Sequential scheduling
Research Paper Business Analytics

Hidde Kok 1767364

October 31, 2012

Abstract

The scheduling of outpatients is studied with general job duration and deterministic interarrival times, where we use a sequential scheduling method to derive these interarrival times. This is done by balancing between idle and waiting time, with the addition of tardiness as hard constraint. In this paper the mathematical equations for calculating an optimal sequential schedule are derived and the results are illustrated with some numerical examples.

1 Introduction

Probably the most important subject of the last elections in the Netherlands was healthcare. How do we keep it affortable and maintain a high level of care. Questions whether to privatize hospitals, cut back on management costs and many others issues were raised. In any case, hospitals have to cut back on their cost, while maintaining a high service level. One of the best ways to archieve such objectives is to work more efficiently.

The scheduling of expensive machines and manpower has become more effective in many departments of hospitals. Outpatient scheduling has been an area where not so much research has been done. The question remains whether the idle time of docters and the waiting time of patients can be reduced, to become more customer friendly with less costs. This kind of problem appears in more industries than in healthcare, for example financial advisors and other consultation branches have to schedule appointments as well. The model from this paper can be used in such kind of branches.

In general there are three measures to rate a schedule, i.e. waiting time (of the customer), idle time (of the server) and tardiness (of the server). This research starts with sequential scheduling as in the research of Kemper et al. [1], which schedules patients in a outpatient setting. The sequential method has as advantage compared to global methods that the speed is higher. In the research of Kemper et al. tardiness is not involved, which is important in practice. Therefore, tardiness is added to the model of Kemper in this research, together with practical approaches to calculate such a schedule.

This paper starts with describing the model for the sequential schedule in chapter 2. After that a formula to calculate an optimal sequential schedule is derived and used to find recursive relations for the waiting time in chapter 3. In this chapter also ways to add tardiness to the model are explored. Numerical results of a program that makes schedules sequential are discussed in chapter 4. In chapter 5 a discussion how the results can be used is given.

2 Model

2.1 The model

We want to schedule n customers and we define a risk for customer i ($i = 1, \ldots, n$) as

$$R_i(t_1, t_2, \ldots, t_i) = \mathbb{E}(g(I_i)) + \mathbb{E}(h(W_i)), \tag{1}$$

with I_i the idle time of the server between customers $i-1$ and i, W_i the waiting time of the ith customer and t_i the arrival time of customer i. This definition is ideal for the sequential scheduling, because when scheduling customer i at an optimal t_i we only need $t_1, \ldots, t_{i-1}$, since that gives all the information about the earlier customers and facts about later customers are not used.

To find these optimal t_i's, we use a loss function l that uses the fact that idle time and waiting time can not occur at the same time

$$l(x) = g(-x)1_{[x<0]} + h(x)1_{[x>0]}. \tag{2}$$

Now can we write Equation (1) as

$$R_i(t_1, t_2, \ldots, t_i) = \mathbb{E}l(W_i - I_i). \tag{3}$$

In the research only

$$\begin{aligned} g(x) &= \alpha x^2 \\ h(x) &= (1-\alpha)x^2 \end{aligned} \tag{4}$$

is taken into account, where $\alpha \in [0, 1]$. In the sequential scheme, we determine optimal interarival times $x_i = t_{i+1} - t_i$ for each customer and these x_i's give the optimal t_i's. In this scheme we assume job durations $B_1, \ldots B_n$ to be independent of each other and of time. We define $S_i = W_i + B_i$ as the sojourn time of customer i.

When tardiness is taken into account, we have an endtime T. Tardiness can be included in two ways, as a parameter and as hard constaint. When tardiness is included with a parameter, we have to consider a tardiness measure for the system. Because the sequential method optimizes the system job for job, it is unclear how to include it. Therefore a hard tardiness constraint is included. That means that only schedules meeting the tardiness constraint are accepted. The most obvious hard constraint is that a schedule has to finish in expectation at time T, but also constraints like with 95% chance the schedule has to be finished before T can be used.

With this endtime, we can also define the load of the system as $\frac{\sum_{i=1}^{n} \mathbb{E}(B_i)}{T}$, which should be less than one when a schedule ending before T is expected. When we have a system with $\frac{\sum_{i=1}^{n} \mathbb{E}(B_i)}{T} > 1$ we know that we can not have a schedule finish in expectation before T.

2.2 Choices in the model

The choice to make the schedule sequential is partly practical. When we do not consider all the jobs at the same time, but only one by one, computing the optimal interarrival times gets easier.

Furthermore, scheduling sequential also has a fairness property, because for each patient we do not consider all other patients which might give him a longer waiting time, because that is preferable in the total schedule. Each patient will be scheduled with balancing between the docters idle time and his own waiting time.

