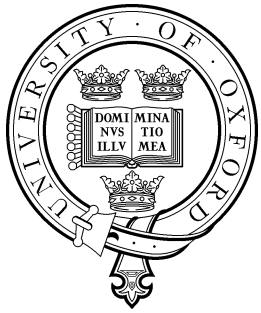
Robust Staff Level Optimisation in Call Centres



Sam Clarke

Jesus College

University of Oxford

A thesis submitted for the degree of

M.Sc. Mathematical Modelling and Scientific Computing

Trinity 2007

Abstract

The aims of this thesis were to:

- understand the main methods employed in call centres to choose the number of agents answering telephones;
- examine relevant literature related to these methods;
- develop a framework with the aim of highlighting any potential flaws in these methods;
- attempt to use this framework to improve these methods, if possible.

The motivation behind this research was of a very practical nature. It is known that the methods employed in many call centres to choose the number of agents to answer the telephones are not up to scratch.

The main results of this thesis are that:

- standard methods are suspect;
- a framework has been developed to examine a relevant queueing model which allows for scenario analysis and stress testing;
- a "cutoff phenomenon" exists in this system;
- the robustness of the system strongly depends on the second eigenvalue of a certain matrix;
- the use of simulation methods alongside analytical models can be valuable;
- an improved method for choosing agent numbers has been developed which we hope can be applied in practical situations.

Acknowledgements

There are many people who I would like to thank for their assistance and support in writing this thesis and during the course as a whole.

I begin by thanking my supervisor, Raphael Hauser, for his ideas and guidance, particularly when I simply did not know which avenues to explore. I would like to thank Bent Grover for suggesting such an interesting project and for his knowledge and advice on real world call centre operations.

A special mention must go to the EPSRC. I would not have been able to accept a place on the course without their financial support, for which I am very grateful.

I would also like to thank Peter Grindrod and everyone who worked at Numbercraft for igniting my interest in mathematics of a practical nature during my industrial placement year. Finally, I would like to thank my family and friends for their support throughout my whole academic career.

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview of Call Centre Operations

A call centre is a centralised office containing resources, such as personnel, computer and telecommunications equipment that delivers services by telephone. Call centres have become an integral part of the operations of most major businesses in recent years. Businesses see call centres not only as a method of delivering services to customers efficiently, but also as a way of maintaining a direct personal link with their customers. There is a current trend to turn call centres into *contact centres*. Contact centres aim to, not only deliver services by telephone, but also by letter, fax and e-mail. A call centre can handle either inbound calls, outbound calls, or both. Typical functions performed by inbound call centres include customer service, order taking and acting as help desks or emergency response desks, whilst functions performed by outbound call centres include sales, advertisement campaigns and customer surveys.

Recent technological developments in information technology have meant that it has become possible to deliver call centre services from anywhere in the world. This has caused many companies to move their call centres to countries where labour costs are much reduced. However, research suggests that this offshoring has not prevented continuing strong growth in UK call centres [11]. In the US in 1999, it was estimated that more than 1.4% of all private-sector employees worked in call centres with this number growing at more than 8% per year [9].

Call centre personnel are referred to as *agents* or *customer service representatives (CSRs)*. The organisation of personnel can vary greatly. In centres where no great skill is required for an agent to perform the required tasks, often every agent is trained to perform all tasks and calls are routed according to a *first come first served* scheme. However, in centres which perform several more complicated tasks, agents are often trained to perform fewer tasks. Callers are then either routed through several layers of

agents, ending with the most specialised if previous agents have been unable to solve their query, or use their telephone keypad to answer a series of automated questions that enable a computer system to automatically route their call to the correct queue.

The information technology equipment available to call centres has improved dramatically in recent years. Services that used to require the customer to use their keypad to select options now often use voice automation, with some banks seeing up to 80% of calls dealt with solely by computer [9]. Some centres even automatically call back a customer who has abandoned their call prior to it being answered, when there is a free agent. Access to customers' information is often made easier by technology known as *computer telephone integration*, which searches for a customers record and displays it on the relevant agent's screen, before the call has even been answered.

1.2 Overview of the Planning Process

One of the most important tasks in call centres is the construction of an agent schedule. Too many agents leads to unnecessary costs, whilst too few leads to substandard service. In order to draw up an agent schedule, forecasts must first be developed for the rate at which calls arrive into the call centre and the *average handling time*. The *average handling time* or *service time* is not defined as the time that an agent spends on the telephone, but as the total time that an agent takes to deal with a call. This is the talk time as well as time spent before or after a call performing tasks related to that call. Based on these forecasts, the number of agents required to achieve the *service level* can be determined. The standard technique used to do this is described in Chapter 2. These numbers can then be used to determine the agent schedule. During the day, forecasts can be updated based on the actual number of calls arriving and agents redeployed as necessary. We go into the details of this process below.

1.2.1 Forecasting

In order to estimate the number of agents required, a call centre needs to produce forecasts for the number of calls arriving into a call centre and an estimate for the average handling time. On short time scales, the number of calls arriving into a call centre exhibits significant stochastic variability. However, on longer time scales, we also see more predictable daily, weekly, monthly and seasonal variability. Figure 1.1 shows a piecewise linear approximation of how the daily arrival rate varies in a typical call centre. As a call arrives and moves through a call centre, information is recorded about the call, for example, the time of arrival, the time spent waiting in the queue

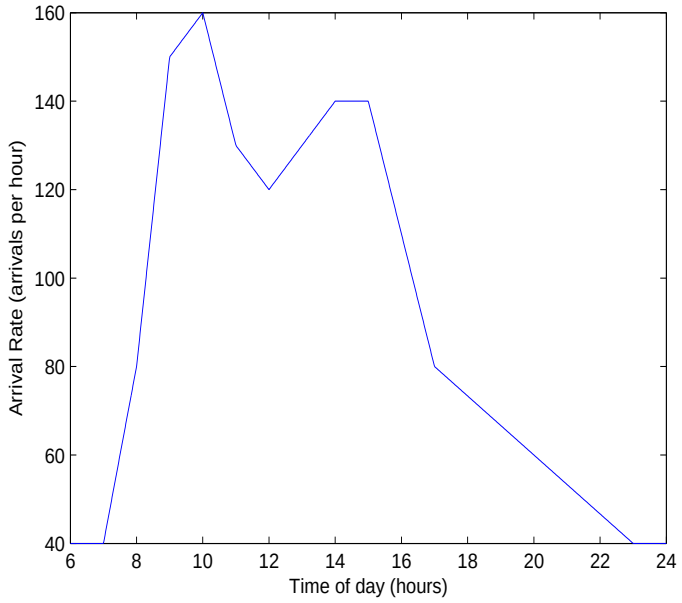


Figure 1.1: Piecewise linear approximation of a typical arrival rate. Adapted from [9], Figure 5

and the service time. Since call centres generate so much data, it is (or certainly used to be) prohibitively expensive to store it all, so summary statistics over short time intervals of half hours or hours are often produced. These summary statistics are important not only for measuring how well the system performed, but for predicting likely arrival rates and service times at comparable times in the future.

Call centres produce forecasts for the expected arrival rate based on both historical data and expected predictable variations. The focus of this thesis is not on how these forecasts are produced, so we point the reader towards [9], Section 3.3, for more information. From this point on, we will assume that a (time-varying) forecast for the expected arrival rate and a point estimate for the average handling time have been produced.

1.2.2 Service Level

An important problem is that of defining the *grade of service* that a call centre achieves. Criteria can be qualitative as well as quantitative. Qualitative service level criteria might involve measuring whether or not the service was *effective*, for example, whether or not the customer's problem was solved, or measuring the quality of the *interactions* between an agent and the customer, such as the politeness of the agent.

Qualitative service levels are mainly aimed at measuring customer satisfaction with the service provided [9].

In this thesis, we will restrict our attention to quantitative service levels, since these are normally associated with the issue that we will be most interested in: how to calculate the minimum number of agents that achieve a desired service level. Quantitative service levels normally measure the *accessibility* of agents. The quantitative service level that is seen most often is related to waiting times and is stated as

ensure that more than α% of callers wait less than T seconds to be served,

where typical values are $\alpha = 80$ and $T = 20$ ([14], Chapter 15). In this case, we can define the *achieved grade of service* as the *percentage of customers who actually wait less than T seconds*. This can be interpreted in several ways. We may want to ensure that this is achieved instantaneously, so that the service level holds at every time throughout the day, or we may want to ensure that it is achieved averaged over the day, or achieved on average for every one of several small time intervals during the day. Another example of a quantitative service level is based on abandonments and stated as

ensure that less than β% of callers abandon their calls prior to them being answered,

where typical values are $\beta = 3\%$ or $\beta = 5\%$ ([14], Chapter 15). Note that satisfying the service level based on abandonments is highly correlated to satisfying the service level based on waiting times: as waiting times increase, more customers will abandon.

A combination of both quantitative and qualitative service levels is normally desirable. A customer who receives excellent service once their call has been answered, but has had to wait a long time to receive it, will probably be dissatisfied. Similarly, a customer whose call is answered immediately, but then receives below average service will probably also be dissatisfied.

1.2.3 Capacity Management

Suppose that we are trying to satisfy a quantitative service level of the form given in the previous section. Increasing the number of agents will certainly improve the achieved grade of service. However, since typically $60-70\%$ of all operating costs of a call centre are personnel costs ([14], Chapter 15), minimising the personnel required to achieve the desired service level is one of the most important problems in call centres and is the problem that we will concentrate on in this thesis. Furthermore, the cost of employing agents may be different at different times of the day, so we will be interested not only in minimising the total number of agent hours, but in

minimising the *total agent cost*. This problem is tackled specifically in Chapters 6 and 7.

However, the problem does not stop here. In practice, knowing how many agents are required at each time of day then needs to be translated into a workforce schedule: agent start times, end times, breaks and personal preferences all need to be accommodated. This issue becomes yet more complicated when different agents have different skill sets. Complicated workforce management software is available and widely used to try and solve these problems. Problems associated with producing agent schedules from required agent numbers are out of the scope of this thesis, so we point the reader towards [9] for more information.

1.3 Notation

Standard notation exists in queueing theory and it is useful to introduce it now. Note that the number of customers in the *system* refers to both the customers currently in the queue and those currently being served.

- λ - the rate parameter if a homogeneous Poisson arrival process is assumed;
- μ - the rate parameter if service times are assumed to be exponentially distributed;
- $\beta = \mu^{-1}$ - the average service time if service times are assumed to be exponentially distributed;
- γ - average patience of a caller if abandonment times are assumed to be exponential;
- s - number of agents;
- $a = \frac{\lambda}{\mu}$ - the offered load;
- $\rho = \frac{\lambda}{s\mu}$ - the load to the system;
- π - the stationary distribution of the number of customers in the system (if it exists);
- W_Q - the time that an arbitrary customer spends waiting in the queue, if the system is in a stationary situation;

- L_Q - the random number of customers in the queue, if the system is in a stationary situation.

Note that a and ρ are dimensionless, but are said to be measured in *Erlang*, a measure of telecommunications traffic.

1.3.1 Kendall's Notation

Kendall's notation is used in queueing theory to classify different queueing systems. Its general form is

$$A/B/C/k/N/D + E.$$

Here, A refers to the arrival process, whilst B refers to the service time distribution. If we assume a homogeneous Poisson arrival process and exponential service times, then A and B are both written as M, corresponding to *Markovian*. Several other codes are common, including G, corresponding to a general distribution. Some authors write this as GI in order to emphasise the fact that arrivals/service times are independent. If a process is an inhomogeneous Poisson process (a Poisson process with a time-varying rate) then this is written M_t.

C corresponds to the *number of servers* or, in the case of call centres, *agents*. k refers to the capacity of the system, which is the number of agents plus the number of places in the queue for call centres. Once this capacity is filled, further arrivals are *blocked* and are prevented from entering the system. Note that, if $k = C$ then there is never a queue; arrivals can only enter the system when there is a free server/agent. N refers to the calling population. This is often assumed to be infinite when the calling population is large compared to the number of agents. D is the queueing discipline. This is typically *first in, first out (FIFO)* but others are possible, such as *last in, first out (LIFO)*. Finally, if the queueing system includes *abandonments*, then the patience distribution is represented as $+E$. We will return to this in Section 3.4.

