffiths, J. D., Price-Lloyd, N., Smithies, M., Williams, J. (2006). A queueing model of activities in an intensive care unit. *Journal of Management Mathematics*, 17:277-288.

[17] Harry, M., Schroeder, R. (1999). *Six Sigma The Breakthrough Management Strategy Revolutionizing the World's Top Corporations*. New York: Currency/Doubleday.

[18] Heroverwegingen. *Min. van Financië*,
http://www.minaz.nl/dsc?c=getobject&s=obj&objectid=123453.

[19] Ivatts, S., Millard, P. (2002). Health care modelling - why should we try? *British Journal of Health Care Management*, 8(6):218-222.

[20] Ivatts, S., Millard, P. (2002). Health care modelling: opening the 'black box'. *British Journal of Health Care Management*, 8(7):251-255.

[21] Jackson, J. R. (1963). Jobshop-like queueing systems. *Management Science*, 10(1):131-142.

[22] Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov Chain. *The Annals of Mathematical Statistics*, 24(3):338-354.

[23] Law, A. M. (2007). *Simulation Modeling and Analysis*. 4th ed. New York:McGraw-Hill.

[24] Little, J. D. C. (1961). A proof of the queuing formula: $L = \lambda W$. *Operations Research*, 9(3):383-387.

[25] Mahajan, P. S., Ingalls, R. G. (2004). Evaluation of methods to detect warm-up period in steady state simulation. *Proceedings of the 2004 Winter Simulation Conference*, 663-671.

[26] De Mast, J., Does, R. J. M. M., De Koning, H. (2006). *Lean Six Sigma for Service and Healthcare*. Alphen aan den Rijn: Beaumont Quality Publications.

[27] Minitab 15. *Minitab Inc.*, www.minitab.com.

[28] Mot, E. (2003). *Care for Competition: An Analysis of the New Health Care System*. The Hague: CPB.

[29] Neuts, M. F., Ingalls, R. G. (1970). Two servers in series, studied in terms of a Markov renewal branching process. *Advances in Applied Probability*, 2(1):110-149.

[30] Norris, J. R. (2004). *Markov Chains*. Cambridge: Cambridge University Press.

[31] Pinedo, M., Wolff, R. W. (1982). A comparison between tandem queues with dependent and independent service times. *Operations Research*, 30(3):464-479.

[32] Ramakrishnan, S., Nagarkar, K., DeGennaro, M., Srihari, K., Courtney, A. K., Emick, F. (2004). A study of the CT scan area of a healthcare provider. *Proceeding of the 2004 Winter Simulation Conference*, 2025-2031.

[33] Reinus, W. R. (2000). A proposed scheduling model to improve use of computed tomography facilities. *Journal of Medical Systems*, 24(2):61-76.

[34] Rhea, J. T., Thrall, J. H., Saini, S., Sumner, R. T. J. (1994). Improving the efficiency and service of computed tomographic scanning. *Academic Radiology*, 1:164-170.

[35] Ross, S. M. (1996). *Stochastic Processes.* New York: Wiley.

[36] Ross, S. M. (2006). *Simulation.* Boston: Elsevier Academic Press.

[37] Ross, S. M. (2007). *Introduction to Probability Models.* Boston: Elsevier Academic Press.

[38] Social report 2008 DZ. *Deventer Ziekenhuis.* http://www.dz.nl/beeldenbank/File/downloads/jaardocument.pdf.

[39] Tijms, H. C. (1986). *Stochastic Modelling and Analysis: A Computational Approach.* New York: Wiley.

[40] Vermeulen, I. B., Bohte, S. M., Elkhuizen, S. G., Lameris, J. S., Bakker, P. J. M., La Poutr, J. A. (2007). Adaptive optimization of hospital resource calendars. *Lecture Notes in Computer Science*, 4594:305-314.

[41] Walrand, J., Varaiya, P. (1980). Sojourn times and the overtaking condition in Jacksonian networks. *Advances in Applied Probability*, 12(4):1000-1018.

[42] Wang, P. P. (1999). Sequencing and scheduling N customers for a stochastic server. *European Journal of Operations Research*, 119:729-738.

[43] Welch, P. D. (1964). On a generalized M/G/1 queuing process in which the first customer of each busy period receives exceptional service. *Operations Research*, 12(5):736-752.

[44] Welch, J. D., Bailey, N. T. J. (1952). Appointment systems in hospital outpatient departments. *The Lancet*, 259(6718):1105-1108.

[45] Whitt, W. (1984). The amount of overtaking in a network of queues. *Networks*, 14(3):411-426.

[46] Wolff, R. W. (1982). Poisson arrivals see time averages. *Operations Research*, 30(2):223-231.

[47] Worthington, D. J. (1987) Queueing models for hospital waiting lists. *The Journal of the Operational Research Society*, 38(5):413-422.

Chapter 12

Appendix

Notation in CT scan simulation

Notation: decision variables

- $npday$ = number of patients scheduled for the working day.
- t_{start} = start time of the working day, CT scan is available from this time.
- t_{end} = end time of the working day, it is possible to exceed this time (overtime).
- $schedule$ = working day schedule.
- nDR = number of dressing rooms.
- T_i = scheduled time of patient i.
- $outp_i$ = indicator variable for patient type of patient i (1 if patient is an out-patient, 0 otherwise).
- IVp_i = indicator variable for the need of contrast of patient i (1 if patient needs contrast, 0 otherwise).
- $examp_i$ = indicator variable whether patient i needs a special exam (1 if patient needs an IVP or colon exam, 0 otherwise).
- $diagp_i$ = indicator variable whether patient i needs a direct diagnosis (1 if patient needs a direct diagnosis, 0 otherwise).
- $2ndscanp_i$ = indicator variable whether patient i needs a second scan (1 if patient needs a second scan, 0 otherwise)
- ave_wDR = average waiting time allowed in dressing room.
- ave_wPR = average waiting time allowed in preparation room.
- ave_rscan = bias of the medical staff's estimation on residual scanning time.
- var_rscan = variance of residual waiting time estimation by medial staff.
- $Epat\_rate$ = rate of urgent patient arrivals.

Notation: stochastic variables

- $punc_{outp} \sim$ 3-par-lognormal$(3.528, 0.5017, -19.09)$ = Out-patients' punctuality in minutes to early.
- $punc_{inp} \sim$ 3-par-lognormal$(3.104, 0.7742, -14.05)$ = In-patients' punctuality in minutes to early.
- aWR = arrival time in waiting room.
- aDR = arrival time in dressing room.
- aPR = arrival time in preparation room.
- aSR = arrival time in scan room.
- dSR = departure time from scan room.
- $aDR2$ = arrival time in dressing room after scanning.
- $dDR2$ = departure time from dressing room after getting dressed.
- $DR_{outp} \sim$ lognormal$(0.3466, 0.9102)$= time between entrance in dressing room and entrance in scan room for an out-patient.
- $IV_{outp} \sim$ lognormal$(0.9044, 0.4560)$ = time it takes to install the IV on an out-patient.
- $IV_{inp} \sim$ lognormal$(0.3148, 0.4142)$ = time it takes to install the IV on an in-patient.
- dDR = departure time from DR after getting dressed, patient leaves th radiology department .

Notation: measures

- $W_{WR} = aDR - aWR$ waiting time of patient in the waiting room.
- $W_{DR} = \min(aPR, aSR) - aDR - c\_t$ waiting time of patient in the dressing room.
- $W_{PR} = aR - aPR - IV\_t$ waiting time of patient in the preparation room.
- other measures are described in 2.1.3.