To prevent long waiting times we have taken g and h in Equation (1) quadratic, because this penalizes large values more severely than linear g and h. It makes sense to do this, because there is no way to prevent all idle and waiting time, so extreme cases should be avoided.

We only consider g and h as in Equations (4), because α gives the freedom to model all quadratic g and h. If we want to use functions $g^{'}(x) = ax^2$ and $h^{'}(x) = bx^2$ in (1), where $a+b \neq 1$, we could simple take $\alpha = \frac{a}{a+b}$. With this α we have the same $R_i(t_1, t_2, \dots, t_i)$ except for a constant, thus we do not have another minimum and we will find the same optimal t_i's.

3 Analysis

3.1 The sequential method

In sequential scheduling the first customer arrives at $t = 0$. Now all the information about the first customer is available to use, and we can calculate the optimal x_2 under Equation (1). Now all the information about the second customer is there, so we can calculate x_3 and so on. The equation to determine x_i given $x_1, \dots, x_{i-1}$ is derived in this subsection.

When we want to find the optimal interarrival time of the next customer, we have

$$\min_{x_i} \mathbb{E}(l_i(S_i - x_i)) = \min_{x_i} \mathbb{E}(\alpha(S_i - x_i)^2 1_{\{S_i - x_i < 0\}} + (1-\alpha)(S_i - x_i)^2 1_{\{S_i - x_i \geq 0\}}). \quad (5)$$

By straightforward calculations we see

$$\begin{aligned}\mathbb{E}(l_i(S_i - x_i)) &= \alpha \int_0^{x_i} f_{S_i}(s)(s-x_i)^2 \,\mathrm{d}s + (1-\alpha)\int_{x_i}^{\infty} f_{S_i}(s)(s-x_i)^2 \,\mathrm{d}s \\ &= \alpha[\int_0^{x_i} f_{S_i}(s)s^2 \,\mathrm{d}s - 2x_i \int_0^{x_i} f_{S_i}(s)s \,\mathrm{d}s + x_i^2 \int_0^{x_i} f_{S_i}(s) \,\mathrm{d}s] \\ &+ (1-\alpha)[\int_{x_i}^{\infty} f_{S_i}(s)s^2 \,\mathrm{d}s - 2x_i \int_{x_i}^{\infty} f_{S_i}(s)s \,\mathrm{d}s + x_i^2 \int_{x_i}^{\infty} f_{S_i}(s) \,\mathrm{d}s].\end{aligned} \quad (6)$$

Now consider the derivative with respect to x_i

$$\begin{aligned}
&\frac{d}{dx_i}\mathbb{E}(l_i(S_i - x_i)) \\
&= \alpha[f_{S_i}(x_i)x_i^2 - 2\int_0^{x_i} sf_{S_i}(s)\,\mathrm{d}s - 2x_i f_{S_i}(x_i)x_i + 2x_i\int_0^{x_i} f_{S_i}(s)\,\mathrm{d}s + x_i^2 f_{S_i}(x_i)] \\
&+ (1-\alpha)[-f_{S_i}(x_i)x_i^2 - 2\int_{x_i}^{\infty} sf_{S_i}(s)\,\mathrm{d}s + 2x_i f_{S_i}(x_i)x_i + 2x_i\int_{x_i}^{\infty} f_{S_i}(s)\,\mathrm{d}s - x_i^2 f_{S_i}(x_i)] \\
&= 2\alpha[x_i(1 - \int_{x_i}^{\infty} f_{S_i}(s)\,\mathrm{d}s) - (\int_0^{\infty} sf_{S_i}(s)\,\mathrm{d}s - \int_{x_i}^{\infty} sf_{S_i}(s)\,\mathrm{d}s)] \\
&- 2(1-\alpha)[\int_{x_i}^{\infty} sf_{S_i}(s)\,\mathrm{d}s - x_i\int_{x_i}^{\infty} f_{S_i}(s)\,\mathrm{d}s].
\end{aligned} \tag{7}$$

We will now only consider the last line of (7) without the constant $-2(1-\alpha)$

$$\begin{aligned}
\int_{x_i}^{\infty} sf_{S_i}(s)\,\mathrm{d}s - x_i\int_{x_i}^{\infty} f_{S_i}(s)\,\mathrm{d}s &= \int_{x_i}^{\infty} f_{S_i}(s)(s - x_i)\,\mathrm{d}s \\
&= \int_{s=x_i}^{\infty}\int_{u=x_i}^{s} f_{S_i}(s)\,\mathrm{d}u\,\mathrm{d}s \\
&= \int_{u=x_i}^{\infty}\int_{s=u}^{\infty} f_{S_i}(s)\,\mathrm{d}s\,\mathrm{d}u \\
&= \int_{x_i}^{\infty} \mathbb{P}(S_j > u)\,\mathrm{d}u.
\end{aligned} \tag{8}$$