When $k = \infty$, $N = \infty$, $D = FIFO$ and no abandonments are assumed, these are often omitted and Kendall's notation becomes $A/B/C$.

Chapter 2

The Erlang C Method

2.1 Queueing Models

Most modern methods for performing network optimisation are based on work done by Agner Erlang almost 100 years ago. He attempted to come up with a method that would give insight into how many trunks are required to carry a certain amount of calling in a telephone network. The answer is that there is always a trade off: increasing the number of trunks results in improved service but increased costs, whilst decreasing the number of trunks decreases costs, but also decreases service quality.

The Erlang C method is the most widely used method in call centres and is based on the $M/M/s$ queueing model. It allows us to calculate the probability that a customer will have to wait longer than a certain time to be served, if various assumptions are satisfied. This then allows us to calculate the minimum number of agents needed to satisfy a service level based on waiting times, of the form described in Section 1.2.2. This method is discussed extensively in Section 2.2. Other queueing models that are also used include the Erlang A (abandonments) model and the Erlang B (blocking) model. The Erlang B model assumes a $M/M/s/s$ queueing system which means that there is never a queue; if a customer calls and there is not a free agent, then the call is blocked. The probability that a call is blocked is given by a simple formula:

$$B(s,a) = \frac{\frac{a^s}{s!}}{\sum_{i=0}^{s} \frac{a^i}{i!}}.$$

In this case, the service level normally defines an acceptable probability that a call is blocked. The minimum number of agents required to achieve this is then easily calculated using the equation above.

The Erlang A model typically assumes a $M/M/s+M$ queue. This means that

whilst no calls are blocked, customers abandonment times follow an exponential distribution with an *average patience*, γ.

Finally, we note a technique known as *square-root safety staffing*. This recommends that the number of agents is determined according to

$$s = a + b\sqrt{a}$$

where a is the offered load and $b \in (0, \infty)$ is known as the *service grade*. Effectively, $b\sqrt{a}$ represents our safeguard against stochastic variability. Formal analysis to support this method only appeared in 1981, however this square root relationship has long been recognised in practice and was used by Erlang himself [9]. This method is examined in much more detail, including analysis suggesting an optimal value for b, in [9].

2.2 The Erlang C Method

2.2.1 Assumptions

We assume that we have a multi-agent single-skill inbound call centre and model the queueing process by a $M/M/s$ queue, which means that arrivals follow a Poisson process with constant arrival rate λ and service times are exponentially distributed with constant rate $1/\mu$. The assumption that the arrival rate follows a Poisson process means that the *inter-arrival times* are independent and identically distributed as exponential random variables with rate $\frac{1}{\lambda}$. We also have s agents serving the calls and an unlimited number of available places in the queue, which means that calls are never blocked: a caller always has a place in the queue.

We will also assume that a caller does not abandon their call whilst they are in the queue; they will always wait to be served. Finally, calls are independent of each other, the service discipline is first come first served and we assume that the system is in a stationary situation.

2.2.2 Formula

If the assumptions above hold, then several useful formulae exist. Recall the *offered load*, $a = \frac{\lambda}{\mu}$ and the *load per agent*, $\rho = \frac{\lambda}{s\mu}$, as defined in Section 1.3. Suppose that $\rho < 1$ and let i denote the number of customers in the *system* (those currently being served as well as those in the queue). Then the stationary distribution for the number

of customers in the system is given by

$$\pi(i) = \begin{cases} \frac{a^i}{i!}\pi(0) & \text{if } i < s, \\ \frac{a^i}{s!s^{i-s}}\pi(0) & \text{otherwise,} \end{cases} \tag{2.1}$$

where

$$\pi(0)^{-1} = \sum_{i=0}^{s-1} \frac{a^i}{i!} + \frac{a^s}{(s-1)!(s-a)}. \tag{2.2}$$

We also have

$$\mathbb{P}(W_Q > T) = C(s,a)e^{-(s\mu-\lambda)T}, \tag{2.3}$$

where

$$C(s,a) = \sum_{i=s}^{\infty} \pi(i) = \frac{a^s}{(s-1)!(s-a)} \left[\sum_{i=0}^{s-1} \frac{a^i}{i!} + \frac{a^s}{(s-1)!(s-a)} \right]^{-1}. \tag{2.4}$$

Finally, we have expressions for the expected time spent waiting in the queue and the expected queue length:

$$\mathbb{E}W_Q = \frac{C(s,a)}{s\mu - \lambda}, \tag{2.5}$$

$$\mathbb{E}L_Q = \frac{\rho C(s,a)}{1-\rho}. \tag{2.6}$$

Instead of proving these expressions here, we will return and prove this later once we have developed a more general framework. Indeed, (2.1) and (2.2) are proved in Section 3.3 whilst (2.3) – (2.6) are proved in Appendix B. For a standard proof, see [10], Section 2.3 or [14].

Note that the probability that a customer has to wait any time at all is given by $C(s,a)$. This is obtained by putting $T = 0$ into (2.3). The condition that $\rho < 1$ is required for stability. This is intuitive; if calls are arriving more quickly on average than the call centre is managing to serve them, so $\lambda > s\mu \iff \rho > 1$, then the queue will continue to grow with no upper bound.

In order to calculate the required number of agents, suppose, as discussed earlier, that the desired grade of service is of the form: answer $\alpha\%$ of calls within T seconds. Then we wish to find the smallest integer s such that

$$\begin{aligned} & \mathbb{P}(W_Q \leq T) > \alpha \\ \iff & \mathbb{P}(W_Q > T) < 1 - \alpha \end{aligned} \tag{2.7}$$

and the left hand side is precisely what is given by (2.3). Efficient calculation of this is discussed in the next section.

2.2.3 Efficient Computation

For large s, we must be careful when attempting to compute $C(s,a)$ numerically as the factorial terms can quickly cause problems. Instead, we find that a better way to compute $C(s,a)$ is to compute its reciprocal first:

$$\begin{aligned}\frac{1}{C(s,a)} &= 1+\frac{\sum_{i=0}^{s-1}\frac{a^i}{i!}}{\frac{a^s}{(s-1)!(s-a)}} \\ &= 1+(s-a)\sum_{i=0}^{s-1}a^{i-s}\frac{(s-1)!}{i!} \\ &= 1+\frac{s-a}{a}\sum_{i=0}^{s-1}\left(\prod_{j=i+1}^{s-1}\frac{j}{a}\right).\end{aligned}$$

Computing $C(s,a)$ in this manner ensures that the numbers remain in a range that can be comfortably handled computationally. However, we can improve the efficiency of the computation by noticing a recurrance relation. If we write

$$S_i=\prod_{j=i+1}^{s-1}\frac{j}{a}$$

then we have that

$$\frac{1}{C(s,a)}=1+\frac{s-a}{a}\sum_{i=0}^{s-1}S_i.$$

It is then straightforward to show that S_i can be calculated from S_{i+1} by

$$S_i=\frac{i+1}{a}S_{i+1}$$

subject to $S_{s-1}=1$. We now have a method of calculating (2.3) efficiently, in $O(s)$ operations in fact, so we move on to the problem of finding s from (2.7). A lower bound for s is given by the stability criterion, so that $s>\frac{\lambda}{\mu}$. We also know from experience that s will be close to this lower bound, hence making it easy to guess an upper bound for s, $\frac{2\lambda}{\mu}$, for example. The bisection method (modified to use only integer values) is one way of efficiently calculating s from (2.7). For more information on the bisection method, see [4], Chapter 2.

2.2.4 Current Practice

Here, we look at how call centre managers typically apply the Erlang C formula. We assume that some forecast has been generated for the arrival rate, over a period of, say, one day. This arrival rate will vary with time, so we cannot apply the Erlang C

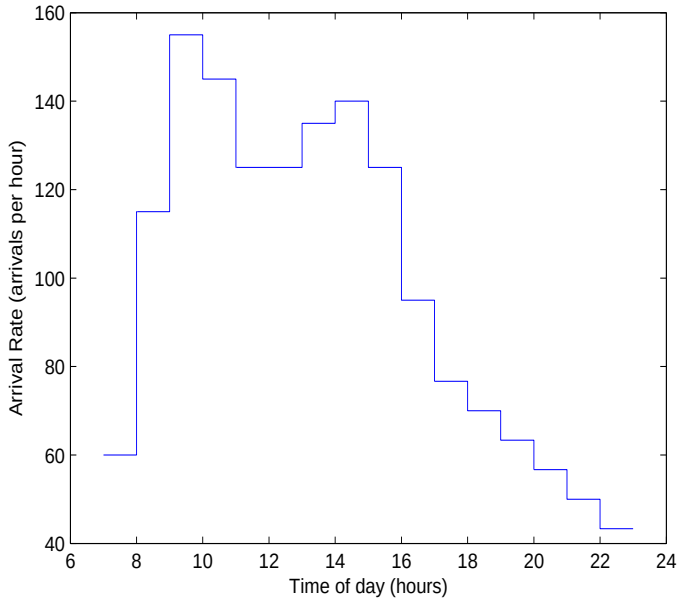


Figure 2.1: Mean arrival rates over one hour intervals corresponding to the arrival rate function given in Figure 1.1

formula directly. Instead, we divide the day into small intervals and approximate the arrival rate by its average value in each interval. Hence, if the arrival rate $\lambda(t)$ varies on the interval $[t_0, t_1]$, the constant Erlang C arrival rate is taken to be

$$\bar{\lambda} = \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} \lambda(t) dt.$$

The mean arrival rates over one hour intervals corresponding to the actual arrival rate function given in Figure 1.1 are shown in Figure 2.1. Whilst we want the intervals to be as small as possible, so that the arrival rate varies as little as possible in each interval, the minimum interval size is often dictated by workforce management issues. Agents can normally only start and end shifts at certain times, on the hour or half hour, for example, limiting the minimum interval size. If we have a point estimate for the (assumed constant) average handling time, then the Erlang C formula can be applied to each interval to produce estimates for the number of agents required in each interval.

2.3 Problems with Current Practice

There are several potential problems with the current practice. Whilst assuming that calls arriving into a call centre follow a Poisson process is justifiable [3], assuming that

they arrive at a known constant rate is not; the rate is neither constant nor known in practice. It is often hoped that the change in arrival rate during each small time interval will be small enough that it is negligible. The assumption that service times are exponentially distributed with a constant rate is also justifiable in many cases [12], but again, this constant rate is treated as known, when it is in fact an estimate.

The Erlang C formula does not take account of caller abandonments or call blocking. Analysis given in [9], Section 4.2.2 suggests that in high traffic situations, even if only a small fraction of calls are either blocked or choose to abandon the queue, this can have a great effect of the performance of the system. The authors state that "a common complaint one hears from call centre managers is that workforce management systems consistently recommend overstaffing" because they fail to take account of abandonments. They go as far as recommending that the Erlang A model should become the standard model, replacing Erlang C. With regard to call blocking, if the number of available lines is large compared to the average queue length, then we can assume the effects of call blocking are small. However, it is suggested in [8] that blocking calls when there is a reasonably long queue may increase customer satisfaction as customers simply hear an engaged tone, rather than waiting for a potentially long time (the length of which is often unknown to the customer) in a queue. In fact, [8] suggests that the number of extra lines should only be 10% of the number of agents.

It is also assumed that the queue is in a stationary situation. Although it is often argued that convergence to stationarity occurs quickly enough to be ignored ([14], Section 15.2), if the queue is longer than expected at the start of an interval, then convergence might not have occurred by the end of the interval. This queue can then take a number of hours to work off. The fact that the arrival rate is unknown can contribute to this; if it is slightly larger than predicted, queues can build up over the course of a day.