12.1 Graphical review data analysis

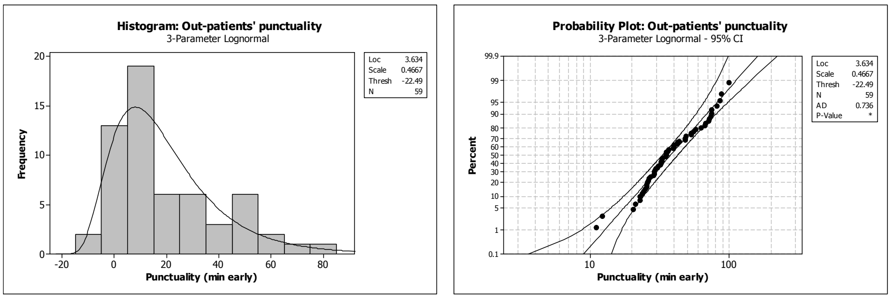


Figure 12.1: Data analysis of out-patients' punctuality

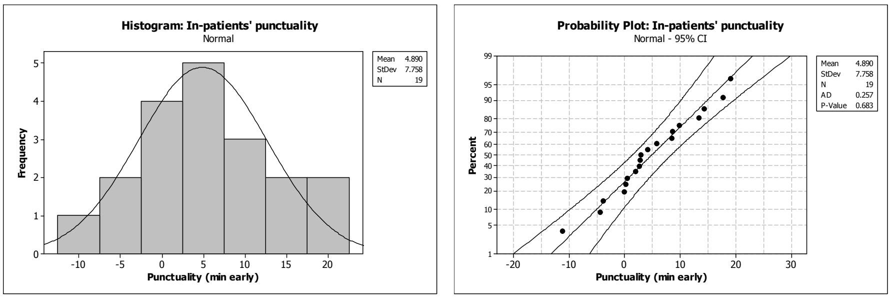


Figure 12.2: Data analysis of in-patients' punctuality

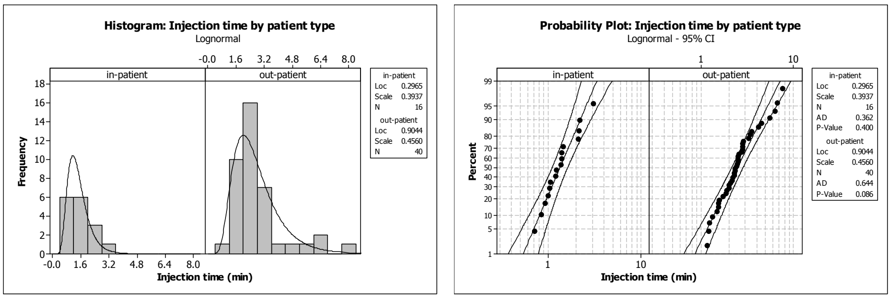


Figure 12.3: Data analysis of injection time

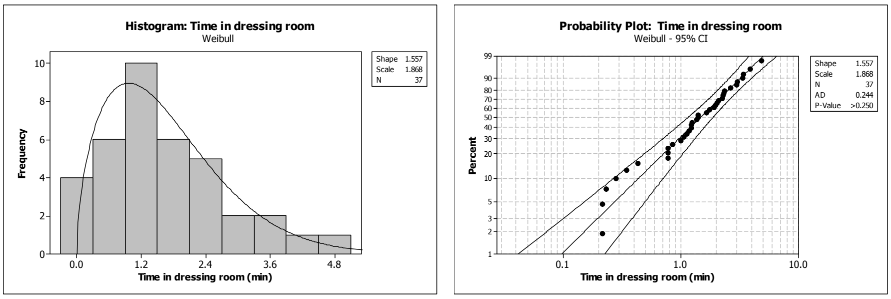
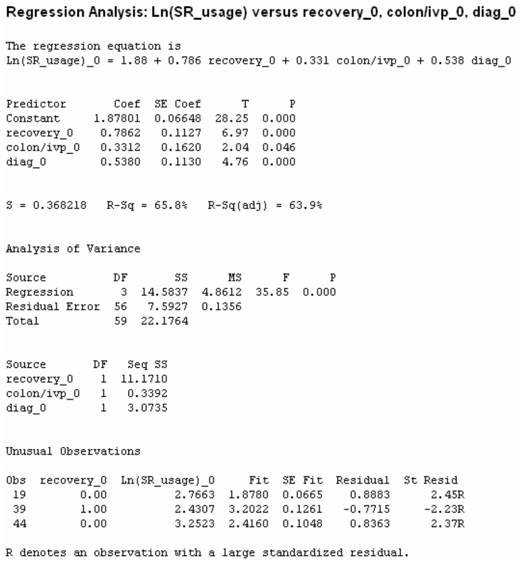


Figure 12.4: Data analysis of time spent in dressing room by out-patients



```
Regression Analysis: Ln(SR_usage) versus recovery_0, colon/ivp_0, diag_0

The regression equation is
Ln(SR_usage)_0 = 1.88 + 0.786 recovery_0 + 0.331 colon/ivp_0 + 0.538 diag_0


Predictor        Coef  SE Coef      T      P
Constant      1.87801  0.06648  28.25  0.000
recovery_0     0.7862   0.1127   6.97  0.000
colon/ivp_0    0.3312   0.1620   2.04  0.046
diag_0         0.5380   0.1130   4.76  0.000


S = 0.368218   R-Sq = 65.8%   R-Sq(adj) = 63.9%


Analysis of Variance

Source          DF       SS      MS      F      P
Regression       3  14.5837  4.8612  35.85  0.000
Residual Error  56   7.5927  0.1356
Total           59  22.1764


Source       DF   Seq SS
recovery_0    1  11.1710
colon/ivp_0   1   0.3392
diag_0        1   3.0735


Unusual Observations

Obs  recovery_0  Ln(SR_usage)_0     Fit  SE Fit  Residual  St Resid
 19        0.00          2.7663  1.8780  0.0665    0.8883      2.45R
 39        1.00          2.4307  3.2022  0.1261   -0.7715     -2.23R
 44        0.00          3.2523  2.4160  0.1048    0.8363      2.37R

R denotes an observation with a large standardized residual.
```

Figure 12.5: Regression for Ln(time spent in scan room)

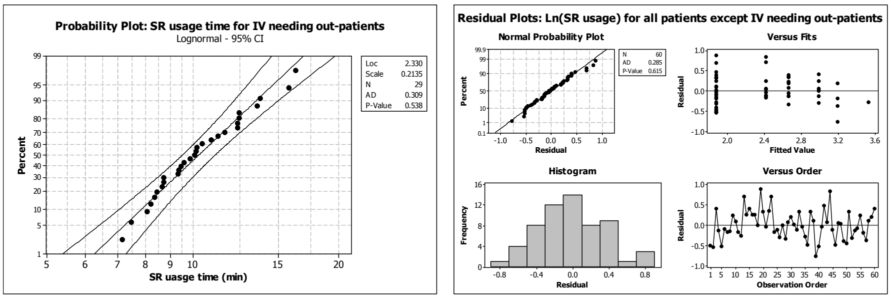


Figure 12.6: Probability plot for scan room usage of IV out-patients and residual analysis of Ln(Scan room usage) other patients

12.2 Schedules for the CT scan case

Appointment time	8:15	8:30	8:45	9:00	9:15	9:30	9:45	10:00	10:15	10:30	10:45	11:00	11:15	11:30	11:45
patient type	1	1	1	E-Slot	1	1	1	1	Break	1	1	1	1	1	1
contrast	0	1	1		1	0	0	1		1	1	0	1	0	1
#scans	1	1	1		2	1	1	1		1	2	1	1	1	1
special exam	0	0	0		0	0	0	0		0	0	0	0	0	0
diagnose	0	0	0		0	0	0	0		0	0	0	0	0	0
Appointment time	12:00	12:15	12:30	13:00	13:15	13:30	13:45	14:00	14:20	14:40	15:00	15:20	15:40	16:00	
patient type	E-Slot	E-Slot	Break	1	1	1	1	0	0	0	0	0	0	E-Slot	
contrast				1	1	1	1	1	1	0	1	0	0		
#scans				2	1	1	1	2	1	1	1	1	1		
special exam				1	1	0	0	0	0	0	0	0	0		
diagnose				0	0	0	0	0	0	0	1	1	0		

Figure 12.7: Current schedule (S_C)

Appointment time	8:15	8:25	8:35	8:45	8:55	9:05	9:15	9:25	9:35	9:45	9:55	10:05	10:15	10:30	10:40	10:50	11:00	11:10	11:20
patient type	1	1	1	1	1	E-Slot	1	1	1	1	1	E-Slot	Break	1	1	1	1	1	E-Slot
contrast	1	1	1	0	1		0	1	1	1	0			1	0	1	0	1	
#scans	1	1	2	1	1		1	1	1	1	1			2	1	1	1	1	
special exam	0	0	0	0	0		0	0	0	0	0			0	0	0	0	0	
diagnose	0	0	0	0	0		0	0	0	0	0			0	0	0	0	0	
Appointment time	11:30	11:40	11:50	12:00	12:10	12:20	12:30	13:00	13:20	13:40	14:00	14:20	14:40	14:50	15:10	15:30	15:50	16:10	16:20
patient type	1	1	1	1	1	E-Slot	Break	1	1	0	0	0	E-Slot	0	0	0	0	0	E-Slot
contrast	1	0	1	0	1			1	0	1	1	1		1	0	1	0	0	
#scans	1	1	1	1	1			2	1	1	1	2		1	1	1	1	1	
special exam	0	0	0	0	0			1	1	0	0	0		0	0	0	0	0	
diagnose	0	0	0	0	0			0	0	1	0	0		0	0	1	1	0	

Figure 12.8: Proposed schedule 1 (S_{P1})

Appointment time	8:15	8:25	8:35	8:45	8:55	9:05	9:15	9:25	9:35	9:45	9:55	10:05	10:15	10:35	10:45	10:55	11:05	11:15
patient type	1	1	1	1	1	E-Slot	1	1	1	1	1	E-Slot	Break	1	1	1	1	1
contrast	0	1	1	0	1		0	1	0	1	1			0	1	1	1	1
#scans	1	1	1	1	1		1	2	1	1	1			1	1	1	1	2
special exam	0	0	0	0	0		0	0	0	0	0			0	0	0	0	1
diagnose	0	0	0	0	0		0	0	0	0	0			0	0	0	0	0
Appointment time	11:35	11:55	12:15	12:30	13:00	13:10	13:20	13:30	13:40	13:50	14:00	14:20	14:40	15:00	15:20	15:30	15:50	16:10
patient type	0	1	E-Slot	Break	1	1	1	1	1	1	0	0	0	0	E-Slot	0	0	E-Slot
contrast	1	1			0	1	1	1	0	1	1	1	1	0		1	0	
#scans	1	1			1	1	1	2	1	1	1	1	1	1		1	1	
special exam	1	1			0	0	0	0	0	0	0	0	0	0		0	0	
diagnose	0	0			0	0	0	0	0	0	1	0	1	1		0	0	