If we now use this, Equation (7) is equal to

$$\begin{aligned}
&2\alpha[x_i - \int_0^{\infty} sf_{S_i}(s)\,\mathrm{d}s + \int_{x_i}^{\infty} \mathbb{P}(S_j > s)\,\mathrm{d}s] + 2(1-\alpha)[-\int_{x_i}^{\infty} \mathbb{P}(S_j > s)\,\mathrm{d}s] \\
&= 2\alpha(x_i - \mathbb{E}(S_i)) + 2(2\alpha - 1)\int_{x_i}^{\infty} \mathbb{P}(S_j > s)\,\mathrm{d}s,
\end{aligned} \tag{9}$$

and we can conclude that x_i has a minimum when

$$\alpha(x_i - \mathbb{E}(S_i)) + (2\alpha - 1)\int_{x_i}^{\infty} \mathbb{P}(S_i > s)\,\mathrm{d}s = 0. \tag{10}$$

3.2 Exact calculations

With help of Equation (10) we theoretically can form a schedule when α and the distributions of the B_i's are given. Here we show that calculating a schedule analytically is in practice hard, even when α and the B_i's are choosen simple. First we assume that the interarival times are given and calculate the cumulative distribution functions (cdf's) of the waiting time. Then we optimize the interarrival times with probability density functions (pdf's) of the sojourn time and $\alpha = \frac{1}{2}$, such that Equation (10) becomes simpler. In both these simple cases we show that computations already get hard.

In this first case, we take all B_i's exponential with rate μ and x_i known. We prove with induction that the waiting time W_i has the form

$$\mathbb{P}(W_i > \omega) = \sum_{j=0}^{i-2} c_j(i)\omega^j e^{-\mu\omega}, \tag{11}$$

for $i \geq 2$, by straightforward calculation and give a recursive manner to find $c_j(i)$. We start with $(i = 1)$

$$\mathbb{P}(W_1 = 0) = 1, \tag{12}$$

because the first customer has no one to wait for. By using the well-known Lindley recursion

$$\begin{aligned}\mathbb{P}(W_2 > w_2) &= \mathbb{P}((W_1 + B_1 - x_1)^+ > w_2)\\ &= \mathbb{P}(B_1 > w_2 + x_1) = e^{-\mu(w_2+x_1)},\end{aligned}$$

for $w_2 > 0$. We need $w_2 > 0$, due to a propability mass in 0 due the $^+$ sign in $(W_1 + B_1 - x_1)^+$. This proves the base case of Equation (11) and we now assume this equation to be true until a certain i and prove the statement for $i+1$ to finish the induction proof.

Using Lindley again

$$\begin{aligned}\mathbb{P}(W_{i+1} > \omega) &= \mathbb{P}((W_i + B_i - x_i)^+ > \omega)\\ &= \mathbb{P}(W_i + B_i > \omega + x_i)\\ &= \int_0^{\omega+x_i} f_{B_i}(y)\mathbb{P}(W_i > \omega + x_i - y)\,\mathrm{d}y + \mathbb{P}(B_i > \omega + x_i)\\ &= \int_0^{\omega+x_i} \mu e^{-\mu y}\sum_{j=0}^{i-2} c_j(i)(\omega + x_i - y)^j e^{-\mu(\omega+x_i-y)}\,\mathrm{d}y + e^{-\mu(\omega+x_i)}\\ &= \sum_{j=0}^{i-2} c_j(i)\int_0^{\omega+x_i} \mu(\omega + x_i - y)^j e^{-\mu(\omega+x_i)}\,\mathrm{d}y + e^{-\mu(\omega+x_i)}\\ &= \sum_{j=0}^{i-2} c_j(i)\mu e^{-\mu(\omega+x_i)}\frac{-(\omega + x_i - y)^{j+1}}{j+1}|_{y=0}^{\omega+x_i} + e^{-\mu(\omega+x_i)}\\ &= e^{-\mu(\omega+x_i)}[1 + \mu\sum_{j=0}^{i-2} c_j(i)\frac{(\omega + x_i)^{j+1}}{j+1}].\end{aligned} \tag{13}$$