2.4 Review of Relevant Literature

In this section, we review some of the research that has been done concerning the issues raised in the previous section. Both [14], Chapter 15 and particularly [9] provide thorough introductions to the issues behind the whole call centre planning process. We will summarise some of the main points made in these documents which are relevant to what we are interested in. Firstly, in [9], Section 4.3, the authors consider a varying arrival rate. They comment that in cases of abrupt changes in the arrival rate, or when the system is overloaded during one or more time intervals, the system

can be far from stationary and this non-stationarity must be accounted for. Whilst uncertainty in the inter-arrival times is modelled explicitly by assuming a Poisson process, the arrival *rate* is assumed known. This is far from true and arrival rates which exceed forecasts by 10% are not unheard of [14]. In [14], it is noted that the inclusion of abandonments is particularly valuable in call centres where the load is high compared to the number of agents. However, it is also stated in [14], Section 15.7 that if abandonments are assumed to be exponentially distributed with a constant average patience, γ, then "estimating this parameter is a non-trivial statistical problem", but that the *Kaplan-Meier estimator* can be used. This is demonstrated in [14], Example 1.8.1. Finally, in [9], the authors recommend using a combination of analytical models and simulation; analytical models for "insight and calibration" and simulation for "fine tuning".

Next, we consider [12]. In this paper, the authors consider the standard Erlang C method, assuming a homogeneous Poisson arrival process and exponential service times. Call volume data studied in [12] suggests that the number of arrivals in a time period is often overdispersed, meaning that the variance is larger than the mean (if the assumption of a homogeneous Poisson arrival process were true, then we would expect the variance to be equal to the mean). The authors respond to this by suggesting that the arrival rate be modelled by a random variable, so that the arrival process is *doubly Poisson*. This makes sense; there will always be variability in the actual arrival rate, no matter how accurate a forecast is and this extra variability needs to be taken into account. Then every data point can be viewed as being generated in two steps. Firstly, the arrival rate, λ, is drawn from the *mixing distribution* (the distribution of the random variable used to model the arrival rate). The arrival count then follows a Poisson process with that rate. If the mixing distribution is known, then we can find a confidence interval for the arrival rate, using this mixing distribution. The number of agents predicted by the Erlang C formula is increasing in the arrival rate, hence the upper and lower bounds from the confidence interval can be inserted into the Erlang C formula to give a confidence interval for the number of agents required. The subject of estimating the mixing distribution is also tackled and both parametric and nonparametric methods are discussed, with the parametric method employing a Gamma distribution to model the arrival rate.

Two different methods of staffing are considered in [12]. The first assumes that the number of agents cannot be altered during a time interval. The upper bound on the number of agents required then becomes important since a worst case scenario must be assumed in order to satisfy the service level as often as possible. However, the lower

bound can be used to show how far off the upper bound can be, which is important as overstaffing translates into unnecessary costs. The second method is preferable; here it is assumed that the number of agents is flexible. We can then assign a number of fixed agents according to the estimate given by the lower bound and a number of flexible agents according to the difference between the two, so that the number of agents answering telephones can be varied between the upper and lower bounds according to real time operating conditions. In a contact centre, flexible agents may be assigned tasks such as answering emails, which are not as urgent, so those agents can be used to answer telephone calls as necessary.

A simulation model for inbound call centres is developed in [20], which includes time varying and uncertain arrival rates as well as varying staffing levels. The authors show that different call centre staffing models are highly sensitive to uncertainties in arrival rates and that performance levels can differ significantly from the target levels, when the arrival rate varies from the forecast. The authors argue that, even though it is common practice, models which assume a known arrival rate are suspect and far from robust.

In [25], the author states that "the queueing model $M/GI/s/k+GI$ has long been regarded as appropriate for call centres". The author then shows that the $M/M/s/k+M(n)$ model often provides an excellent approximation to the $M/GI/s/k+GI$ queueing model. This is extremely helpful; whilst the latter is relatively intractable, the former is not. Note that $M(n)$ refers to the fact that abandonments are exponentially distributed and the abandonment rate is *state dependent*, so the rate at which customers abandon is allowed to depend on their position in the queue. This behaviour can certainly be imagined in call centres that provide customers with information about where they are in the queue, or give customers an expected waiting time. A customer who is told that they have a longer wait, or are further down the queue, is probably more likely to abandon than if they are near to the front.

Complete call-by-call data over the duration of a year is examined in [3]. The data, obtained from a call centre belonging to a bank, included all calls from customers who wished to speak to an agent - about 450 000 in total. The analysis supports the assertion that the arrival process is an inhomogeneous Poisson process with additional randomness in its arrival rate, as suggested in [12]. However, they find that, rather than being exponentially distributed, service times tend to be lognormally distributed. Time to abandonment was curiously found to have two peaks. The first occurred after a few seconds, whilst the second occurred after around 60 seconds. This corresponded to the customer being played a message informing them that they were in a queue,

causing many to give up and hang up. Another interesting result was that the Erlang A model was found to describe the performance of this call centre well and predictions made using it"proved surprisingly robust". This is echoed in [25], where it is stated that the Erlang A model "is certainly superior to Erlang C".

In [2], the authors develop stochastic models of time-dependent arrivals with application to call centres specifically in mind. The focus is on reproducing the behaviour that has been observed in recent empirical studies of call centre arrival data. Firstly, the total daily demand is overdispersed compared to the Poisson distribution (as observed in [12]). Secondly, there are large changes in the arrival rate as the time of day varies (shown in [22]). Thirdly, arrival counts in different time periods are correlated, and finally, arrival counts on successive days are also correlated [3]. The authors develop three models of a time-dependent arrival process, two of which are similar to the doubly stochastic Poisson process suggested in [12]. They examine data obtained from a Bell Canada call centre and find that it exhibits every type of behaviour suggested above. One of the concerns about the doubly stochastic Poisson models is that, although they capture a time-varying arrival rate, they do not support correlation between arrival counts in different time periods.

Fluid approximations to queueing processes have been considered [9, 17, 26] and in [17], the authors allow time varying parameters and not only abandonments, but also retrials. Numerical results show that the simple fluid approximation suggested in [17] is fairly accurate.

2.5 Aims

Based on Section 2.4, we can now state the aims of this thesis. We will attempt to develop a general framework, based on a Markov chain, for examining the $M_t/M/s/k+M(n)$ queue, which includes time-varying arrival rates, as recommended by [25]. We will not assume that the system is stationary (in fact, a stationary situation does not exist when the arrival rate is time-dependent). The Erlang C model then becomes a special case of this and so we hope to be able to examine the inadequacies of the Erlang C method within this framework.

We can then look at some of the properties of the $M_t/M/s/k+M(n)$ queueing model and examine the effect of abandonments on the system, for example. This Markov chain approach will only be able to tell us the average grade of service at any point; in reality the achieved grades of service will vary about this average. It is important to look at the spread of the grade of service and this can be done via

simulation methods, as recommended in [9]. Note that simulation methods are also valuable if the system becomes analytically intractable, as we will see later.

We will also look at optimisation problems to do with minimising the total agent cost over the course of, say, one day, as described in Section 1.2.3. Whilst the arrival rate and the average handling time will still be treated as known, we hope that the system will yield to scenario analysis so that, if a call centre manager knows a worst case scenario, recommendations for the staffing levels can be made.

Chapter 3

Framework

In this chapter, we begin by deriving a method to describe the $M_t/M/s$ queueing model based on a Markov chain. Whilst the average handling time is assumed constant, we allow a time-varying arrival rate. We then extend this method and show how it can be extended to study the $M_t/M/s/k+M$ queue.

3.1 $M_t/M/s$ queue

Consider the $M_t/M/s$ queue with a time-varying arrival rate, $\lambda(t)$. We assume that service times follow an exponential distribution with constant rate $\frac{1}{\mu}$ and denote the constant integer number of agents by s. Now, the number of customers in the system forms a continuous time Markov chain $\{X_t, t \in [0, \infty)\}$, where X_t represents the number of customers in the system at time t. This process must take values in the non-negative integers, however, to make analysis possible, we limit the process to taking values in $\{0, 1, ..., N\}$, where N is chosen to be large enough that there will very rarely be N customers in the system. Note that we have implicitly modelled the $M_t/M/s/k$ queue, where $k = N - s$ (the number of places in the queue) is large. This is discussed further in Section 3.5.

From now on, we will denote $p_i(t) = \mathbb{P}(X_t = i)$, the probability that there are i customers in the system at time t. Before we deal with how these values are calculated, we show how they can be used to calculate the waiting time distribution. Suppose a customer arrives into the system at time τ. Then, applying the law of

total probability:

$$\begin{aligned}\mathbb{P}\left(W_Q \leq T|\tau\right) &= \sum_{i=0}^{N} \mathbb{P}\left(W_Q \leq T|X_\tau = i\right) \mathbb{P}\left(X_\tau = i\right) \\ &= \sum_{i=0}^{N} \mathbb{P}\left(W_Q \leq T|X_\tau = i\right) p_i(\tau). \end{aligned} \tag{3.1}$$

If $i < s$, then an arriving customer will be served immediately, so we must have that

$$\mathbb{P}(W_Q \leq T|X_\tau = i) = 1 \text{ for } i < s. \tag{3.2}$$

If $i \geq s$, then the probability that the customer will be served within time T from now is equal to the probability that more than $i - s$ customers are served and leave the system within time T. We have assumed that service times are exponential with rate $\frac{1}{\mu}$ and so the number of customers leaving the system after being served follows a Poisson process. Hence, we have

$$\begin{aligned}\mathbb{P}(W_Q \leq T|X_\tau = i) &= \mathbb{P}(Y > i - s) \\ &= \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} e^{-T\mu s} \text{ for } i \geq s \end{aligned} \tag{3.3}$$

where $Y \sim Po(T\mu s)$, a Poisson distribution with mean $T\mu s$. Putting (3.2) and (3.3) into (3.1) yields

$$\mathbb{P}(W_Q \leq T|\tau) = \sum_{i=0}^{s-1} p_i(\tau) + \sum_{i=s}^{N} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) p_i(\tau). \tag{3.4}$$

This equation effectively gives an *instantaneous* grade of service; if a customer arrives into the system time τ, then this is the probability that they will be served within time T. However, as described in Section 1.2.2, grades of service are often calculated over periods of time such as an hour. Suppose we wish to calculate the grade of service over the interval $I = [t_0, t_1]$ where the time-dependent arrival rate, $\lambda(t)$, is defined on I. Then the probability that a call arrives into the system at time τ, given that it arrives in I, is given by

$$f(\tau) = \frac{\lambda(\tau)}{\int_{t_0}^{t_1} \lambda(t)dt}.$$

Now, using the continuous version of the law of total probability:

$$\begin{aligned}\mathbb{P}(W_Q \leq T) &= \int_{t_0}^{t_1} \mathbb{P}(W_Q \leq T|\tau) f(\tau) d\tau \\ &= \frac{\int_{t_0}^{t_1} \mathbb{P}(W_Q \leq T|\tau)\lambda(\tau)d\tau}{\int_{t_0}^{t_1} \lambda(t)dt} \end{aligned} \tag{3.5}$$

where $\mathbb{P}(W_Q \leq T|\tau)$ is given by (3.4).

3.2 Time Evolution of the Queue

We now consider the problem of calculating the time evolution of the distribution of the number of customers in the system. We begin by writing the distribution in vector form:

$$\mathbf{p}(t) = (p_0(t), ..., p_N(t))^T .$$

Note that

$$\|\mathbf{p}(t)\|_1 = \sum_{i=0}^{N} p_i(t) = 1 \tag{3.6}$$

since each entry is a probability and the state space is $\{0, ..., N\}$. Suppose that $\mathbf{p}(t)$ is known and consider what will happen in the next time Δt, where Δt is small. Indeed:

$$p_i(t + \Delta t) = \sum_{j=0}^{N} q_{j,i}(t, t + \Delta t) p_j(t) \tag{3.7}$$

where $q_{j,i}(t, t + \Delta t)$ is the probability that we move from state j at time t to state i at time $t + \Delta t$. This is equivalent to the probability that the number of customers arriving into the system minus the number of customers leaving the system in the interval $[t, t + \Delta t]$ is $i - j$.