Figure 12.9: Proposed schedule 2 (S_{P2})

Appointment time	8:15	8:35	8:55	9:15	9:25	9:35	10:05	10:15	10:30	10:50	11:10	11:30	11:50	12:00	12:10	12:20	12:30	13:00	13:10
patient type	0	0	0	0	0	0	E-Slot	Break	0	0	1	1	E-Slot	1	1	1	Break	1	1
contrast	1	1	1	1	0	1			0	0	0	1		1	1	0		1	0
#scans	1	1	2	1	1	1			1	1	1	1		1	1	1		1	1
special exam	0	0	0	0	0	0			0	0	1	2		0	0	0		0	0
diagnose	1	0	0	0	0	1			1	0	0	0		0	0	0		0	0
Appointment time	13:20	13:30	13:40	13:50	14:00	14:10	14:20	14:30	14:40	14:50	15:00	15:10	15:20	15:30	15:40	15:50	16:00	16:20	
patient type	1	1	1	E-Slot	1	1	1	1	1	1	E-Slot	1	1	1	1	1	1	E-Slot	
contrast	1	0	1		0	1	0	1	0	1		0	1	0	1	1	0		
#scans	2	1	1		1	2	1	1	1	1		1	1	1	1	1	1		
special exam	0	0	0		0	0	0	0	0	0		0	0	0	0	0	0		
diagnose	0	0	0		0	0	0	0	0	0		0	0	0	0	0	0		

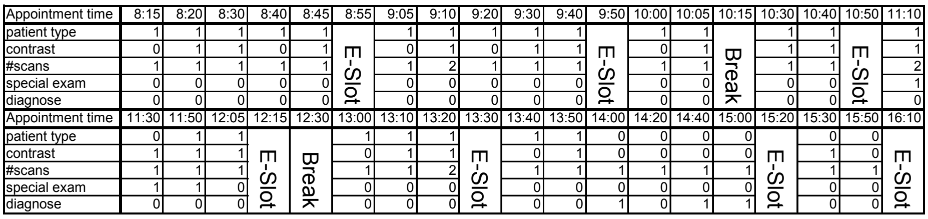
Figure 12.10: Reversed proposed schedule 1 (S_{R-P})

Appointment time	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15	8:15
patient type	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
contrast	1	1	1	0	1	0	1	1	1	0	1	0	1	0	1
#scans	1	1	2	1	1	1	1	1	1	1	2	1	1	1	1
special exam	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
diagnose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Appointment time	8:15	8:15	8:15	8:15	8:15	13:00	13:00	13:00	13:00	13:00	13:00	13:00	13:00	13:00	13:00
patient type	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
contrast	1	0	1	0	1	1	0	1	1	1	1	0	1	0	0
#scans	1	1	1	1	1	2	1	1	1	2	1	1	1	1	1
special exam	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
diagnose	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0

Figure 12.11: 2 Block schedule (S_{2B})

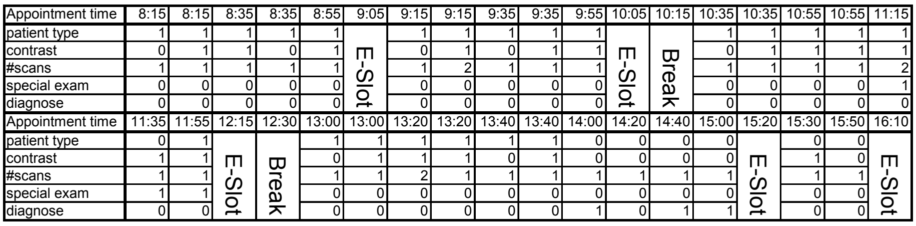
Appointment time	8:15	8:25	8:40	8:50	9:05	9:15	9:25	9:40	9:50	10:05	10:30	10:40	10:55	11:05	11:20
patient type	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
contrast	1	1	1	0	1	0	1	1	1	0	1	0	1	0	1
#scans	1	1	2	1	1	1	1	1	1	1	2	1	1	1	1
special exam	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
diagnose	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Appointment time	11:30	11:40	11:55	12:05	12:20	13:00	13:20	13:40	14:00	14:20	14:40	15:00	15:20	15:40	16:00
patient type	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
contrast	1	0	1	0	1	1	0	1	1	1	1	0	1	0	0
#scans	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1
special exam	0	0	0	0	0	2	1	0	0	0	0	0	0	0	0
diagnose	0	0	0	0	0	0	0	1	0	0	0	0	1	1	0

Figure 12.12: No emergency block schedule (S_{P1} noE)



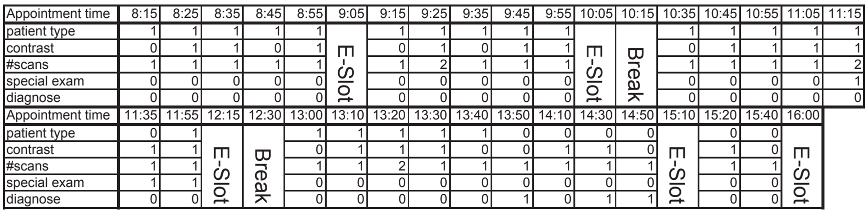
Appointment time	8:15	8:20	8:30	8:40	8:45	8:55	9:05	9:10	9:20	9:30	9:40	9:50	10:00	10:05	10:15	10:30	10:40	10:50	11:10
patient type	1	1	1	1	1	E-Slot	1	1	1	1	1	E-Slot	1	1	Break	1	1	E-Slot	1
contrast	0	1	1	0	1		0	1	0	1	1		0	1		1	1		1
#scans	1	1	1	1	1		1	2	1	1	1		1	1		1	1		2
special exam	0	0	0	0	0		0	0	0	0	0		0	0		0	0		1
diagnose	0	0	0	0	0		0	0	0	0	0		0	0		0	0		0
Appointment time	11:30	11:50	12:05	12:15	12:30	13:00	13:10	13:20	13:30	13:40	13:50	14:00	14:20	14:40	15:00	15:20	15:30	15:50	16:10
patient type	0	1	1	E-Slot	Break	1	1	1	E-Slot	1	1	0	0	0	0	E-Slot	0	0	E-Slot
contrast	1	1	1			0	1	1		0	1	0	0	0	0		1	0	
#scans	1	1	1			1	1	2		1	1	1	1	1	1		1	1	
special exam	1	1	0			0	0	0		0	0	0	0	0	0		0	0	
diagnose	0	0	0			0	0	0		0	0	1	0	1	1		0	0	

Figure 12.13: Variable interval lengths (S_V)



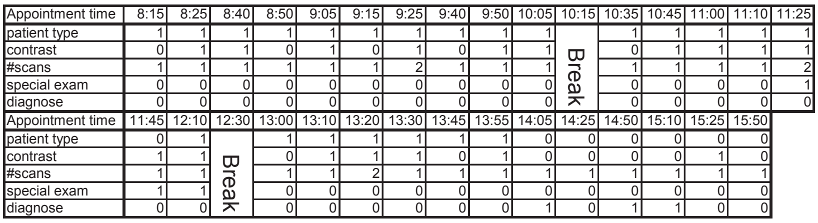
Appointment time	8:15	8:15	8:35	8:35	8:55	9:05	9:15	9:15	9:35	9:35	9:55	10:05	10:15	10:35	10:35	10:55	10:55	11:15
patient type	1	1	1	1	1	E-Slot	1	1	1	1	1	E-Slot	Break	1	1	1	1	1
contrast	0	1	1	0	1		0	1	0	1	1			0	1	1	1	1
#scans	1	1	1	1	1		1	2	1	1	1			1	1	1	1	2
special exam	0	0	0	0	0		0	0	0	0	0			0	0	0	0	1
diagnose	0	0	0	0	0		0	0	0	0	0			0	0	0	0	0
Appointment time	11:35	11:55	12:15	12:30	13:00	13:00	13:20	13:20	13:40	13:40	14:00	14:20	14:40	15:00	15:20	15:30	15:50	16:10
patient type	0	1	E-Slot	Break	1	1	1	1	1	1	0	0	0	0	E-Slot	0	0	E-Slot
contrast	1	1			0	1	1	1	0	1	0	0	0	0		1	0	
#scans	1	1			1	1	2	1	1	1	1	1	1	1		1	1	
special exam	1	1			0	0	0	0	0	0	0	0	0	0		0	0	
diagnose	0	0			0	0	0	0	0	0	1	0	1	1		0	0	

Figure 12.14: Multiple number of patients per slot (S_{P2})



Appointment time	8:15	8:25	8:35	8:45	8:55	9:05	9:15	9:25	9:35	9:45	9:55	10:05	10:15	10:35	10:45	10:55	11:05	11:15
patient type	1	1	1	1	1	E-Slot	1	1	1	1	1	E-Slot	Break	1	1	1	1	1
contrast	0	1	1	0	1		0	1	0	1	1			0	1	1	1	1
#scans	1	1	1	1	1		1	2	1	1	1			1	1	1	1	2
special exam	0	0	0	0	0		0	0	0	0	0			0	0	0	0	1
diagnose	0	0	0	0	0		0	0	0	0	0			0	0	0	0	0
Appointment time	11:35	11:55	12:15	12:30	13:00	13:10	13:20	13:30	13:40	13:50	14:10	14:30	14:50	15:10	15:20	15:40	16:00	
patient type	0	1	E-Slot	Break	1	1	1	1	1	0	0	0	0	E-Slot	0	0	E-Slot	
contrast	1	1			0	1	1	1	0	0	1	1	0		1	0		
#scans	1	1			1	1	2	1	1	1	1	1	1		1	1		
special exam	1	1			0	0	0	0	0	0	0	0	0		0	0		
diagnose	0	0			0	0	0	0	0	1	0	1	1		0	0		