Now the binomial theorem is used for $(\omega + x_i)^{j+1}$ in (13) to find

$$\begin{aligned}\mathbb{P}(W_{i+1} > \omega) &= e^{-\mu(\omega+x_i)}[1 + \mu\sum_{j=0}^{i-2} c_j(i)\frac{1}{j+1}\sum_{k=0}^{j+1}\binom{j+1}{k}\omega^k x_i^{j+1-k}]\\ &= e^{-\mu\omega}e^{-\mu x_i}[1 + \mu\sum_{j=0}^{i-2} c_j(i)\frac{1}{j+1}\binom{j+1}{0}\omega^0 x_i^{j+1}\\ &+ \mu\sum_{k=1}^{i-1}\sum_{j=k-1}^{i-2} c_j(i)\frac{1}{j+1}\binom{j+1}{k}\omega^k x_i^{j+1-k}].\end{aligned} \tag{14}$$

Which is of the form of Equation (11), so we have finished the induction proof and we found the recursive relations

$$c_0(i+1) = e^{-\mu x_i}[1 + \mu\sum_{j=0}^{i-2} c_j(i)\frac{1}{j+1}x_i^{j+1}] \tag{15}$$

and

$$c_k(i+1) = \mu e^{-\mu x_i}\sum_{j=k-1}^{i-2} c_j(i)\frac{1}{j+1}\binom{j+1}{k}x_i^{j+1-k}. \tag{16}$$

This makes clear that in this simple case, n does not need to get very big, before computations become too cumbersome to do analytically. When we do not have the same rate for each job duration, we see that the recursive relations already do not hold anymore.

We now try to calculate an optimal schedule analytically by using the pdf's instead of the cdf's and combine this with choosing some parameters for simplification. Clearly the analysis is very similar to the analysis with the cdf's. The biggest different is that the x_i's are now derived.

We now calculate the x_i's and we take $B_i = e^{-x}$, such that we have less parameters to keep track of. Furthermore we take $\alpha = \frac{1}{2}$, such that Equation (10) gives the optimal interarrival time $x_i = \mathbb{E}S_i$. We can simply start with $f_{S_1}(s) = f_{B_1}(s)$, because the first customer has no waiting time and conclude that $x_1 = 1$.

Now it is possible to calculate $f_{S_2}(s)$:

$$\begin{aligned} f_{S_2}(s) &= \mathbb{P}(S_1 < x_1) f_{B_2} + \int_{x_1}^{s+x_1} f_{S_1}(t) f_{B_2}(s-t+x_1)\,\mathrm{d}t \\ &= \int_0^{x_1} e^{-x}\,\mathrm{d}x e^{-s} + \int_{x_1}^{s+x_1} e^{-t} e^{-(s-t+x_1)}\,\mathrm{d}t \\ &= (1-e^{-1})e^{-s} + \int_1^{s+1} e^{-s-1}\,\mathrm{d}t \\ &= (1-e^{-1})e^{-s} + [te^{-(s+1)}]_{t=1}^{s+1} \\ &= (1-e^{-1})e^{-s} + se^{-(s+1)}. \end{aligned} \tag{17}$$

We determine $x_2 = \mathbb{E}(S_2)$ by using the fact that the first moment of an exponential distribution is $\frac{1}{\mu}$ and the second moment is $\frac{2}{\mu^2}$. With this we see that $\mathbb{E}S_2 = ((1-e^{-1}) + 2e^{-1} = 1 + e^{-1}$.

Here we see that in a simple case it is again possible to find some expression for S_n, when n doesn't get too large. Again we see that when we take different rates for the exponential distribution, or take some other distribution, computations like this will get extremely hard.

So we have seen that already in simple cases, it gets very hard, when n is large, to calculate a schedule with Equation (10). When we assume that all job duration are exponentially distributed with the same rate, you could find some acceptable recursive relations, but even here you need the computation strength of a computer to practically calculate an optimal schedule.

3.3 Tardiness

In the present model there is no way to control the tardiness, which is an important measure for the quality of a schedule. The time the server finishes is, in expectation, equal to the sum of expected service times plus the sum of all the expected idle times. So we can only change the tardiness by changing the idle times. When Equation (1) is taken with g and h as in (4) the idle times can only be influenced by changing α. So in the current model tardiness can be included by searching for the best α such that idle time, waiting time and tardiness are balanced.

To define what a best α is, we consider tardiness as a hard constraint, because people do not like overtime and overtime is very expensive. So when we plan enough customers such that the total expected job durations get close to T, we could choose α as small as possible such that the expected endtime still is less than T. This way we have a schedule that includes tardiness, still does not have to much idle time, because big gaps in the schedule will make that we do not finish before T, and with the lowest waiting time the tardiness constraint allows us.