In the interval $[t, t + \Delta t]$, we know that the number of customers arriving into the system follows a Poisson distribution with rate parameter $\lambda(t)\Delta t$. Similarly, the number of customers leaving the system follows a Poisson distribution with rate $\mu_i \Delta t$, where i is the number of customers currently in the system and $\mu_i = \mu \min(i, s)$. We therefore know that the probability of m customers arriving into the system and n customers leaving in the interval $[t, t + \Delta t]$ is

$$\left(e^{-\lambda(t)\Delta t} \frac{(\lambda(t)\Delta t)^m}{m!} \right) \left(e^{-\mu_i \Delta t} \frac{(\mu_i \Delta t)^n}{n!} \right). \tag{3.8}$$

Now, noting that

$$e^{-(\lambda(t)+\mu_i)\Delta t} = 1 - (\lambda(t) + \mu_i)\Delta t + O(\Delta t^2)$$

we see that (3.8) becomes

$$(1 - (\lambda(t) + \mu_i)\Delta t) \frac{\lambda(t)^m \mu_i^n}{m!n!} \Delta t^{(n+m)} + O(\Delta t^{(n+m+2)}).$$

In particular, if $n + m > 1$, then this is $O(\Delta t^2)$. We now proceed to calculate $q_{j,i}(t, t + \Delta t)$. Firstly:

$$\begin{aligned} q_{i,i}(t, t + \Delta t) &= \mathbb{P}(m = n) \\ &= \mathbb{P}(m = n = 0) + O(\Delta t^2) \\ &= 1 - (\lambda(t) + \mu_i)\Delta t + O(\Delta t^2). \end{aligned}$$

Secondly:

$$\begin{aligned} q_{i,i+1}(t, t+\Delta t) &= \mathbb{P}(m = n+1) \\ &= \mathbb{P}(m = 1, n = 0) + O(\Delta t^2) \\ &= \lambda(t)\Delta t + O(\Delta t^2). \end{aligned}$$

Thirdly:

$$\begin{aligned} q_{i+1,i}(t, t+\Delta t) &= \mathbb{P}(m+1 = n) \\ &= \mathbb{P}(m = 0, n = 1) + O(\Delta t^2) \\ &= \mu_{i+1}\Delta t + O(\Delta t^2). \end{aligned}$$

Finally, for $|i-j| > 1$, $q_{j,i}(t, t+\Delta t) = O(\Delta t^2)$. Now, putting this into (3.7), we obtain:

$$\begin{aligned} p_i(t+\Delta t) &= \sum_{j=0}^{N} q_{j,i}(t, t+\Delta t) p_j(t) \\ &= q_{i-1,i}p_{i-1} + q_{i,i}p_i + q_{i+1,i}p_{i+1} + O(\Delta t^2) \qquad (3.9) \\ &= \lambda(t)\Delta t p_{i-1}(t) + (1 - (\lambda(t) + \mu_i)\Delta t)\, p_i(t) + \mu_{i+1}\Delta t p_{i+1}(t) + O(\Delta t^2) \end{aligned}$$

if $1 \leq i \leq N-1$. We also have the special cases

$$p_0(t+\Delta t) = (1 - (\lambda(t))\Delta t)\, p_0(t) + \mu_1 \Delta t p_1(t) + O(\Delta t^2)$$

and

$$p_N(t+\Delta t) = \lambda(t)\Delta t p_{N-1}(t) + (1 - \mu_N \Delta t)\, p_N(t) + O(\Delta t^2).$$

Re-arranging (3.9), we get:

$$\frac{p_i(t+\Delta t) - p_i(t)}{\Delta t} = \lambda(t)p_{i-1}(t) - (\lambda(t) + \mu_i)p_i(t) + \mu_{i+1}p_{i+1}(t) + O(\Delta t).$$

So, letting $\Delta t \downarrow 0$, we have:

$$\frac{dp_i}{dt} = \lambda(t)p_{i-1}(t) - (\lambda(t) + \mu_i)p_i(t) + \mu_{i+1}p_{i+1}(t).$$

Similarly, for the special cases, we have:

$$\frac{dp_0}{dt} = -\lambda(t)p_0(t) + \mu_1 p_1(t)$$

and

$$\frac{dp_N}{dt} = \lambda(t)p_{N-1}(t) - \mu_N p_N(t).$$

Hence, we have a matrix differential equation for $\mathbf{p}$ of the form

$$\frac{d\mathbf{p}}{dt} = A(t)\mathbf{p}(t) \tag{3.10}$$

where

$$A(t) = \begin{pmatrix} -\lambda(t) & \mu & 0 & 0 & \dots & 0 \\ \lambda(t) & -(\lambda(t)+\mu) & 2\mu & 0 & \dots & 0 \\ 0 & \lambda(t) & -(\lambda(t)+2\mu) & 3\mu & \dots & 0 \\ \vdots & & & & & \vdots \\ 0 & \dots & 0 & \lambda(t) & -(\lambda(t)+s\mu) & s\mu \\ 0 & \dots & 0 & 0 & \lambda(t) & -s\mu \end{pmatrix} \tag{3.11}$$

since $\mu_i = i\mu$ for $i < s$ and $\mu_i = s\mu$ otherwise.

We can solve (3.10) approximately using a Crank-Nicolson method as follows. Assume $\mathbf{p}(0)$ is known and we want to solve (3.10) to obtain $\mathbf{p}(t)$, for some $t > 0$. Let $\Delta t = \frac{t}{M}$ and $t_i = i\Delta t$ for $0 \leq i \leq M$, where M is large. We then approximate (3.10) by

$$\frac{\mathbf{p}_{n+1} - \mathbf{p}_n}{\Delta t} = \frac{1}{2}\left(A_n\mathbf{p}_n + A_{n+1}\mathbf{p}_{n+1}\right) \tag{3.12}$$

where $A_n = A(t_n)$ and $\mathbf{p}_0 = \mathbf{p}(0)$ is known. We hope that $\mathbf{p}_n \approx \mathbf{p}(t_n)$. (3.12) then becomes

$$(I - \frac{\Delta t}{2}A_{n+1})\mathbf{p}_{n+1} = (I + \frac{\Delta t}{2}A_n)\mathbf{p}_n$$

which we can apply iteratively for $0 \leq n \leq M-1$. Thus, we require one matrix solve at each step. The advantage of this method, over a built in higher order differential equation solver in *Matlab*, for example, is that this allows us to construct an approximation for the *transition matrix*, T. This satisfies $\mathbf{p}(t) = T(t)\mathbf{p}(0)$. Define

$$T_n(t) = (I - \frac{\Delta t}{2}A_{n+1})^{-1}(I + \frac{\Delta t}{2}A_n).$$

Then the transition matrix is approximated by

$$T_{M-1}(t)T_{M-2}(t)...T_0(t).$$

This method also gives us an approximation of $\mathbf{p}(t_n)$ for $0 \leq n \leq M$, so that we are able to calculate an instantaneous grade of service at each of these times. Whilst the Crank-Nicolson solution is only approximate, we know that its truncation error is $O(\Delta t^2)$. The derivation of the truncation error is given in Appendix A.

3.3 Properties of the Markov chain

In the case that the arrival rate, λ and the average handling time, μ are constant, a stationary distribution exists, as long as $\frac{\lambda}{s\mu} < 1$. Indeed, in this case, the process $\{X_t, t \geq 0\}$ is a continuous time Markov chain which is both *irreducible* and *aperiodic*. Irreducibility essentially means that *all states communicate*, so it is possible to get from one state to any other state in a finite time. Aperiodicity is guaranteed by the fact that it is always possible to remain in the same state. Moreover, a Markov chain is guaranteed to have a stationary distribution as long as it is both irreducible and aperiodic [19]. Indeed, the stationary distribution is the vector $\mathbf{p}^\infty$ that satisfies

$$\frac{d}{dt}\mathbf{p}^\infty = \mathbf{0}$$

which implies that

$$A\mathbf{p}^\infty = \mathbf{0}. \tag{3.13}$$

Note that a constant arrival rate means that the matrix A is constant too and that the stationary vector is the normalised eigenvector of A corresponding to the zero eigenvalue. This eigenvector can be easily calculated by finding a recurrence relation using (3.11) and (3.13). Indeed, we find that

$$\lambda p_0^\infty = \mu p_1^\infty$$

and

$$\lambda p_{i-1}^\infty - (\lambda + \mu_i)p_i^\infty + \mu_{i+1}p_{i+1}^\infty = 0 \text{ for } i < N$$

and

$$\lambda p_{N-1}^\infty = s\mu p_N^\infty$$

with $\mu_i = \min(i, s)\mu$. Solving these gives each entry of $\mathbf{p}$ in terms of the first:

$$p_i^\infty = \begin{cases} \frac{a^i}{i!}p_0^\infty & \text{if } i < s, \\ \frac{a^i}{s!s^{i-s}}p_0^\infty & \text{if } i \geq s. \end{cases}$$

The first entry of the vector can then be calculated by using the condition that this vector is normalised, as in (3.6), which leads directly to

$$p_0^\infty = \left(\sum_{i=0}^{s-1} \frac{a^i}{i!} + \frac{a^s \left(1 - (\frac{a}{s})^{N-s+1}\right)}{(s-1)!(s-a)} \right)^{-1}.$$

Note that, if we let $N \to \infty$ in the above, since $\frac{\lambda}{\mu s} < 1$, we obtain (2.1) and (2.2), which is the stationary distribution of the $M/M/s$ queue.

3.4 Modelling Abandonments

We model abandonments by modelling a customer's *patience*. Suppose that a customer is given no information as to the probable length of their wait, or their position in the queue whilst they are waiting. Then it may be reasonable to assume that every customer's time to abandonment follows an exponential distribution with constant average patience, γ. However, when customers are given specific information about their position or expected waiting time, the rate of abandonment may well depend on a customer's position in the queue, as noted in Section 2.4 and in [25]. The assumption of exponential times to abandonment with a constant average patience is also motivated by mathematical convenience. Whilst this is fairly simple to include in the model and it is what we do next, other distributions or state-dependent abandonment rates are not and so simulation methods must be used. Simulation methods including abandonments are examined in Chapter 5.

The derivation follows a very similar pattern to that used in Section 3.2 and so we don't repeat it here; rather, we state the results and confine the analysis to Appendix C. Suppose, as before, that a call arrives into the system at time τ and denote the rate parameter of the patience distribution by $\delta = \gamma^{-1}$. Then (3.2) holds as before but (3.3) becomes

$$\begin{aligned}
\mathbb{P}(W_Q \leq T | X_\tau = i) &= \mathbb{P}\left(Y_1 + Y_2(i) > i - s\right) \\
&= 1 - \mathbb{P}\left(Y_1 + Y_2(i) \leq i - s\right) \\
&= 1 - e^{-T(\mu s + \delta(i-s))} \sum_{k=0}^{i-s} \frac{(T\mu s)^k}{k!} \left(\sum_{j=0}^{i-s-k} \frac{(T\delta(i-s))^j}{j!} \right) \text{ for } i \geq s
\end{aligned}$$

where $Y_1 \sim Po(T\mu s)$ and $Y_2(i) \sim Po(T\delta(i-s))$. (3.4) holds as before, giving the instantaneous grade of service, as does (3.5), giving the interval grade of service. We now need to derive the relevant equation for the time evolution of $\mathbf{p}$. Indeed, we find that it is of exactly the same form as (3.10), except that the matrix A becomes

$$A(t) = \begin{pmatrix}
-\lambda(t) & \alpha_1 & 0 & \dots & 0 \\
\lambda(t) & -(\lambda(t) + \alpha_1) & \alpha_2 & \dots & 0 \\
\vdots & & & & \vdots \\
0 & \lambda(t) & -(\lambda(t) + \alpha_i) & \alpha_{i+1} & 0 \\
\vdots & & & & \vdots \\
0 & \dots & \lambda(t) & -(\lambda(t) + \alpha_{N-1}) & \alpha_N \\
0 & \dots & 0 & \lambda(t) & -\alpha_N
\end{pmatrix} \tag{3.14}$$

where

$$\begin{aligned}\alpha_i &= \mu_i + \delta_i \\ &= \min(i, s)\mu + \max(i - s, 0)\delta.\end{aligned}$$

The Crank-Nicolson method can then be used as before in order to approximate the solution to this differential equation. The stationary distribution, in the case of a constant arrival rate, can easily be calculated in the same way as we have done previously, giving:

$$p_i^\infty = \begin{cases} \frac{a^i}{i!} p_0^\infty & \text{if } i < s, \\ \frac{\lambda^i}{s!\mu^s \prod_{j=1}^{i-s}(j\delta + \mu s)} p_0^\infty & \text{if } i \geq s. \end{cases} \tag{3.15}$$

As before, if the number of lines is assumed infinite, $\mathbf{p}_0^\infty$ is found by using the condition

$$\sum_{i=0}^{\infty} p_i^\infty = 1,$$

or if, as in the following section, the number of places in the system is N, then

$$\sum_{i=0}^{N} p_i^\infty = 1.$$

In the case of infinite lines and a constant arrival rate, in contrast to the $M/M/s$ queue, this system is *always stable* if $\delta > 0$. This can be seen by noting that, as $i \to \infty$ in (3.15), $p_i^\infty \to 0$ because of the product in the denominator. Hence a stationary distribution always exists in the case of constant arrival rates and as we see in Chapter 4, convergence to this stationary distribution is guaranteed from any initial distribution.