Figure 12.15: Second proposed schedule with 1 out-patient less ($S_{P2}\ -1\ outp$)



Appointment time	8:15	8:25	8:40	8:50	9:05	9:15	9:25	9:40	9:50	10:05	10:15	10:35	10:45	11:00	11:10	11:25
patient type	1	1	1	1	1	1	1	1	1	1	Break	1	1	1	1	1
contrast	0	1	1	0	1	0	1	0	1	1		0	1	1	1	1
#scans	1	1	1	1	1	1	2	1	1	1		1	1	1	1	2
special exam	0	0	0	0	0	0	0	0	0	0		0	0	0	0	1
diagnose	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0
Appointment time	11:45	12:10	12:30	13:00	13:10	13:20	13:30	13:45	13:55	14:05	14:25	14:50	15:10	15:25	15:50	
patient type	0	1	Break	1	1	1	1	1	1	0	0	0	0	0	0	
contrast	1	1		0	1	1	1	0	1	0	0	0	0	1	0	
#scans	1	1		1	1	2	1	1	1	1	1	1	1	1	1	
special exam	1	1		0	0	0	0	0	0	0	0	0	0	0	0	
diagnose	0	0		0	0	0	0	0	0	1	0	1	1	0	0	

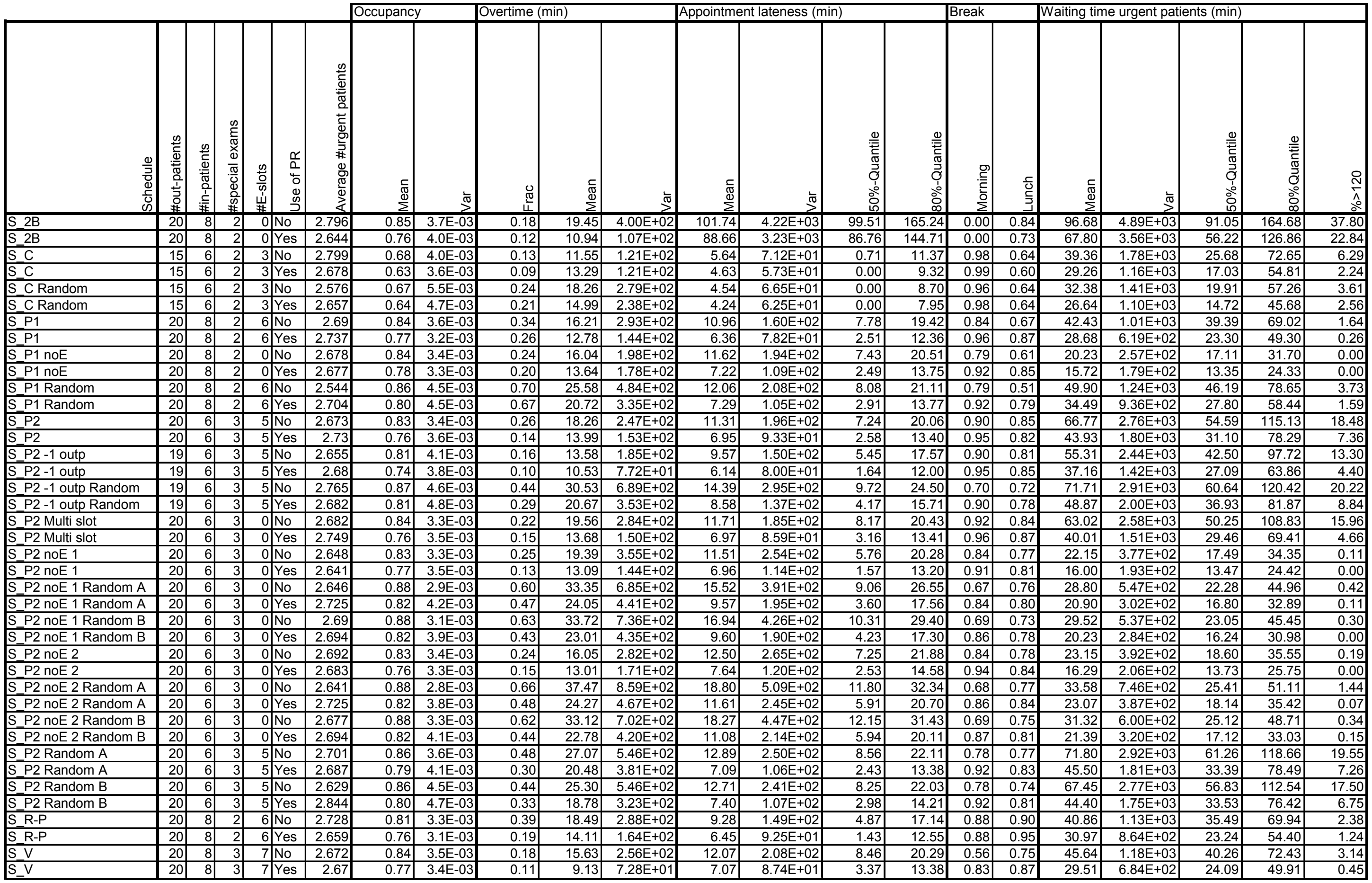
Figure 12.16: Second proposed schedule with no emergency slots version 1 ($S_{P2}\ NoE\ 1$)

Appointment time	8:15	8:15	8:40	8:40	9:05	9:05	9:30	9:30	9:55	9:55	10:15	10:35	10:35	11:00	11:00	11:25
patient type	1	1	1	1	1	1	1	1	1	1	Break	1	1	1	1	1
contrast	0	1	1	0	1	0	1	0	1	1		0	1	1	1	1
#scans	1	1	1	1	1	1	2	1	1	1		1	1	1	1	2
special exam	0	0	0	0	0	0	0	0	0	0		0	0	0	0	1
diagnose	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0

Appointment time	11:25	12:05	12:30	13:00	13:00	13:25	13:25	13:50	13:50	14:15	14:15	14:55	14:55	15:35	15:35
patient type	0	1	Break	1	1	1	1	1	1	0	0	0	0	0	0
contrast	1	1		0	1	1	1	0	1	0	1	1	0	1	0
#scans	1	1		1	1	2	1	1	1	1	1	1	1	1	1
special exam	1	1		0	0	0	0	0	0	0	0	0	0	0	0
diagnose	0	0		0	0	0	0	0	0	1	0	1	1	0	0

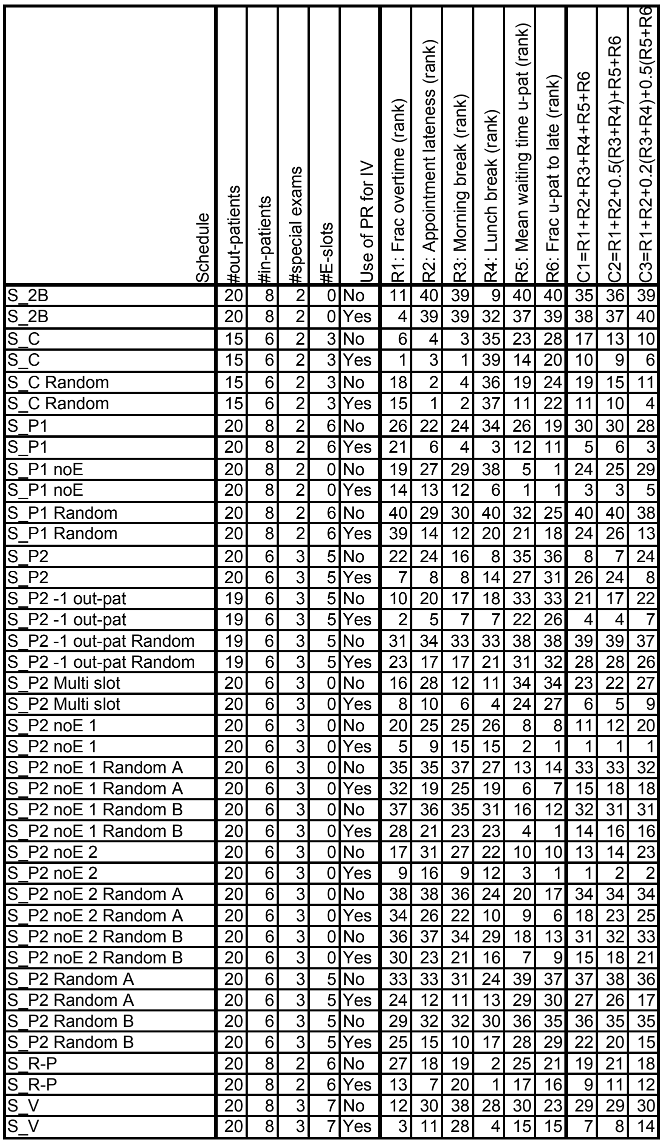
Figure 12.17: Second proposed schedule with no emergency slots version 2 (S_{P2} NoE 2)