The α were the expected endtime of the schedule is close to T is considered to be the best α. We already concluded that a higher α always leads to less idle time and tardiness and more waiting time. So α as small as possible such that the expected endtime still is less than T makes sense as the best α. But it is not very important to keep the expected endtime under T. Because the model has a lot of stochasticity in it, we can not make sure that the model always ends before T, but only in expectation. Whether this expectation is $T + \epsilon$ or $T - \epsilon$ with ϵ small, will not matter much and thus we can take the α with an expected endtime of the schedule close to T.

To find such an α, we use a heurisic improving scheme. We start with an initial α and improve until we have an expected endtime close to T. Because the expected endtime of a schedule is stricly decreasing in α, we take α_{new} as $\alpha_{new} = \alpha_{old} * \frac{\text{expected endtime of the schedule with } \alpha_{old}}{T}$. This will give a new α closer to our optimal α. We can continue until the expected endtime of the α schedule is close enough to T according to a user defined tolerance interval.

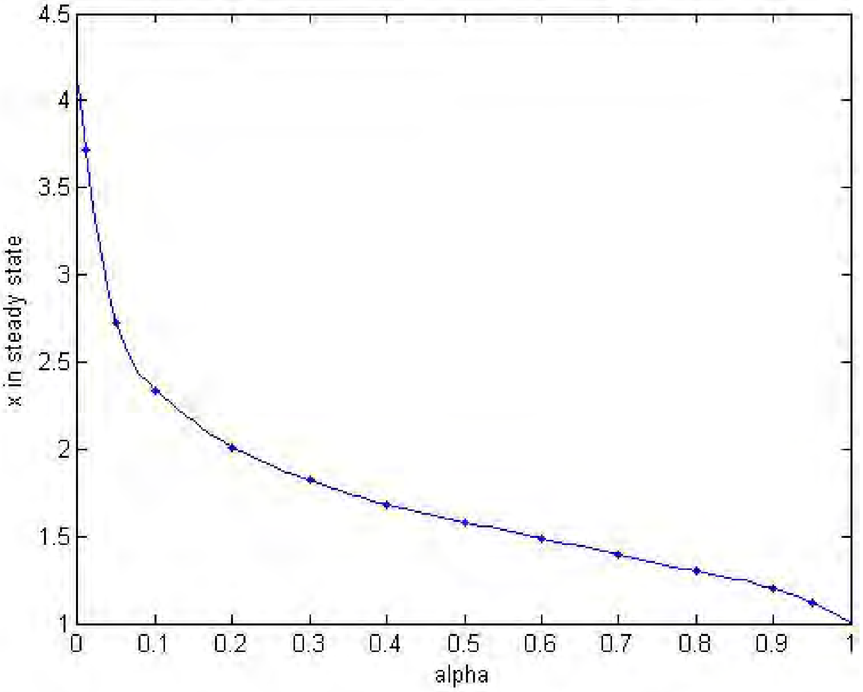

Figure 1: The steady state interarrival time for different α's of a schedule with all job durations exponentially distributed with rate one.

Very important for such an improving scheme is the choice of the starting α. By having a starting α close to the optimum, we have to do only a few iterations and decrease the number of calculations in this manner. For this an estimation of extra time (x^*) per job is made, where $x^* = \frac{T}{\sum_{i=1}^{n} \mathbb{E}(B_i)}$. This extra time is used to guess the steady state interarrival time $\bar{x_i} = x^* \mathbb{E} B_i$ and such a steady state interarrival time has a unique α related to it, which would be a reasonable starting α. So it would be nice when a closed formula for α could be find, with as input the steady state interarrival time, but that is quite hard. During the research we tried to solve this by using estimations for the waiting time from [3], but an appropriate closed formula has not been found. Therefore α now is guessed by using figures as Figure 1, by looking up the value of α that belongs to a certain $\bar{x_i}$. For example, when we would have an estimated steady state interarrival time of 1.5 we could see that a corresponding α would lie around 0.6. Figure 1 is only for exponential distributions, but also for other distributions it works decent when we simply take the α value connected with x^*. So improvement could be archieved by making plots for other distributions as well, but Figure 1 can also be used for non exponential distributions.

This α would be a good starting point, when we would reach steady state fast. In Figure 2 it is shown how fast steady state is reached. In this figure we see the x_i's of a schedule with exponential distributed job durations with rate 1. The green line is $\alpha = 0.1$, the blue line is $\alpha = 0.5$, the red line is $\alpha = 0.9$ and the yellow line is $\alpha = 0.95$. Steady state is reached when the difference between some x_i and x_{i+1} is small. We can see that when we have a full system (which most practical cases have) with $\alpha \geq 0.9$, we do not reach steady state very fast, so our method for guessing the starting α with steady state interarrival times does not work optimal for such α's. This is solved by using a correction, such that α is decreased towards the optimal α. This correction is guessed at $\frac{n-1}{n}$, but should be studied further, where the load probably should be taken into account to have a better estimation for this correction value.