3.5 Modelling Call Blocking

Our motivation behind limiting the number of places available in the system to N was to make the system solvable; if we hadn't assumed this, we would have a matrix of infinite size. However, this has implicitly introduced the idea of call blocking. Whilst we can make N large and hope that this approximates an infinite line system, in reality there are never infinite lines. Often, call centres may have many lines and use the infinite line case as approximation ([9], Section 4.2.2). So we have accidentally modelled something that is advantageous in practice; setting $N = s + k$ in our model actually describes the $M_t/M/s/k + M$ queue.

The only problem that may be encountered whilst employing this method occurs when one changes the number of agents between time intervals, but wishes to keep the number of lines constant (as would be the case in practice). If the number of agents is increased by c, say, then c zeros can be added to the end of $\mathbf{p}$, whilst A is simply modified to be the $(N+c+1)\times(N+c+1)$ version of (3.11). This represents the fact that there are now c more available positions in the system that were not previously occupied. If the number of agents is reduced by c positions, A is simply modified to be the smaller version of (3.11). The $\mathbf{p}$ vector should be modified so that the last $c+1$ entries are summed to form the last entry of the new, smaller, $\mathbf{p}$ vector. This represents the fact that c positions have been removed from the system.

Chapter 4

System Analysis

4.1 Inadequacies of Erlang C

This framework has the advantage that, not only can we include time-varying arrival rates, but we can also examine the state of the system at any time and calculate the instantaneous grade of service. Knowledge of the state of the system at the end of one time interval can be passed through to the next interval via the $\mathbf{p}$ vector, even if the number of agents has changed. Classical Erlang C methods assume that the system becomes stationary instantly in each time interval, hoping that the transient period is small; this assumption is not necessary with our framework.

Here, we consider two examples. In the first, we suppose that the arrival rate function is as shown in Figure 2.1. This arrival rate is piecewise constant over time intervals of one hour, as assumed by the Erlang C method. We also assume that the average handling time is 7.5 minutes, so each agent deals with 8 calls an hour on average. We assume no abandonments and choose N to be large, so that the effect of call blocking is negligible. The number of agents is calculated according to the Erlang C formula in each interval in order to satisfy a service level requiring more than 80% of customers to wait less than 20 seconds. The initial $\mathbf{p}$ vector is generated from the stationary distribution of the system during the first time interval. Table 4.1 shows the number of agents predicted by the Erlang C method during each time interval, along with the grade of service that the Erlang C method predicted would be achieved with these agent numbers, compared with the grade of service that would actually be achieved according to our framework, taking knowledge about the state of the system from the previous time interval and lack of stationarity into account. Several disparities are obvious. Note that several time intervals have a grade of service which has fallen below the 80% requirement. Figure 4.1 shows how the instantaneous grade of service varies throughout the day. We see here that convergence to stationarity

Time	7-8	8-9	9-10	10-11	11-12	12-13	13-14	14-15
Agents	11	19	24	23	20	20	22	22
Expected	0.8495	0.8524	0.8098	0.8379	0.8217	0.8217	0.864	0.8151
Example 1	0.8495	0.8954	0.8540	0.8224	0.7896	0.8209	0.8709	0.8267
Example 2	0.8605	0.8577	0.8234	0.7828	0.8030	0.8333	0.8701	0.8206
Time	15-16	16-17	17-18	18-19	19-20	20-21	21-22	22-23
Agents	20	16	13	12	12	11	10	9
Expected	0.8217	0.8408	0.8059	0.8015	0.8902	0.8914	0.8936	0.8973
Example 1	0.7969	0.7786	0.7658	0.7856	0.8724	0.8804	0.8822	0.8851
Example 2	0.7601	0.7626	0.7887	0.7853	0.8718	0.8793	0.8809	0.8836

Table 4.1: Erlang C agent numbers with the expected grade of service according to the Erlang C method, together with grades of service for Examples 1 and 2 over each interval during the day

only just occurs by the end of many of the time intervals. This shows that, even if the arrival rate function is exactly as assumed by the Erlang C formula, the lack of stationarity for short periods of time can be problematic.

The second example is the same as the first, except that we suppose the arrival rate varies with time and is piecewise linear, as shown in Figure 1.1. This is a much more realistic situation than the first example. The mean arrival rate in each interval corresponds to the constant arrival rate in the previous example, so calculated agent numbers using the Erlang C formula are identical. Results are shown in Table 4.1 and again differences are seen between the grade of service that the Erlang C formula implies and the actual grade of service.

4.2 Convergence

In this section, we consider the simple case of a constant arrival rate, constant average handling times, no abandonments and a constant number of agents and attempt to understand how the system converges to stationarity. Suppose that the system is initially distributed according to $\mathbf{p}(0)$. We now derive an error bound for the difference between the grade of service at some time, t, and the *limiting* grade of service, that being the grade of service obtained in the stationary situation. Indeed, we begin by considering a more general case. Consider two cases where the initial distributions are given by $\mathbf{p}^1(0)$ and $\mathbf{p}^2(0)$ and suppose that there are no abandonments. We then denote the difference between the grades of service achieved in these cases by d, so that:

$$d(t, \mathbf{p}^1(t), \mathbf{p}^2(t)) = \left|\mathbb{P}(W_Q \leq T | t, \mathbf{p}^1(t)) - \mathbb{P}(W_Q \leq T | t, \mathbf{p}^2(t))\right|.$$

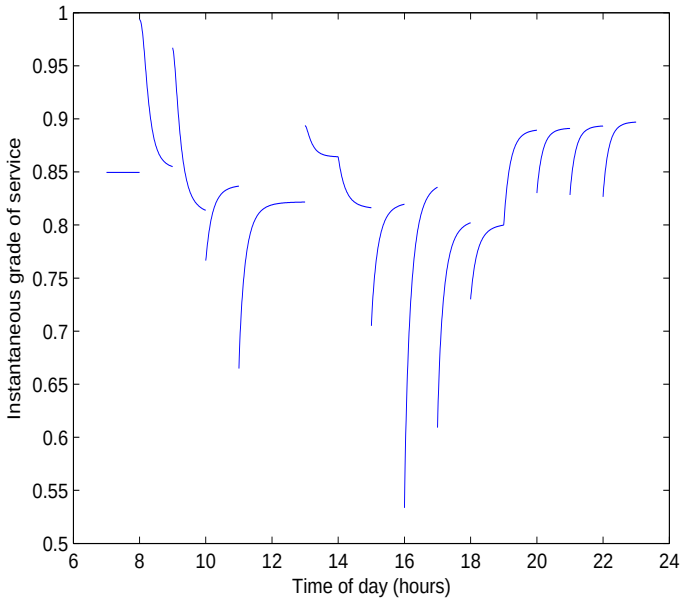


Figure 4.1: Instantaneous grade of service over the course of the day in Example 1

Then

$$\begin{aligned} d(t,\mathbf{p}^1(t),\mathbf{p}^2(t)) &= \Big|\sum_{i=0}^{s-1}(p_i^1-p_i^2)+\sum_{i=s}^{N}e^{-T\mu s}\left(\sum_{k=i-s+1}^{\infty}\frac{(T\mu s)^k}{k!}\right)(p_i^1-p_i^2)\Big| \\ &\leq \sum_{i=0}^{s-1}|(p_i^1-p_i^2)|+\sum_{i=s}^{N}e^{-T\mu s}\left(\sum_{k=i-s+1}^{\infty}\frac{(T\mu s)^k}{k!}\right)|(p_i^1-p_i^2)|. \end{aligned}$$

However, since

$$\sum_{k=0}^{\infty}\frac{(T\mu s)^k}{k!}=e^{T\mu s}$$

it follows that

$$e^{-T\mu s}\left(\sum_{k=i-s+1}^{\infty}\frac{(T\mu s)^k}{k!}\right)\leq 1 \tag{4.1}$$

as long as $i \geq s$. Then

$$\begin{aligned} d(t,\mathbf{p}^1(t),\mathbf{p}^2(t)) &\leq \sum_{i=0}^{s-1}|(p_i^1-p_i^2)|+\sum_{i=s}^{N}|(p_i^1-p_i^2)| \\ &= \|\mathbf{p}^1(t)-\mathbf{p}^2(t)\|_1. \end{aligned} \tag{4.2}$$

We can hope that this is reasonably sharp; indeed, (4.1) becomes less sharp as i increases, but this effect is negated if the entries of $\mathbf{p}^1$ and $\mathbf{p}^2$ are small for large i, which we would hope to be the case if the queue is not too long. Note that a very

similar method shows that this result holds equally if there are abandonments. Now denote the limiting distribution of customers in the system by $\mathbf{p}^\infty$ and the transition matrix by $T(t)$, as in Section 3.2. Then, for any initial vector $\mathbf{p}(0)$, we have

$$\mathbf{p}(t) = T(t)\mathbf{p}(0).$$

Also, if we define $T^\infty = \lim_{t\to\infty} T(t)$, then

$$\mathbf{p}^\infty = T^\infty \mathbf{p}(0).$$

Then

$$\begin{aligned} \|\mathbf{p}(t) - \mathbf{p}^\infty\|_1 &= \|T(t)\mathbf{p}(0) - T^\infty\mathbf{p}(0)\|_1 \\ &= \|\left(T(t) - T^\infty\right)\mathbf{p}(0)\|_1 \\ &\leq \|T(t) - T^\infty\|_1 \end{aligned} \tag{4.3}$$

since $\|\mathbf{p}(0)\|_1 = 1$. Note that equality is experimentally seen to hold in the above if the system is initially "full", so that

$$\mathbf{p}(0) = \mathbf{e}_{N+1} = (0, \ldots, 0, 1)^T.$$

Now, putting (4.2) and (4.3) together, we have

$$d(t, \mathbf{p}(t), \mathbf{p}^\infty) \leq \|T(t) - T^\infty\|_1. \tag{4.4}$$

Note that, as $t \to \infty$, the right hand side of (4.4) goes to zero, meaning that convergence to stationarity is guaranteed from any initial distribution, in the case of constant arrival rates. Equation (4.4) gives us an upper bound on the difference between the instantaneous grade of service when the system is not stationary and the constant instantaneous grade of service achieved in a stationary situation, which, as stated above, we can hope to reasonably sharp.

4.3 Cutoff Phenomenon and Asymptotic Convergence

Here, we show that this Markov chain can exhibit a "cutoff phenomenon" in the case of a constant arrival rate. This means that initially, very little convergence (measured in the 1-norm) occurs, but after some time, convergence to the stationary situation occurs rapidly, in fact at an exponential rate determined by the second eigenvalue of the matrix A. A cutoff phenomenon has been shown to occur in many Markov

chains, including the cases of riffle shuffling a deck of cards and in the Ehrenfest's urn problem [6]. We will base our analysis on that given in [13], which looks at cutoff phenomena from a linear algebra point of view. Indeed, define the decay matrix, D, by

$$D(t) = T(t) - T^{\infty}.$$

In (4.4), we were interested in the 1-norm of the decay matrix and this is precisely the quantity that is examined in [13]. We now consider an example where the arrival rate ($\lambda = 100$ calls per hour) and average handling time ($\beta = 7.5$ minutes) are constant and we assume no abandonments and make N large. The initial distribution is chosen so that the system is *not* in a stationary situation. Indeed, the average queue length according to the initial distribution is longer than the average queue length in the stationary situation. We can then vary the number of agents to show, firstly that this cutoff phenomenon occurs and secondly, how it varies as the number of agents varies. Indeed, Figure 4.2 shows how $\log \|D(t)\|_1$ varies over time for different numbers of agents. Three things are evident from this graph. Firstly, a cutoff phenomenon exists since there is a length of time where relatively little convergence occurs, after which exponential convergence occurs (corresponding to the eventual straight line). Secondly, as the number of agents increases, the "cutoff time" decreases. This is because exponential convergence is seen to set in earlier for the lower curves. Finally, the rate of exponential convergence increases as the number of agents increases since the eventual straight line is steeper for the lower curves. It makes sense that a cutoff phenomenon should be seen. If we think about a situation where we initially have a long queue and a near-zero achieved grade of service, we may have to work off a number of customers from the queue before we begin improving the grade of service.