12.3 Simulation results of the CT scan case



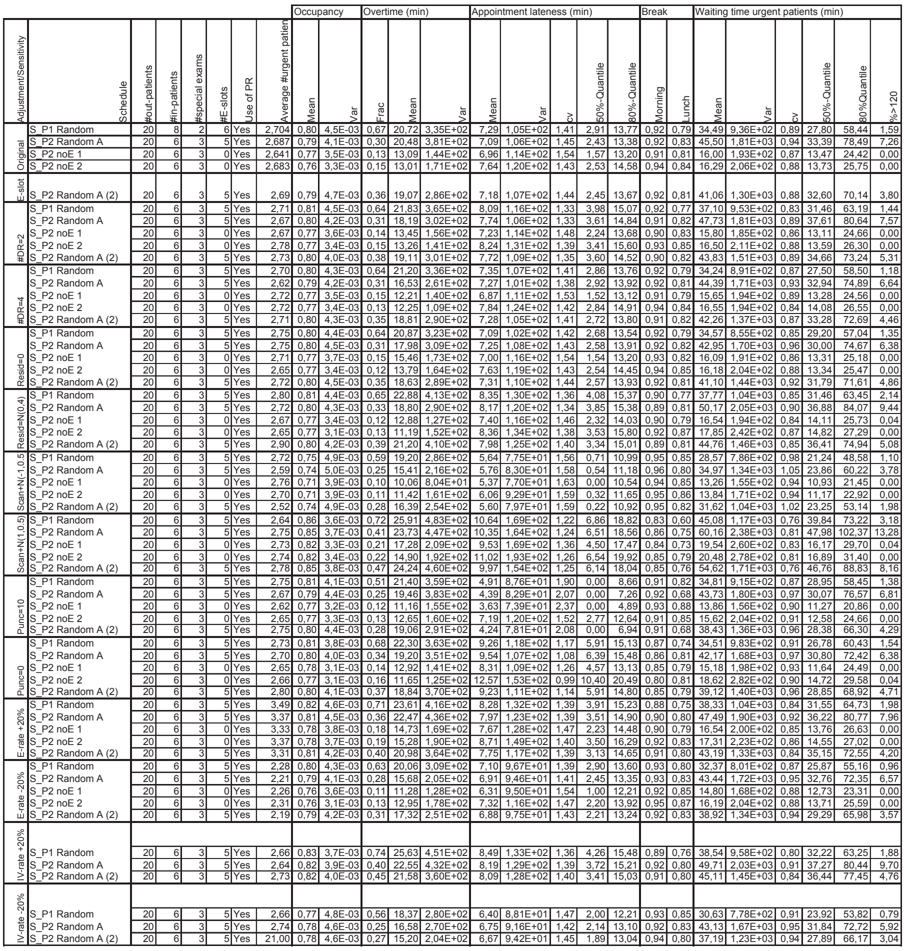
							Occupancy		Overtime (min)			Appointment lateness (min)				Break		Waiting time urgent patients (min)				
Schedule	#out-patients	#in-patients	#special exams	#E-slots	Use of PR	Average #urgent patients	Mean	Var	Frac	Mean	Var	Mean	Var	50%-Quantile	80%-Quantile	Morning	Lunch	Mean	Var	50%-Quantile	80%Quantile	%>120
S_2B	20	8	2	0	No	2.796	0.85	3.7E-03	0.18	19.45	4.00E+02	101.74	4.22E+03	99.51	165.24	0.00	0.84	96.68	4.89E+03	91.05	164.68	37.80
S_2B	20	8	2	0	Yes	2.644	0.76	4.0E-03	0.12	10.94	1.07E+02	88.66	3.23E+03	86.76	144.71	0.00	0.73	67.80	3.56E+03	56.22	126.86	22.84
S_C	15	6	2	3	No	2.799	0.68	4.0E-03	0.13	11.55	1.21E+02	5.64	7.12E+01	0.71	11.37	0.98	0.64	39.36	1.78E+03	25.68	72.65	6.29
S_C	15	6	2	3	Yes	2.678	0.63	3.6E-03	0.09	13.29	1.21E+02	4.63	5.73E+01	0.00	9.32	0.99	0.60	29.26	1.16E+03	17.03	54.81	2.24
S_C Random	15	6	2	3	No	2.576	0.67	5.5E-03	0.24	18.26	2.79E+02	4.54	6.65E+01	0.00	8.70	0.96	0.64	32.38	1.41E+03	19.91	57.26	3.61
S_C Random	15	6	2	3	Yes	2.657	0.64	4.7E-03	0.21	14.99	2.38E+02	4.24	6.25E+01	0.00	7.95	0.98	0.64	26.64	1.10E+03	14.72	45.68	2.56
S_P1	20	8	2	6	No	2.69	0.84	3.6E-03	0.34	16.21	2.93E+02	10.96	1.60E+02	7.78	19.42	0.84	0.67	42.43	1.01E+03	39.39	69.02	1.64
S_P1	20	8	2	6	Yes	2.737	0.77	3.2E-03	0.26	12.78	1.44E+02	6.36	7.82E+01	2.51	12.36	0.96	0.87	28.68	6.19E+02	23.30	49.30	0.26
S_P1 noE	20	8	2	0	No	2.678	0.84	3.4E-03	0.24	16.04	1.98E+02	11.62	1.94E+02	7.43	20.51	0.79	0.61	20.23	2.57E+02	17.11	31.70	0.00
S_P1 noE	20	8	2	0	Yes	2.677	0.78	3.3E-03	0.20	13.64	1.78E+02	7.22	1.09E+02	2.49	13.75	0.92	0.85	15.72	1.79E+02	13.35	24.33	0.00
S_P1 Random	20	8	2	6	No	2.544	0.86	4.5E-03	0.70	25.58	4.84E+02	12.06	2.08E+02	8.08	21.11	0.79	0.51	49.90	1.24E+03	46.19	78.65	3.73
S_P1 Random	20	8	2	6	Yes	2.704	0.80	4.5E-03	0.67	20.72	3.35E+02	7.29	1.05E+02	2.91	13.77	0.92	0.79	34.49	9.36E+02	27.80	58.44	1.59
S_P2	20	6	3	5	No	2.673	0.83	3.4E-03	0.26	18.26	2.47E+02	11.31	1.96E+02	7.24	20.06	0.90	0.85	66.77	2.76E+03	54.59	115.13	18.48
S_P2	20	6	3	5	Yes	2.73	0.76	3.6E-03	0.14	13.99	1.53E+02	6.95	9.33E+01	2.58	13.40	0.95	0.82	43.93	1.80E+03	31.10	78.29	7.36
S_P2 -1 outp	19	6	3	5	No	2.655	0.81	4.1E-03	0.16	13.58	1.85E+02	9.57	1.50E+02	5.45	17.57	0.90	0.81	55.31	2.44E+03	42.50	97.72	13.30
S_P2 -1 outp	19	6	3	5	Yes	2.68	0.74	3.8E-03	0.10	10.53	7.72E+01	6.14	8.00E+01	1.64	12.00	0.95	0.85	37.16	1.42E+03	27.09	63.86	4.40
S_P2 -1 outp Random	19	6	3	5	No	2.765	0.87	4.6E-03	0.44	30.53	6.89E+02	14.39	2.95E+02	9.72	24.50	0.70	0.72	71.71	2.91E+03	60.64	120.42	20.22
S_P2 -1 outp Random	19	6	3	5	Yes	2.682	0.81	4.8E-03	0.29	20.67	3.53E+02	8.58	1.37E+02	4.17	15.71	0.90	0.78	48.87	2.00E+03	36.93	81.87	8.84
S_P2 Multi slot	20	6	3	0	No	2.682	0.84	3.3E-03	0.22	19.56	2.84E+02	11.71	1.85E+02	8.17	20.43	0.92	0.84	63.02	2.58E+03	50.25	108.83	15.96
S_P2 Multi slot	20	6	3	0	Yes	2.749	0.76	3.5E-03	0.15	13.68	1.50E+02	6.97	8.59E+01	3.16	13.41	0.96	0.87	40.01	1.51E+03	29.46	69.41	4.66
S_P2 noE 1	20	6	3	0	No	2.648	0.83	3.3E-03	0.25	19.39	3.55E+02	11.51	2.54E+02	5.76	20.28	0.84	0.77	22.15	3.77E+02	17.49	34.35	0.11
S_P2 noE 1	20	6	3	0	Yes	2.641	0.77	3.5E-03	0.13	13.09	1.44E+02	6.96	1.14E+02	1.57	13.20	0.91	0.81	16.00	1.93E+02	13.47	24.42	0.00
S_P2 noE 1 Random A	20	6	3	0	No	2.646	0.88	2.9E-03	0.60	33.35	6.85E+02	15.52	3.91E+02	9.06	26.55	0.67	0.76	28.80	5.47E+02	22.28	44.96	0.42
S_P2 noE 1 Random A	20	6	3	0	Yes	2.725	0.82	4.2E-03	0.47	24.05	4.41E+02	9.57	1.95E+02	3.60	17.56	0.84	0.80	20.90	3.02E+02	16.80	32.89	0.11
S_P2 noE 1 Random B	20	6	3	0	No	2.69	0.88	3.1E-03	0.63	33.72	7.36E+02	16.94	4.26E+02	10.31	29.40	0.69	0.73	29.52	5.37E+02	23.05	45.45	0.30
S_P2 noE 1 Random B	20	6	3	0	Yes	2.694	0.82	3.9E-03	0.43	23.01	4.35E+02	9.60	1.90E+02	4.23	17.30	0.86	0.78	20.23	2.84E+02	16.24	30.98	0.00
S_P2 noE 2	20	6	3	0	No	2.692	0.83	3.4E-03	0.24	16.05	2.82E+02	12.50	2.65E+02	7.25	21.88	0.84	0.78	23.15	3.92E+02	18.60	35.55	0.19
S_P2 noE 2	20	6	3	0	Yes	2.683	0.76	3.3E-03	0.15	13.01	1.71E+02	7.64	1.20E+02	2.53	14.58	0.94	0.84	16.29	2.06E+02	13.73	25.75	0.00
S_P2 noE 2 Random A	20	6	3	0	No	2.641	0.88	2.8E-03	0.66	37.47	8.59E+02	18.80	5.09E+02	11.80	32.34	0.68	0.77	33.58	7.46E+02	25.41	51.11	1.44
S_P2 noE 2 Random A	20	6	3	0	Yes	2.725	0.82	3.8E-03	0.48	24.27	4.67E+02	11.61	2.45E+02	5.91	20.70	0.86	0.84	23.07	3.87E+02	18.14	35.42	0.07
S_P2 noE 2 Random B	20	6	3	0	No	2.677	0.88	3.3E-03	0.62	33.12	7.02E+02	18.27	4.47E+02	12.15	31.43	0.69	0.75	31.32	6.00E+02	25.12	48.71	0.34
S_P2 noE 2 Random B	20	6	3	0	Yes	2.694	0.82	4.1E-03	0.44	22.78	4.20E+02	11.08	2.14E+02	5.94	20.11	0.87	0.81	21.39	3.20E+02	17.12	33.03	0.15
S_P2 Random A	20	6	3	5	No	2.701	0.86	3.6E-03	0.48	27.07	5.46E+02	12.89	2.50E+02	8.56	22.11	0.78	0.77	71.80	2.92E+03	61.26	118.66	19.55
S_P2 Random A	20	6	3	5	Yes	2.687	0.79	4.1E-03	0.30	20.48	3.81E+02	7.09	1.06E+02	2.43	13.38	0.92	0.83	45.50	1.81E+03	33.39	78.49	7.26
S_P2 Random B	20	6	3	5	No	2.629	0.86	4.5E-03	0.44	25.30	5.46E+02	12.71	2.41E+02	8.25	22.03	0.78	0.74	67.45	2.77E+03	56.83	112.54	17.50
S_P2 Random B	20	6	3	5	Yes	2.844	0.80	4.7E-03	0.33	18.78	3.23E+02	7.40	1.07E+02	2.98	14.21	0.92	0.81	44.40	1.75E+03	33.53	76.42	6.75
S_R-P	20	8	2	6	No	2.728	0.81	3.3E-03	0.39	18.49	2.88E+02	9.28	1.49E+02	4.87	17.14	0.88	0.90	40.86	1.13E+03	35.49	69.94	2.38
S_R-P	20	8	2	6	Yes	2.659	0.76	3.1E-03	0.19	14.11	1.64E+02	6.45	9.25E+01	1.43	12.55	0.88	0.95	30.97	8.64E+02	23.24	54.40	1.24
S_V	20	8	3	7	No	2.672	0.84	3.5E-03	0.18	15.63	2.56E+02	12.07	2.08E+02	8.46	20.29	0.56	0.75	45.64	1.18E+03	40.26	72.43	3.14
S_V	20	8	3	7	Yes	2.67	0.77	3.4E-03	0.11	9.13	7.28E+01	7.07	8.74E+01	3.37	13.38	0.83	0.87	29.51	6.84E+02	24.09	49.91	0.45