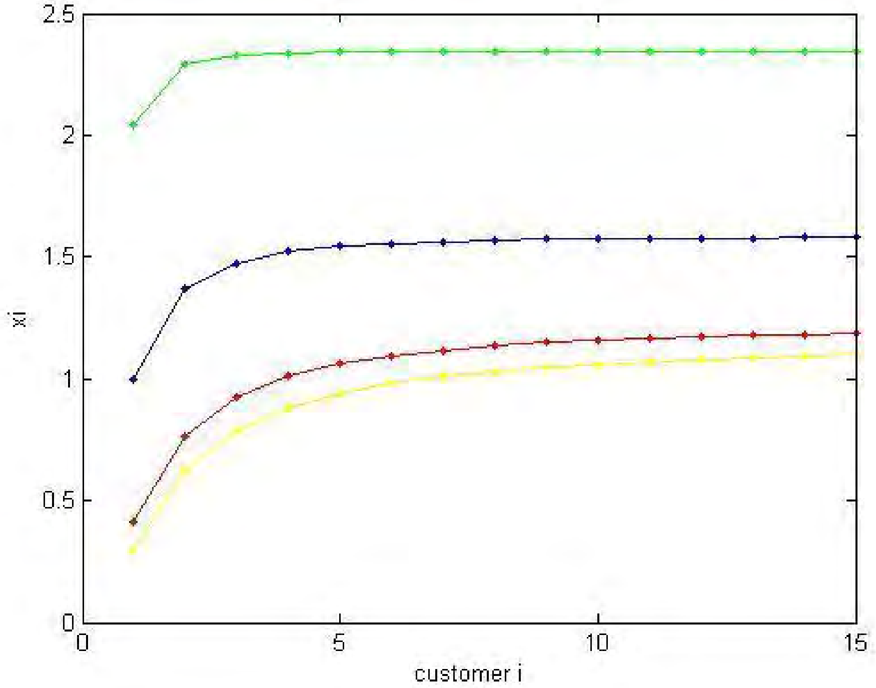

Figure 2: The interarrival times of the customers of schedules with different α.

4 Numeric examples

4.1 Choices in programming

In Chapter 3 we derived Equation (10) to make an optimal schedule, but this chapter also shows that it is hard to determine such a schedule analytically. Therefore code in matlab to calculate an optimal schedule is programmed. Matlab is chosen because lots of calculations (i.e. integration) made for creating a sequential schedule are preprogrammed in matlab.

The most important choice in the program is between symbolic and numeric computation. The advantages of symbolic computation is that it does not need any asumptions and it is similar with the analytical derivations. The downside is that when n gets bigger the computations get hard and a matlab program is slow and in the end fails in the calculation, because to calculate the results in closed form is not possible anymore.

Numeric calculation uses a time grid with intervals that contain the chance that a customer finishes (when using sojourn time) in that time interval. How big these time intervals are, is an important issue of numeric calculation. By taking them too big, you will lose too much information. By taking them too small, you will need a lot of computation power. But not only the size of the intervals is important, also when to stop calculating how large the propability is that a customer finishes in that interval. With an exponential distribution there is a propability that an appointment with expected duration of 10 minutes, takes weeks. This propability is very small, but the propability that it takes 2 hours should be considered. So when you stop considering small propabilities too fast, you lose information.

The choice during this research has been to use the symbolic method, because we prefer calculations that are the same as the analytical calculations. We prefer not to make assumptions that in a program for practical use might are easy to make, to increase the speed of the program.

Moreover the program makes use of the pdf's of the sojourn times, instead of cdf's and/or waiting times. These choices will not make a big difference, because pdf's and cdf's are connected though integration. Waiting times are more basic to calculate compared to sojourn times, but need a convolution with the job duration when you use Equation (10).

As mentioned before, when n is large, the program gets slower and might not be able to calculate a schedule. How big this n can be, is heavily dependent on which distributions are used for the job durations. Some distributions like lognormal, even can not be taken into account at all by the program.

Estimating the starting α in the improving scheme could quite easily be improved. At the moment schedules that only have exponential job durations has a good guess by the plot of Figure 1. When you want to use the same method, you have to make such a plot for each distribution, most having multiple parameters. At the moment those α values are all guessed by using the exponential line, which makes the estimation of the starting α worse.