We now turn our attention to the asymptotic rate of convergence. An important result relating to this is that *all of the eigenvalues of A (and hence T) are real.* This holds regardless of whether the arrival rate is constant or time-varying. A proof of this, using [24], is given in Appendix D. Whilst [13] recommends looking at the pseudospectra of the decay matrix, since all of the eigenvalues of A are real, we suspect that the asymptotic convergence is exponential and its rate is determined by the second eigenvalue of A. Indeed, this is what we experimentally observe. Here, we consider the previous example with only one number of agents. In Figure 4.3, we plot $\log \|D(t)\|_1$ and also a straight line, whose gradient is the second eigenvalue of A. We see that eventually, the two lines are parallel, confirming what we suspected: asymptotic convergence is exponential and its rate is the second eigenvalue of A.

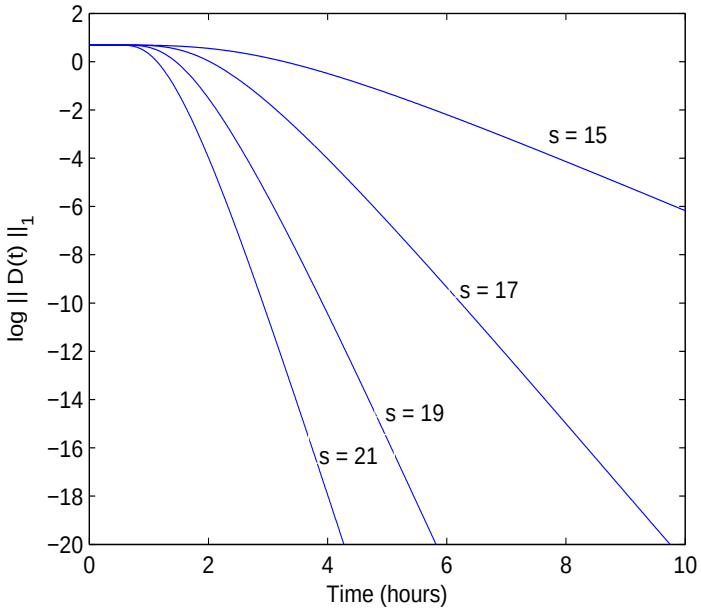


Figure 4.2: $\log \|D(t)\|_1$ plotted for several different agent numbers

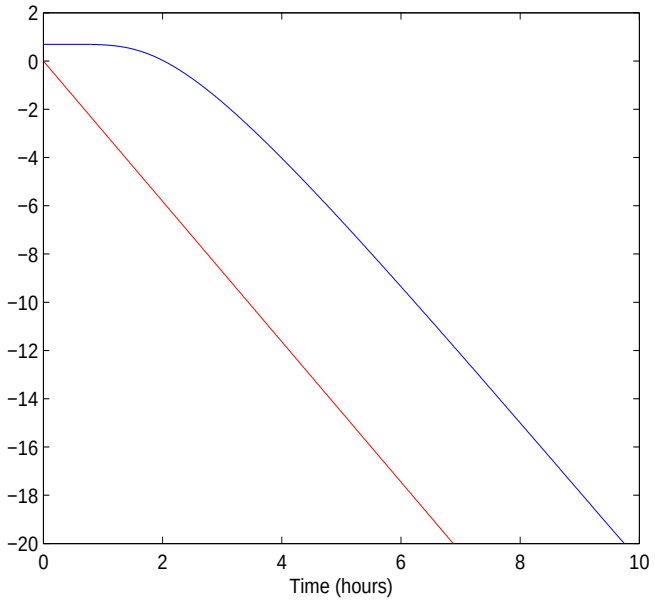


Figure 4.3: $\log \|D(t)\|_1$ with a straight line whose gradient is the second eigenvalue of A

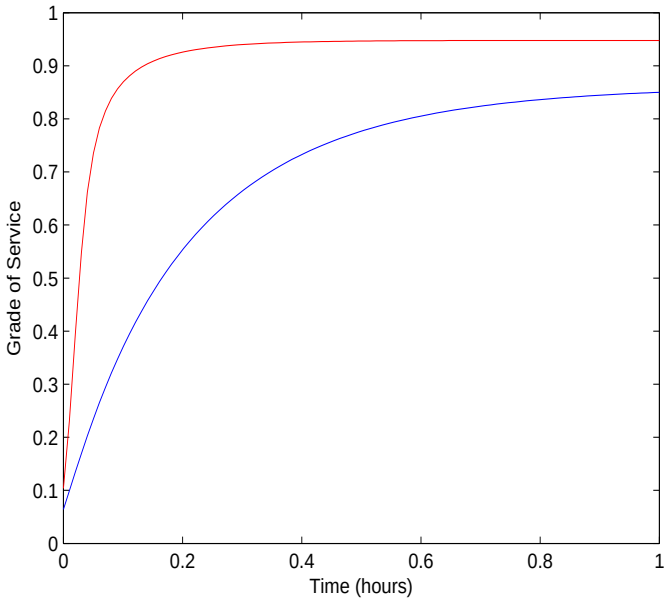


Figure 4.4: Instantaneous grade of service with and without abandonments, when the queue is initially long. The higher curve includes abandonments

4.4 Effect of Abandonments

In situations where the service level requires waiting times to be short, the effect of abandonments may be small, but when waiting times get longer, abandonments may have a big effect. If the queue is longer than expected at the beginning, the effect of abandonments is that the grade of service will improve more quickly than if there are no abandonments, simply because customers are hanging up and shortening the queue. This can be viewed in either a positive or a negative way: whilst the grade of service increases more quickly, the customers who have abandoned are probably very unhappy.

Next, we consider an example. Figure 4.4 shows the difference between the grade of service with and without abandonments, when the queue is initially long. We assume that the arrival rate is $\lambda = 100$ calls per hour, the average service time is 7.5 minutes and the number of agents is 17, chosen so that 80% of customers are served within 20 seconds, according to the Erlang C method. The higher curve assumes that all customers abandonment times follow an exponential distribution with an average patience of 60 seconds. We can see here that the effect of abandonments is that the grade of service improves more quickly and the stationary grade of service is better. We can also examine how abandonments affect the existence of a cutoff

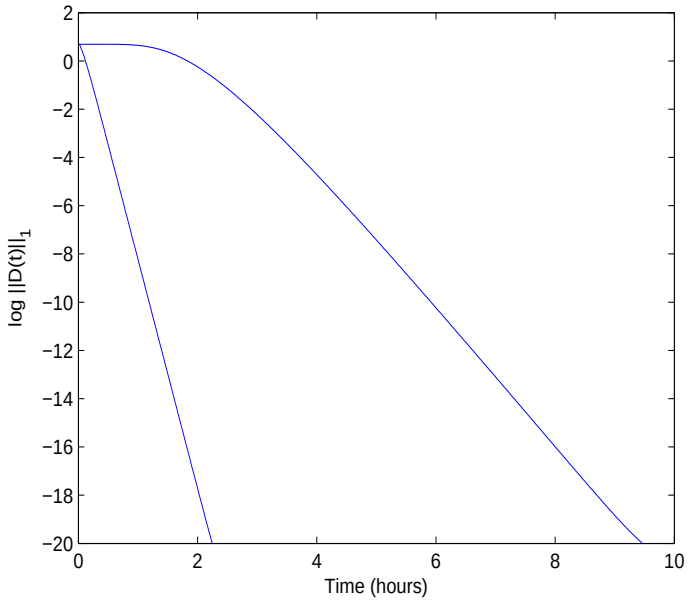


Figure 4.5: $\log \|D(t)\|_1$ plotted with and without abandonments. The lower curve includes abandonments

phenomenon. Indeed, the same example is shown in Figure 4.5. Here, we see that, not only is the cutoff time reduced (almost to zero), but also the asymptotic rate of convergence is larger, if abandonments are included. Note that this asymptotic rate of convergence depends on the second eigenvalue of A in exactly the same way as when no abandonments were assumed.

Chapter 5

Simulation

So far, we have only considered the expected value of the Markov chain. Suppose that we have m customers arriving into the system. Define the satisfaction of the i^{th} customer as

$$S_i = \begin{cases} 1 & \text{if } W_Q \leq T, \\ 0 & \text{if } W_Q > T. \end{cases}$$

We then have that

$$\begin{aligned} \mathbb{E}(S_i) &= 1 \cdot \mathbb{P}(W_Q \leq T) + 0 \cdot \mathbb{P}(W_Q > T) \\ &= \mathbb{P}(W_Q \leq T). \end{aligned}$$

In practice, the achieved grade of service is calculated as

$$S = \frac{1}{m} \sum_{i=1}^{m} S_i$$

and so

$$\mathbb{E}(S) = \mathbb{P}(W_Q \leq T).$$

This is important since the quantity we have referred to so far as the achieved grade of service is in fact the average achieved grade of service; if the process was repeated several times then the average would be $\mathbb{P}(W_Q \leq T)$, but the individual achieved grades of service would be distributed about this average. When measuring the performance of the system, it is important that the spread of these grades of service is not large, otherwise the system performance can be misleading. We demonstrate here the importance of measuring the grade of service over a long enough period. Consider two examples. In both, we suppose the arrival rate is 100 calls per hour, the average handling time is 7.5 minutes, there are no abandonments or blocking and the number of agents is 15, corresponding to an average grade of service of 59.9% of customers having their calls answered within 20 seconds. We simulate the system,

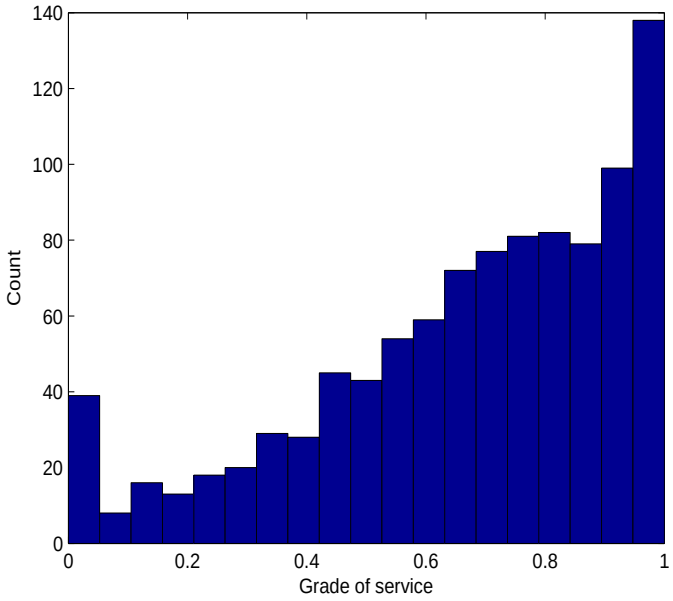


Figure 5.1: 1000 simulations over one hour

firstly over an hour and then over five hours. We repeat this 1000 times and look at the spread of the achieved grades of service in both cases. We see that the histogram is much more spread out in Figure 5.1 than in 5.2, highlighting the importance of measuring the grade of service over a long enough period, in practice.

We now repeat the same example, but include abandonments. We assume that each customer's patience follows an exponential distribution with a constant patience of 60 seconds. A customer is assumed dissatisfied if they abandon. The histogram of achieved grades of service over 1000 simulations is shown in Figure 5.3. This is very interesting indeed. Not only do abandonments improve the grade of service, but they also reduce the spread of achieved grades of service dramatically.