Figure 12.18: Simulation results



Schedule	#out-patients	#in-patients	#special exams	#E-slots	Use of PR for IV	R1: Frac overtime (rank)	R2: Appointment lateness (rank)	R3: Morning break (rank)	R4: Lunch break (rank)	R5: Mean waiting time u-pat (rank)	R6: Frac u-pat to late (rank)	C1=R1+R2+R3+R4+R5+R6	C2=R1+R2+0.5(R3+R4)+R5+R6	C3=R1+R2+0.2(R3+R4)+0.5(R5+R6
S_2B	20	8	2	0	No	11	40	39	9	40	40	35	36	39
S_2B	20	8	2	0	Yes	4	39	39	32	37	39	38	37	40
S_C	15	6	2	3	No	6	4	3	35	23	28	17	13	10
S_C	15	6	2	3	Yes	1	3	1	39	14	20	10	9	6
S_C Random	15	6	2	3	No	18	2	4	36	19	24	19	15	11
S_C Random	15	6	2	3	Yes	15	1	2	37	11	22	11	10	4
S_P1	20	8	2	6	No	26	22	24	34	26	19	30	30	28
S_P1	20	8	2	6	Yes	21	6	4	3	12	11	5	6	3
S_P1 noE	20	8	2	0	No	19	27	29	38	5	1	24	25	29
S_P1 noE	20	8	2	0	Yes	14	13	12	6	1	1	3	3	5
S_P1 Random	20	8	2	6	No	40	29	30	40	32	25	40	40	38
S_P1 Random	20	8	2	6	Yes	39	14	12	20	21	18	24	26	13
S_P2	20	6	3	5	No	22	24	16	8	35	36	8	7	24
S_P2	20	6	3	5	Yes	7	8	8	14	27	31	26	24	8
S_P2 -1 out-pat	19	6	3	5	No	10	20	17	18	33	33	21	17	22
S_P2 -1 out-pat	19	6	3	5	Yes	2	5	7	7	22	26	4	4	7
S_P2 -1 out-pat Random	19	6	3	5	No	31	34	33	33	38	38	39	39	37
S_P2 -1 out-pat Random	19	6	3	5	Yes	23	17	17	21	31	32	28	28	26
S_P2 Multi slot	20	6	3	0	No	16	28	12	11	34	34	23	22	27
S_P2 Multi slot	20	6	3	0	Yes	8	10	6	4	24	27	6	5	9
S_P2 noE 1	20	6	3	0	No	20	25	25	26	8	8	11	12	20
S_P2 noE 1	20	6	3	0	Yes	5	9	15	15	2	1	1	1	1
S_P2 noE 1 Random A	20	6	3	0	No	35	35	37	27	13	14	33	33	32
S_P2 noE 1 Random A	20	6	3	0	Yes	32	19	25	19	6	7	15	18	18
S_P2 noE 1 Random B	20	6	3	0	No	37	36	35	31	16	12	32	31	31
S_P2 noE 1 Random B	20	6	3	0	Yes	28	21	23	23	4	1	14	16	16
S_P2 noE 2	20	6	3	0	No	17	31	27	22	10	10	13	14	23
S_P2 noE 2	20	6	3	0	Yes	9	16	9	12	3	1	1	2	2
S_P2 noE 2 Random A	20	6	3	0	No	38	38	36	24	20	17	34	34	34
S_P2 noE 2 Random A	20	6	3	0	Yes	34	26	22	10	9	6	18	23	25
S_P2 noE 2 Random B	20	6	3	0	No	36	37	34	29	18	13	31	32	33
S_P2 noE 2 Random B	20	6	3	0	Yes	30	23	21	16	7	9	15	18	21
S_P2 Random A	20	6	3	5	No	33	33	31	24	39	37	37	38	36
S_P2 Random A	20	6	3	5	Yes	24	12	11	13	29	30	27	26	17
S_P2 Random B	20	6	3	5	No	29	32	32	30	36	35	36	35	35
S_P2 Random B	20	6	3	5	Yes	25	15	10	17	28	29	22	20	15
S_R-P	20	8	2	6	No	27	18	19	2	25	21	19	21	18
S_R-P	20	8	2	6	Yes	13	7	20	1	17	16	9	11	12
S_V	20	8	3	7	No	12	30	38	28	30	23	29	29	30
S_V	20	8	3	7	Yes	3	11	28	4	15	15	7	8	14