4.2 Results from symbolic computations

When we reconsider Figure 2 we have more information than the rate of convergence, also the schedules made with a certain α are visible in there. All the x_i's are there and thus we know how the schedule looks like. In Figure 3 you can see the same schedules, but now shown with the t_i's instead of the x_i's. The the horizontal axis denotes the α value for that particular schedule. The schedule with $\alpha = 0.1$ has only 12 jobs, the other 16, for scale reasons. The black dots stand for the t_i's and the colors give the duration of the full schedule until the last customer arrives. The job durations are exponential distributed with rate one and the time scale is based on this rate. We can clearly see here that lower α leads to larger x's and thus a later endtime. Also the rate of converge can be seen in the figure, by looking at the distance between the t_i dots, but not as clear as in Figure 2.

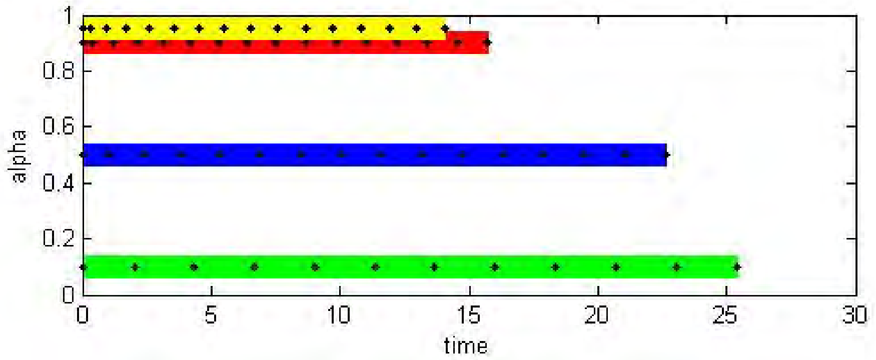

Figure 3: The schedules of exponential distributed job duration with rate 1.

The rate of converge with different α is very clear in Figure 4. Here we see the convergence of the pdf of the sojourn time. The black line of the figures is the sojourn time of the second customer, and the color progresses over the customers from more grey towards blue. We again can conclude that when we have higher α, it takes more iterations to reach steady state. The figure also shows that the tails of the sojourn times distribution gets heavier, thus we see the increase in interarrival times over time explained in this figure as well.

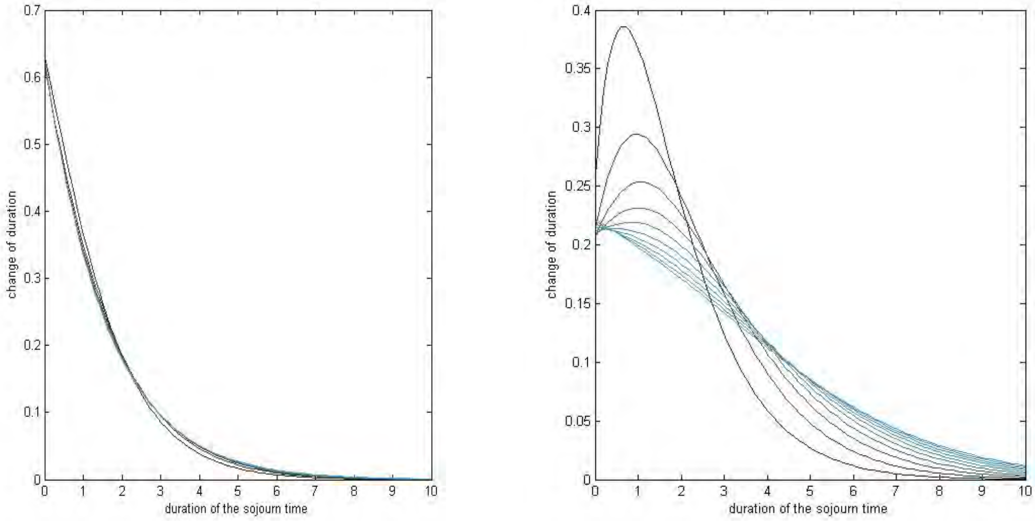

Figure 4: The pdf's of the sojourn times with (left) $\alpha = 0.5$ and (right) $\alpha = 0.95$.

In [1] is stated that the jobs should be ordered in increasing variance. We check this by making a few schedules with the same collection of jobs (exponential with different rates) and α, but in different job order. The result can be found in Figure 5 and in the next Table specifications about this figure are given. This figure indeed shows that that the order with increasing variance is the best and the order with decreasing variance the worst.