The use of simulations can also be valuable in situations where the system is analytically intractable, for example, when the abandonment rate is state-dependent. Figure 5.4 shows the results of simulating the same example, but assuming that a customer's average patience is state-dependent, according to

$$\gamma(n) = \frac{60}{n},$$

which is measured in seconds, where n is that customer's position in the queue. This means that a customer's patience is shorter if they are further back in the queue. Again, we see that the histogram is less spread. We can also obtain an approximation for the average grade of service by taking the average of the grades of service achieved

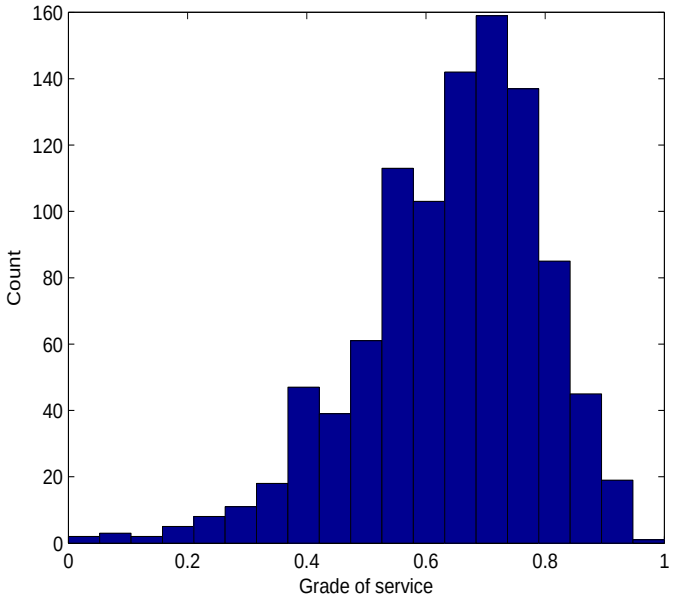


Figure 5.2: 1000 simulations over five hours

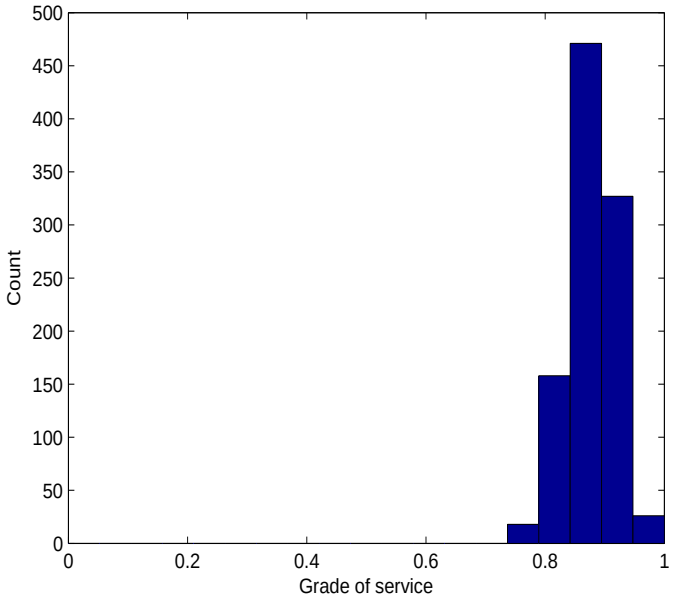


Figure 5.3: 1000 simulations over five hours, assuming abandonments with a constant average patience of 60 seconds

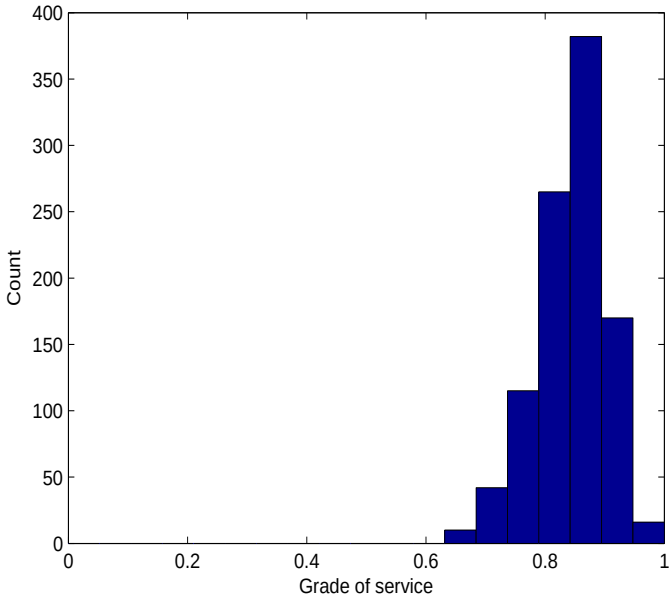


Figure 5.4: 1000 simulations over five hours, assuming abandonments with state-dependent rates

in each simulation. Here, it is 84%. This is useful in cases like this, where analytical methods cannot calculate the average grade of service.

Chapter 6

Optimisation

6.1 Problem Description

We now move on to consider the optimisation problem at the centre of capacity management in call centres. Suppose we have a forecasted daily arrival rate of the form given in Figure 1.1 and a forecasted constant average handling time, over the course of one day. Suppose also that the day has been divided up into time intervals such that, during each time interval, the number of agents must be constant, due to workforce management constraints. Assume that the cost of employing an agent during each interval is constant over the interval, but is allowed to vary between time intervals. We are interested in the problem of minimising the *total agent cost* over the course of the day, whilst achieving a given service level.

Suppose that there are n time intervals in the day given by $I_1, ..., I_n$, with the start and end times of each interval being given by $t_0, t_1, ..., t_n$. Suppose that $\mathbf{p}^i = \mathbf{p}(t_i)$ and $\mathbf{p}^0$ is a given initial distribution, probably calculated from historical data about the number of customers in the system at the start of a day. If the agent numbers are given by $s_1, ..., s_n$ and the cost per agent in each interval is given by $c_1, ..., c_n$, the problem can be written as: find

$$\mathbf{s}^* = \arg\min_{\mathbf{s}\in\mathbb{N}^n} \mathbf{c}^T\mathbf{s}$$

such that

$$\begin{aligned} g(s_1, \mathbf{p}^0) &\geq \alpha, \\ &\vdots \\ g(s_n, \mathbf{p}^{n-1}) &\geq \alpha. \end{aligned}$$

Here, $g(s_i, \mathbf{p}^{i-1})$ represents the achieved grade of service over the interval I_i, using s_i agents and initial distribution $\mathbf{p}^{i-1}$. This might refer to the service level in each

interval as given by (3.5), or we might wish the instantaneous grade of service to satisfy these criteria at all times.

6.2 Full Optimisation Problem

Firstly, we note that the agent numbers produced by the Erlang C formula are rarely even a candidate solution for the problem, meaning that they rarely satisfy the constraints. Solving this problem is expensive since the number of agents chosen in interval i affects the choice in every subsequent interval. This problem must be solved in the following way. Firstly, develop a method to solve the problem: given $\mathbf{p}^{n-1}$, find

$$s_n^*(\mathbf{p}^{n-1}) = \arg\min_{s_n \in \mathbb{N}} c_n s_n \tag{6.1}$$

subject to

$$g(s_n, \mathbf{p}^{n-1}) \geq \alpha. \tag{6.2}$$

Next, develop a method to solve the following problem that can be called recursively: for any $1 \leq i < n$, given $\mathbf{p}^{i-1}$, find

$$\mathbf{s_i^*}(\mathbf{p}^{i-1}) = \arg\min_{s_i \in \mathbb{N}} \left(c_i s_i + \mathbf{c_{i+1}} \mathbf{s_{i+1}^*}(\mathbf{p}^i(s_i, \mathbf{p}^{i-1})) \right) \tag{6.3}$$

subject to

$$g(s_i, \mathbf{p}^{i-1}) \geq \alpha \tag{6.4}$$

where

$$\begin{aligned} \mathbf{s_i} &= (s_i, s_{i+1}, \ldots, s_n)^T \\ \mathbf{c_i} &= (c_i, c_{i+1}, \ldots, c_n)^T. \end{aligned}$$

At each step, there is only one constraint to satisfy, since the other constraints are satisfied by the definition of the optimal subproblem. So the full optimisation problem is solved by developing a method to solve (6.3) and (6.4) for any value of i. Then set $i = 1$ and call this method recursively until we hit the base case given by (6.1) and (6.2), which is easily solved. This is a very expensive problem to solve in general, but information given by the Erlang C method or, particularly by the method given in the next section, can make its solution much easier to find in simple cases by indicating likely values for the solution. Indeed, this is what we will do in Chapter 7. Note that the optimal solution is by no means unique. Having one more agent in the first interval often means one fewer can be used in the next interval, for example.

6.3 Approximate Solution in the Case of Constant Cost

Consider the special case where the agent cost is constant throughout the day. Then we can obtain a candidate solution very quickly as follows. We begin by finding the minimum number of agents that guarantees the service level in the first interval. This is a simple problem as the grade of service is always an increasing function of the agent number. Then calculate the $\mathbf{p}$ vector at the end of this first interval for input into the second interval, using the calculated agent number. Repeat this process through each interval. This gives us agent numbers that satisfy the constraints of the problem and are minimal in the sense that they minimise the cost in the current interval, without regard for the following intervals. Whilst there are no guarantees that this is an optimal solution to the cost minimisation problem in the previous section, the values given are certainly candidate values that can be useful as a starting point in the full optimisation problem. Moreover, when the agent cost through the day is constant, our experience suggests that this solution is very often an optimal solution.

Chapter 7

Case Study

In this section, we suppose that we manage a call centre. We suppose that we have a forecast for the time-varying arrival rate, a point estimate for average handling time, as well as knowledge about a *worst case scenario* for the actual realised arrival rate and for the number of customers in the system at the beginning of the day. We then show how the ideas developed in this thesis can be applied to choose agent numbers which not only satisfy the service level, but also have a built in level of robustness against deviation in the arrival rate or initial distribution away from the forecast. We then use scenario analysis to show how well each choice of agent numbers would perform.

Suppose that the arrival rate is as shown in Figure 7.1. This is very similar to Figure 1.1, except that here, the arrival rate function is piecewise linear over intervals of two hours, rather than one. Note that a complicated arrival rate function does not make the analysis more difficult at any point, so, for this reason, we have been content to use simple arrival rate functions throughout this thesis; the same ideas carry through to more complex functions. Note also that we have lengthened the time intervals to two hours. We do this to simplify the solving of the optimisation problem in Section 6.2. As we will see later, we solve the full problem in this simplified case to add weight to the argument that the method described in Section 6.3 produces near optimal solutions, in the case of constant cost. We assume that the point estimate for the average service time is 7.5 minutes, so that $\mu = 8$ calls per hour. The inclusion of abandonments makes no difference to the difficulty of the analysis, so essentially without loss of generality, we suppose that no abandonments occur. We also suppose that the number of lines is large, so that calls are only blocked if there are very many more customers in the system than normal. The service level that we suppose that we are trying to achieve is: ensure that at least 80% of customers in each time interval are served within 20 seconds. We also assume that a worst case scenario for the arrival

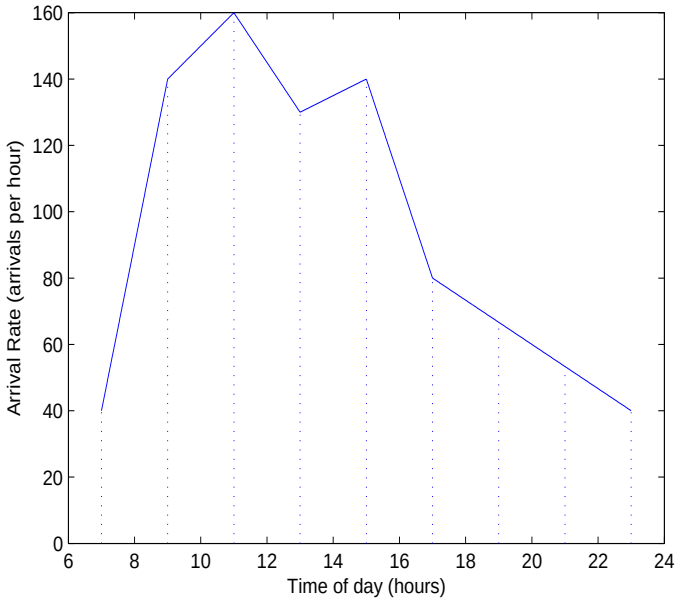


Figure 7.1: Piecewise linear arrival rate function assumed in Chapter 7

rate is that it is 10% above the forecast during the course of the day. In practice, call centres can compare the actual arrival rate with the forecast over historical data in order to find a worst case scenario. This might be a time-varying percentage of the forecast, rather than the constant that we have assumed. Similarly, we suppose that a worst case scenario for the length of the queue at the beginning of the day is that there are initially 40 customers in the system. This is then be represented

$$\mathbf{p}(0) = \mathbf{e}_{41} = (0, \ldots, 0, 1, 0, \ldots, 0)^T$$

where the 1 lies in the position 41.