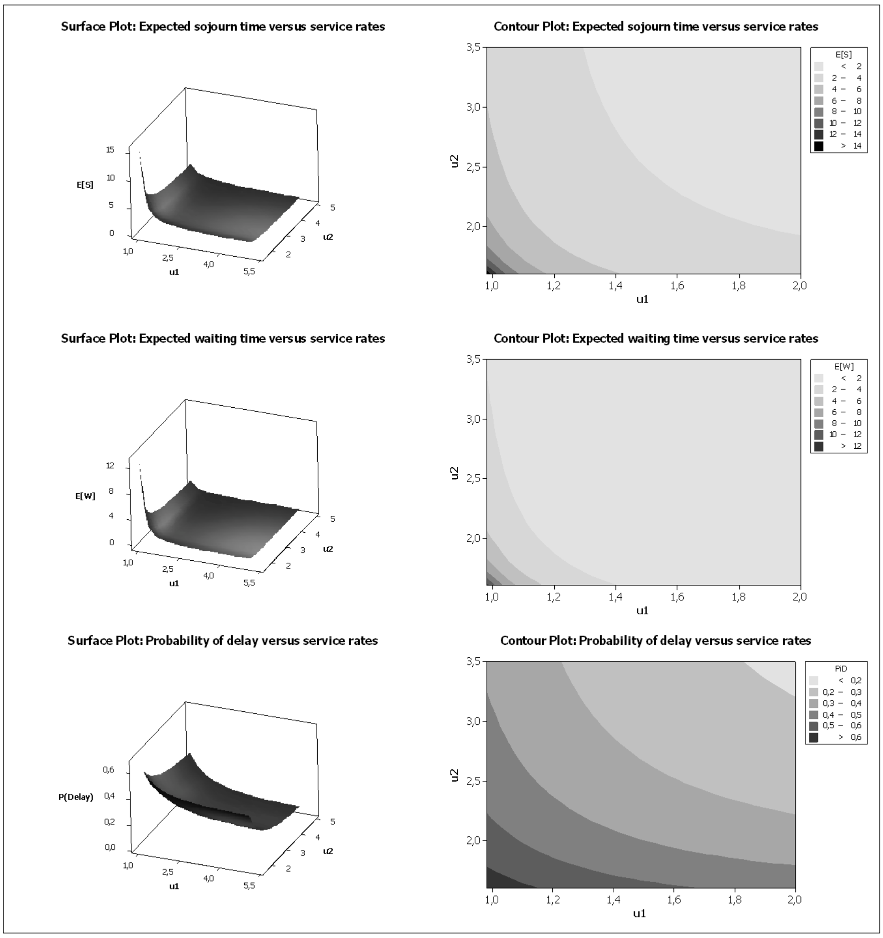
Figure 12.19: Simulation results in ranks