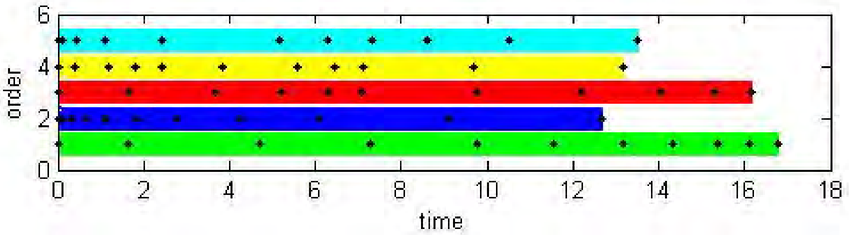

Figure 5: The schedules of exponential distributed jobs in different job orders and with $\alpha = 0.9$.

hight	color	order in rates
1	green	0.25, 0.25, 0.5, 0.5, 1, 1, 2, 2, 4, 4
2	blue	4, 4, 2, 2, 1, 1, 0.5, 0.5, 0.25, 0.25
3	red	0.25, 0.5, 1, 2, 4, 0.25, 0.5, 1, 2, 4
4	yellow	1, 1, 2, 2, 0.5, 0.5, 4, 4, 0.25, 0.25
5	light blue	4, 2, 1, 0.5, 0.25, 4, 2 , 1, 0.5, 0.25

When we make a schedule that includes tardiness, by having an expected endtime around T, not much changes in the model. We use exactly the same model as before, only we are estimating α. The question that we do like to answer is how good this model is compared to non-sequential models, where we calculate how much tardiness is expected in a system instead of considering only whether the expected endtime is before T.

In the paper of Kaandorp and Koole [2] such a non-sequential schedule is calculated by means of local search. The downside of this method is that it uses a time grid instead of continues time as in the case of the sequential scheduling. We tried to use their method with a small gridsizes (10 exponential distributed customers with rate $\frac{1}{20}$ and interval sizes of 1), but after about 18 hour of calculating, the program gave

```
Searching full neighborhood...
Checked 116820 of 12388450832 schedules,
```

which means that with such a small gridsize calculation takes too long, even if we would use a very good computer. With a bigger gridsize a comparison with the sequential schedule became incorrect, because the local search method uses the fact that customers arrive at the starting of the timegrid. Putting the appointments after the sequential scheduling on the starting point of a timegrid, has big impact at the schedule when the sequential scheduled time and the timegrid time are far apart.

5 Discussion

Compared to earlier research in sequential scheduling tardiness is included. This has been based on some guesses and should be studied further to be able to improve these guesses or find optimal values. You might be able to use steady state estimation from [3] to be able to get a better basic α used in the improving scheme of a schedule with tardiness. Also more options in the tardiness constraint are possible. At the moment only an expected endtime around T is considered as a hard constraint. In practice a schedule with, in terms of tail probabilities, a maximum probability of exceeding a threshold might be asked. Such hard constraints could be added to the present program.

In this research recursive relations for simple cases where found. When you have exponential distributed job duration with the same rate, we have found Equations (15) and (16) as recursive relation for the waiting time. With these equations a very quick program for calculating waiting times could be made. When the assumption of having all job durations distributed exponential with the same rate μ is a realistic one in practice, a very quick program for sequential scheduling is possible.

Limitations in the program used are in distributions possible in the program and the size of n, both important in practice. Both can be prevented with a numeric approach instead of a symbolic approach. The downside of the numeric approach is that you have to make time intervals, where assumptions about interval size and when a propability still has a significant influence have to be made. In practice, the first will not be a big issue, because you will not make appointments to a second precise, but at rounded times. The second might be more troublesome, because a logical value for this quantity is not available in practice.

No shows are not explicitly considered in the current model and program. But when we lower the whole distribution of a job duration by multiply it by $\rho < 1$, we still have correct calculation. See [1] for the mathematical details. This way no shows can be considered in the present program.

We can conclude that the sequential scheduling works, it gives a schedule that is sequential optimal and can be calculate fast. For a practical program, some problems should still be solved, but no big issues are expected. How good an optimal sequential scheduled solution is compared to a true optimal schedule, is hard to decide. The methods used to calculate such an optimum does not use continues time and takes very long when a fine time grid is taken, such that it gets close to continues time. We can therefore not conclude whether the sequential model performs worse, but can conclude that the speed of the sequential method is significantly better.

References

[1] Kemper, B., C. Klaassen, and M. Mandjes. Utility-based appointment scheduling. *Submitted 2011.*

[2] G. Kaandorp and G. Koole. Optimal outpatient appointment scheduling. *Health Care Manage Sci* 2007 10:217 - 229.

[3] D.P. Heyman. A diffusion model approximation for the GI/G/1 Queue in heavy trafic. *The bell system technical journal*, Volume 54, Number 9, 1975.