								Occupancy		Overtime (min)			Appointment lateness (min)					Break		Waiting time urgent patients (min)					
Adjustment/Sensitivity	Schedule	#out-patients	#in-patients	#special exams	#E-slots	Use of PR	Average #urgent patien	Mean	Var	Frac	Mean	Var	Mean	Var	cv	50%-Quantile	80%-Quantile	Morning	Lunch	Mean	Var	cv	50%-Quantile	80%Quantile	%>120
Original	S_P1 Random	20	8	2	6	Yes	2,704	0,80	4,5E-03	0,67	20,72	3,35E+02	7,29	1,05E+02	1,41	2,91	13,77	0,92	0,79	34,49	9,36E+02	0,89	27,80	58,44	1,59
	S_P2 Random A	20	6	3	5	Yes	2,687	0,79	4,1E-03	0,30	20,48	3,81E+02	7,09	1,06E+02	1,45	2,43	13,38	0,92	0,83	45,50	1,81E+03	0,94	33,39	78,49	7,26
	S_P2 noE 1	20	6	3	0	Yes	2,641	0,77	3,5E-03	0,13	13,09	1,44E+02	6,96	1,14E+02	1,54	1,57	13,20	0,91	0,81	16,00	1,93E+02	0,87	13,47	24,42	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,683	0,76	3,3E-03	0,15	13,01	1,71E+02	7,64	1,20E+02	1,43	2,53	14,58	0,94	0,84	16,29	2,06E+02	0,88	13,73	25,75	0,00
E-slot	S_P2 Random A (2)	20	6	3	5	Yes	2,69	0,79	4,7E-03	0,36	19,07	2,86E+02	7,18	1,07E+02	1,44	2,45	13,67	0,92	0,81	41,06	1,30E+03	0,88	32,60	70,14	3,80
#DR=2	S_P1 Random	20	6	3	5	Yes	2,71	0,81	4,5E-03	0,64	21,83	3,65E+02	8,09	1,16E+02	1,33	3,98	15,07	0,92	0,77	37,10	9,53E+02	0,83	31,46	63,19	1,44
	S_P2 Random A	20	6	3	5	Yes	2,67	0,80	4,2E-03	0,31	18,19	3,02E+02	7,74	1,06E+02	1,33	3,61	14,84	0,91	0,82	47,73	1,81E+03	0,89	37,61	80,64	7,57
	S_P2 noE 1	20	6	3	0	Yes	2,67	0,77	3,6E-03	0,14	13,45	1,56E+02	7,23	1,14E+02	1,48	2,24	13,68	0,90	0,83	15,80	1,85E+02	0,86	13,11	24,66	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,78	0,77	3,4E-03	0,15	13,26	1,41E+02	8,24	1,31E+02	1,39	3,41	15,60	0,93	0,85	16,50	2,11E+02	0,88	13,59	26,30	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,73	0,80	4,0E-03	0,38	19,11	3,01E+02	7,72	1,09E+02	1,35	3,60	14,52	0,90	0,82	43,83	1,51E+03	0,89	34,66	73,24	5,31
#DR=4	S_P1 Random	20	6	3	5	Yes	2,70	0,80	4,3E-03	0,64	21,20	3,36E+02	7,35	1,07E+02	1,41	2,86	13,76	0,92	0,79	34,24	8,91E+02	0,87	27,50	58,50	1,18
	S_P2 Random A	20	6	3	5	Yes	2,62	0,79	4,2E-03	0,31	16,53	2,61E+02	7,27	1,01E+02	1,38	2,92	13,92	0,92	0,81	44,39	1,71E+03	0,93	32,94	74,89	6,64
	S_P2 noE 1	20	6	3	0	Yes	2,72	0,77	3,5E-03	0,15	12,21	1,40E+02	6,87	1,11E+02	1,53	1,52	13,12	0,91	0,79	15,65	1,94E+02	0,89	13,28	24,56	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,72	0,77	3,4E-03	0,13	12,25	1,09E+02	7,84	1,24E+02	1,42	2,84	14,91	0,94	0,84	16,55	1,94E+02	0,84	14,08	26,55	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,71	0,80	4,3E-03	0,35	18,81	2,90E+02	7,28	1,05E+02	1,41	2,72	13,80	0,91	0,82	42,26	1,37E+03	0,87	33,28	72,69	4,46
Resid=0	S_P1 Random	20	6	3	5	Yes	2,75	0,80	4,4E-03	0,64	20,87	3,23E+02	7,09	1,02E+02	1,42	2,68	13,54	0,92	0,79	34,57	8,55E+02	0,85	29,20	57,04	1,35
	S_P2 Random A	20	6	3	5	Yes	2,75	0,80	4,5E-03	0,31	17,98	3,09E+02	7,25	1,08E+02	1,43	2,58	13,91	0,92	0,82	42,95	1,70E+03	0,96	30,00	74,67	6,38
	S_P2 noE 1	20	6	3	0	Yes	2,71	0,77	3,7E-03	0,15	15,46	1,73E+02	7,00	1,16E+02	1,54	1,54	13,20	0,93	0,82	16,09	1,91E+02	0,86	13,31	25,18	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,65	0,77	3,4E-03	0,12	13,79	1,64E+02	7,63	1,19E+02	1,43	2,54	14,45	0,94	0,85	16,18	2,04E+02	0,88	13,34	25,47	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,72	0,80	4,5E-03	0,35	18,63	2,89E+02	7,31	1,10E+02	1,44	2,57	13,93	0,92	0,81	41,10	1,44E+03	0,92	31,79	71,61	4,86
Resid=N(0,4)	S_P1 Random	20	6	3	5	Yes	2,80	0,81	4,4E-03	0,65	22,88	4,13E+02	8,35	1,30E+02	1,36	4,08	15,37	0,90	0,77	37,77	1,04E+03	0,85	31,46	63,45	2,14
	S_P2 Random A	20	6	3	5	Yes	2,72	0,80	4,3E-03	0,33	18,80	2,90E+02	8,17	1,20E+02	1,34	3,85	15,38	0,89	0,81	50,17	2,05E+03	0,90	36,88	84,07	9,44
	S_P2 noE 1	20	6	3	0	Yes	2,67	0,77	3,4E-03	0,12	12,88	1,27E+02	7,40	1,16E+02	1,46	2,32	14,03	0,90	0,79	16,54	1,94E+02	0,84	14,11	25,73	0,04
	S_P2 noE 2	20	6	3	0	Yes	2,65	0,77	3,1E-03	0,13	11,19	1,52E+02	8,36	1,34E+02	1,38	3,53	15,80	0,92	0,87	17,85	2,42E+02	0,87	14,82	27,29	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,90	0,80	4,2E-03	0,39	21,20	4,10E+02	7,98	1,25E+02	1,40	3,34	15,01	0,89	0,81	44,76	1,46E+03	0,85	36,41	74,94	5,08
Scan+N(-1,0.5	S_P1 Random	20	6	3	5	Yes	2,72	0,75	4,9E-03	0,59	19,20	2,86E+02	5,64	7,75E+01	1,56	0,71	10,99	0,95	0,85	28,57	7,86E+02	0,98	21,24	48,58	1,10
	S_P2 Random A	20	6	3	5	Yes	2,59	0,74	5,0E-03	0,25	15,41	2,16E+02	5,76	8,30E+01	1,58	0,54	11,18	0,96	0,80	34,97	1,34E+03	1,05	23,86	60,22	3,78
	S_P2 noE 1	20	6	3	0	Yes	2,76	0,71	3,9E-03	0,10	10,06	8,04E+01	5,37	7,70E+01	1,63	0,00	10,54	0,94	0,85	13,26	1,55E+02	0,94	10,93	21,45	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,70	0,71	3,9E-03	0,11	11,42	1,61E+02	6,06	9,29E+01	1,59	0,32	11,65	0,95	0,86	13,84	1,71E+02	0,94	11,17	22,92	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,52	0,74	4,9E-03	0,28	16,39	2,54E+02	5,60	7,97E+01	1,59	0,22	10,92	0,95	0,82	31,62	1,04E+03	1,02	23,25	53,14	1,98
Scan+N(1,0.5)	S_P1 Random	20	6	3	5	Yes	2,64	0,86	3,6E-03	0,72	25,91	4,83E+02	10,64	1,69E+02	1,22	6,86	18,82	0,83	0,60	45,08	1,17E+03	0,76	39,84	73,22	3,18
	S_P2 Random A	20	6	3	5	Yes	2,75	0,85	3,7E-03	0,41	23,73	4,47E+02	10,35	1,64E+02	1,24	6,51	18,56	0,86	0,75	60,16	2,38E+03	0,81	47,98	102,37	13,28
	S_P2 noE 1	20	6	3	0	Yes	2,73	0,82	3,3E-03	0,21	17,28	2,09E+02	9,53	1,69E+02	1,36	4,50	17,47	0,84	0,73	19,54	2,60E+02	0,83	16,17	29,70	0,04
	S_P2 noE 2	20	6	3	0	Yes	2,74	0,82	3,4E-03	0,22	14,90	1,92E+02	11,02	1,93E+02	1,26	6,54	19,92	0,85	0,79	20,48	2,78E+02	0,81	16,89	31,40	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,78	0,85	3,8E-03	0,47	24,24	4,60E+02	9,97	1,54E+02	1,25	6,14	18,04	0,85	0,76	54,62	1,71E+03	0,76	46,76	88,83	8,16
Punc=10	S_P1 Random	20	6	3	5	Yes	2,75	0,81	4,1E-03	0,51	21,40	3,59E+02	4,91	8,76E+01	1,90	0,00	8,66	0,91	0,82	34,81	9,15E+02	0,87	28,95	58,45	1,38
	S_P2 Random A	20	6	3	5	Yes	2,67	0,79	4,4E-03	0,25	19,46	3,83E+02	4,39	8,29E+01	2,07	0,00	7,26	0,92	0,68	43,73	1,80E+03	0,97	30,07	76,57	6,81
	S_P2 noE 1	20	6	3	0	Yes	2,62	0,77	3,2E-03	0,12	11,16	1,55E+02	3,63	7,39E+01	2,37	0,00	4,89	0,93	0,88	13,86	1,56E+02	0,90	11,27	20,86	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,65	0,77	3,3E-03	0,13	12,65	1,60E+02	7,19	1,20E+02	1,52	2,77	12,64	0,91	0,85	15,62	2,04E+02	0,91	12,58	24,66	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,75	0,80	4,4E-03	0,28	19,06	2,91E+02	4,24	7,81E+01	2,08	0,00	6,94	0,91	0,68	38,43	1,36E+03	0,96	28,38	66,30	4,29
Punc=0	S_P1 Random	20	6	3	5	Yes	2,73	0,81	3,8E-03	0,68	22,30	3,63E+02	9,26	1,18E+02	1,17	5,91	15,13	0,87	0,74	34,51	9,83E+02	0,91	26,78	60,43	1,54
	S_P2 Random A	20	6	3	5	Yes	2,70	0,80	4,0E-03	0,34	19,20	3,51E+02	9,54	1,07E+02	1,08	6,39	15,48	0,86	0,81	42,17	1,68E+03	0,97	30,80	72,42	6,38
	S_P2 noE 1	20	6	3	0	Yes	2,65	0,78	3,1E-03	0,14	12,92	1,41E+02	8,31	1,09E+02	1,26	4,57	13,13	0,85	0,79	15,18	1,98E+02	0,93	11,64	24,49	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,66	0,77	3,1E-03	0,16	11,65	1,25E+02	12,57	1,53E+02	0,99	10,40	20,49	0,80	0,81	18,62	2,82E+02	0,90	14,72	29,58	0,04
	S_P2 Random A (2)	20	6	3	5	Yes	2,80	0,80	4,1E-03	0,37	18,84	3,70E+02	9,23	1,11E+02	1,14	5,91	14,80	0,85	0,79	39,12	1,40E+03	0,96	28,85	68,92	4,71
E-rate +20%	S_P1 Random	20	6	3	5	Yes	3,49	0,82	4,6E-03	0,71	23,61	4,16E+02	8,28	1,32E+02	1,39	3,91	15,23	0,88	0,75	38,33	1,04E+03	0,84	31,55	64,73	1,98
	S_P2 Random A	20	6	3	5	Yes	3,37	0,81	4,5E-03	0,36	22,47	4,36E+02	7,97	1,23E+02	1,39	3,51	14,90	0,90	0,80	47,49	1,90E+03	0,92	36,22	80,77	7,96
	S_P2 noE 1	20	6	3	0	Yes	3,33	0,78	3,8E-03	0,18	14,73	1,69E+02	7,67	1,28E+02	1,47	2,23	14,48	0,90	0,79	16,54	2,00E+02	0,85	13,76	26,63	0,00
	S_P2 noE 2	20	6	3	0	Yes	3,37	0,78	3,7E-03	0,19	15,28	1,90E+02	8,71	1,49E+02	1,40	3,50	16,29	0,92	0,83	17,31	2,23E+02	0,86	14,55	27,02	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	3,31	0,81	4,2E-03	0,40	20,98	3,64E+02	7,75	1,17E+02	1,39	3,13	14,65	0,91	0,80	43,19	1,33E+03	0,84	35,15	72,55	4,20
E-rate -20%	S_P1 Random	20	6	3	5	Yes	2,28	0,80	4,3E-03	0,63	20,06	3,09E+02	7,10	9,67E+01	1,39	2,90	13,60	0,93	0,80	32,37	8,01E+02	0,87	25,87	55,16	0,96
	S_P2 Random A	20	6	3	5	Yes	2,21	0,79	4,1E-03	0,28	15,68	2,05E+02	6,91	9,46E+01	1,41	2,45	13,35	0,93	0,83	43,44	1,72E+03	0,95	32,76	72,35	6,57
	S_P2 noE 1	20	6	3	0	Yes	2,26	0,76	3,6E-03	0,11	11,28	1,28E+02	6,31	9,50E+01	1,54	1,00	12,21	0,92	0,85	14,80	1,68E+02	0,88	12,73	23,31	0,00
	S_P2 noE 2	20	6	3	0	Yes	2,31	0,76	3,1E-03	0,13	12,95	1,78E+02	7,32	1,16E+02	1,47	2,20	13,92	0,95	0,87	16,19	2,04E+02	0,88	13,71	25,59	0,00
	S_P2 Random A (2)	20	6	3	5	Yes	2,19	0,79	4,2E-03	0,31	17,32	2,51E+02	6,88	9,75E+01	1,43	2,21	13,24	0,92	0,83	38,92	1,34E+03	0,94	29,29	65,98	3,57
IV-rate +20%	S_P1 Random	20	6	3	5	Yes	2,66	0,83	3,7E-03	0,74	25,63	4,51E+02	8,49	1,33E+02	1,36	4,26	15,48	0,89	0,76	38,54	9,58E+02	0,80	32,22	63,25	1,88
	S_P2 Random A	20	6	3	5	Yes	2,64	0,82	3,9E-03	0,40	22,55	4,32E+02	8,19	1,29E+02	1,39	3,72	15,21	0,92	0,80	49,71	2,03E+03	0,91	37,27	80,44	9,70
	S_P2 Random A (2)	20	6	3	5	Yes	2,73	0,82	4,0E-03	0,45	21,58	3,60E+02	8,09	1,28E+02	1,40	3,41	15,03	0,91	0,80	45,11	1,45E+03	0,84	36,44	77,45	4,76
IV-rate -20%	S_P1 Random	20	6	3	5	Yes	2,66	0,77	4,8E-03	0,56	18,37	2,80E+02	6,40	8,81E+01	1,47	2,00	12,21	0,93	0,85	30,63	7,78E+02	0,91	23,92	53,82	0,79
	S_P2 Random A	20	6	3	5	Yes	2,74	0,78	4,6E-03	0,25	16,58	2,70E+02	6,75	9,16E+01	1,42	2,14	13,10	0,92	0,83	43,13	1,67E+03	0,95	31,84	72,72	5,92
	S_P2 Random A (2)	20	6	3	5	Yes	21,00	0,78	4,6E-03	0,27	15,20	2,04E+02	6,67	9,42E+01	1,45	1,89	13,04	0,94	0,80	37,19	1,23E+03	0,94	27,89	66,17	3,04

Figure 12.20: Simulations results for adjustments and sensitivity analysis

12.4 Queues with non-overtaking


Surface Plot: Expected sojourn time versus service rates

Contour Plot: Expected sojourn time versus service rates

Surface Plot: Expected waiting time versus service rates

Contour Plot: Expected waiting time versus service rates

Surface Plot: Probability of delay versus service rates

Contour Plot: Probability of delay versus service rates

Figure 12.21: Performance measures for the non-overtaking M/M/2→M/1 queue with different service rates

IBIS UvA
Instituut voor Bedrijfs- en Industriële Statistiek

Plantage Muidergracht 12, 1018 TV Amsterdam