7.1 Optimisation

Here, we suppose that the cost of employing an agent is constant throughout the day. Suppose that the aim of the call centre manager is to satisfy the service level in most cases, but at the minimum cost. In practice, it is also possible that a call centre manager might be given a certain budget and told to achieve the best possible service level with that budget. In this case, the various methods used below to choose the number of agents would need to modified accordingly.

There are several ways of choosing the number of agents required over each time interval. The first method is obvious: solve the optimisation problem in Section 6.2

with the forecast arrival rate. Secondly, we can solve the optimisation problem, but assume a worst case scenario for the arrival rate. This may lead to overstaffing most of the time. If flexible agents are available, as described in Section 2.4, we may want to have a number of fixed agents according to the forecast arrival rate and a number of flexible agents corresponding to the difference between the two sets of agent numbers.

Recall from Chapter 4 the importance of the second eigenvalue of the matrix $A(t)$. This, in some sense, defines how robust the system is to an inflated arrival rate or to a long initial queue. Hence, we may wish to ensure that the second eigenvalue is below a certain threshold, certainly in those parts of the day when an increased arrival rate is possible, or at the start of the day, if a large queue at the start of the day is seen regularly. We consider the following different constraints whilst choosing the agent numbers:

1. use Erlang C agent numbers;
2. satisfy the service level assuming the forecast arrival rate;
3. satisfy the service level assuming the forecast arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -1 at all times;
4. satisfy the service level assuming the forecast arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -2 only in the first time interval, in case the queue is long at the start of the day;
5. satisfy the service level assuming the forecast arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -2 at all times;
6. satisfy the service level assuming worst case scenario arrival rate;
7. satisfy the service level assuming worst case scenario arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -1 at all times;
8. satisfy the service level assuming worst case scenario arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -2 only in the first time interval, in case the queue is long at the start of the day;
9. satisfy the service level assuming worst case scenario arrival rate and guarantee that the second eigenvalue of $A(t)$ is smaller than -2 at all times.

Method	Time Interval								Cost
	7-9	9-11	11-13	13-15	15-17	17-19	19-21	21-23	
1	15	24	23	22	18	13	11	9	135
2	17	24	24	21	20	13	11	9	139
3	19	24	24	21	20	13	11	9	141
4	21	23	24	21	20	13	11	9	142
5	21	23	24	21	21	13	12	10	145
6	18	26	26	23	22	14	12	10	151
7	20	26	26	23	22	14	12	10	153
8	23	25	26	23	22	14	12	10	155
9	23	25	26	23	22	14	13	11	157

Table 7.1: Agent numbers in each time interval as calculated according to various different requirements

Obviously these are only a few of an infinite number of possible staffing methods. Solving the optimisation problem subject to the criteria given above leads to the agent numbers given in Table 7.1. Since agent costs are constant throughout the day, we can apply the method described in Section 6.3 to get a fast approximate solution and we find that in all nine of the above cases, the agent numbers given by this suboptimal method have the same total cost as those given by the full optimisation problem. Whilst the two solutions were different in two of the nine cases, the cost was the same and the difference was purely down to the implementation. It might be desirable to have a criteria for choosing between different optimal solutions, for example, if we expect heavier traffic at the start of the day, we might choose the optimal solution that has the most agents at the start of the day and this is what we have done here.

This suggests that the method described in Section 6.3 is indeed a very good way of finding agent numbers when costs are constant and the number of time intervals makes solving the full optimisation problem impossible.

7.2 Scenario Analysis

We are now in a position to use these calculated agent numbers to test how well the system copes under various scenarios. We vary the system in the following ways. Unless otherwise mentioned, we take the value of the arrival rate function at the beginning of the day and calculate the stationary distribution as if this were a constant arrival rate. This is then used as the initial distribution. In practice, a call centre

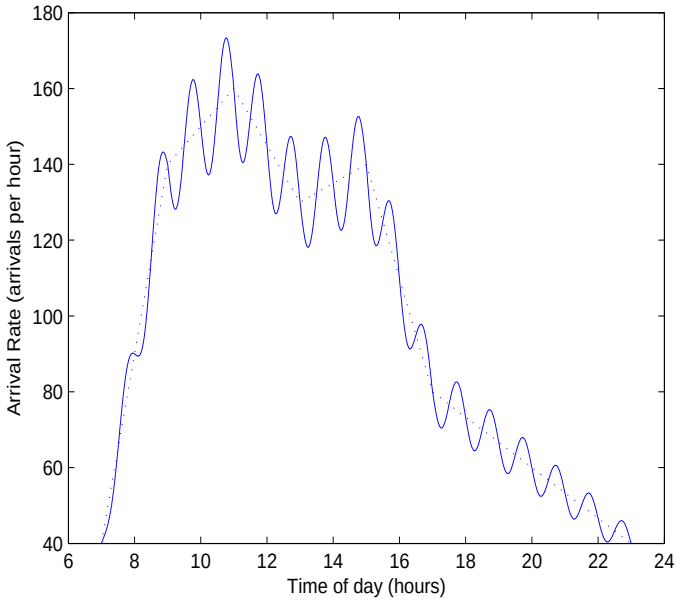


Figure 7.2: Sinosoidal error in the arrival rate function in Scenario 3

manager would have data about the likely number of people in the system at the start of a day. The scenarios we consider are:

1. the arrival rate is as forecast;
2. the arrival rate is 10% more than forecast all day;
3. the arrival rate follows the forecast but with a sinosoidal error around the forecast, with peaks at 10% of the forecast. This is shown in Figure 7.2;
4. the initial distribution is $\mathbf{p}(0) = \mathbf{e}_{41}$, corresponding to many more people in the system than expected;
5. the busy part of the day (around lunchtime) experiences much more traffic than forecast. The exact arrival rate function used in this scenario is shown in Figure 7.3.

We can then produce tables of results for each scenario by running each set of agent numbers through our framework, tailored for each scenario and looking at the grades of service that are achieved for each method. The results produced for Scenario 1 are given in Table 7.2. Here, we consider three ways of looking at the grade of service. Firstly, we consider the instantaneous grade of service. Its maximum, minimum and mean over the day are given. Secondly, we consider the grade of service as calculated

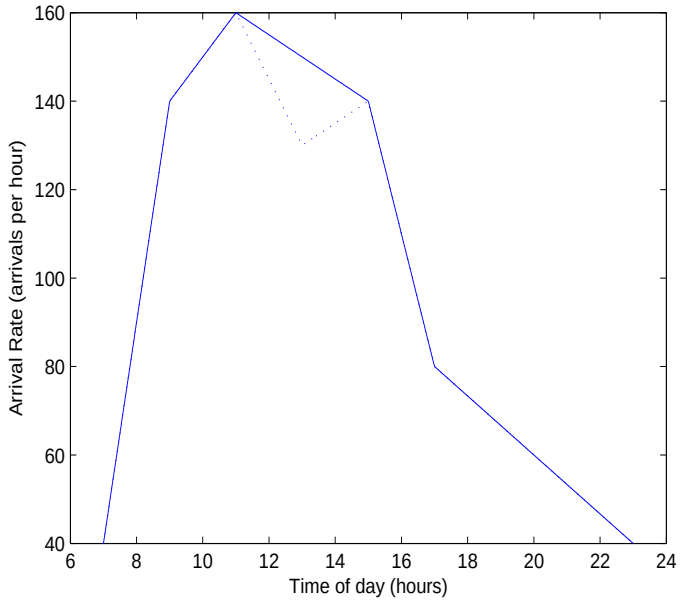


Figure 7.3: Extra traffic during the busy part of the day in Scenario 5

by (3.5), over the course of each interval. The maximum, minimum and mean of these eight interval values are given. Finally, the grade of service given by (3.5) over the course of the whole day is given in the final column. Tables of results for all five scenarios are given in Appendix E and they are analysed in the following section.

7.3 Results

Firstly, we examine the results of Scenario 1. Recall that the total number of agents employed increases as we go down the table. The main point shown by Scenario 1 is that the Erlang C agent numbers do not perform well: several interval grades of service and in fact the average of the interval grades of service fall below the 80% point. We also note that, whilst methods 2 to 9 satisfy the desired grade of service, methods 6 to 9 perhaps do this a little too well, leading to unnecessary cost.

Scenario 2 supposes that the worst case arrival rate occurs. We see that methods 6 to 9 satisfy the service level with little to spare. This should be the case: they were chosen to do precisely that. However, the lack of robustness to an unknown arrival rate in methods 2 to 5 is exposed here: the service levels are quite a way below what we desire. If agents numbers are fixed, so there are no flexible agents, it would be up to an individual call centre manager to decide whether this is an acceptable

Scenario 1	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.1877	0.8108	0.9993	0.6859	0.7963	0.8693	0.7970
2	0.4283	0.8448	0.9999	0.8027	0.8353	0.8625	0.8378
3	0.6666	0.8542	1.0000	0.8027	0.8473	0.9196	0.8491
4	0.6682	0.8503	1.0000	0.8027	0.8448	0.9664	0.8420
5	0.6882	0.8763	1.0000	0.8027	0.8718	0.9664	0.8597
6	0.5536	0.9248	1.0000	0.8819	0.9184	0.9496	0.9243
7	0.7611	0.9313	1.0000	0.9055	0.9270	0.9496	0.9321
8	0.8431	0.9309	1.0000	0.9059	0.9282	0.9876	0.9300
9	0.8431	0.9420	1.0000	0.9059	0.9396	0.9876	0.9359

Table 7.2: Results from Scenario 1

service level when faced with the worst case scenario, or whether methods 2 to 5 are unacceptable.

Scenario 3 examines the effect of an arrival rate which oscillates around the forecast with an amplitude of 10% of the arrival rate function. We see very little difference between the results here and those in Scenario 1, showing that these methods are quite robust to this type of error. Whilst the interval grade of service in methods 2 to 5 occasionally falls below 80%, it is by approximately 1%, which must surely be regarded as acceptable.

Scenario 4 examines what happens if the queue is much longer than expected at the start of the day. In all other cases, the expected initial queue length was less than 1 person. Here, we assume there are 40 people in the system at the beginning of the day. We see that, in all cases, there was a time when the instantaneous grade of service was zero, while the system was working off the initial queue. However, methods 3 to 9 all recover from this sufficiently enough that every interval satisfies the grade of service requirement. Note the difference between methods 2 and 3. Method 3 uses a mere 2 extra agent slots and dramatically improves the minimum grade of service over the intervals. This demonstrates the value of requiring the second eigenvalue to be below a certain threshold: robustness against the unexpected is built in. Method 4 uses one more agent than 3 and was specifically designed to build in robustness against a long queue at the start of the day, but does not perform significantly better. This is probably due to the fact that method 3 already had enough robustness built in at the beginning of the day to deal with our worst case and the extra given by method 4 is unnecessary.

Finally, Scenario 5 supposes that the busy period sees a higher arrival rate than expected, as shown in Figure 7.3. This has relatively little effect on the average

grades of service over the course of the day, but the minimum grades of service are particularly low for methods 2 to 5.

Based on this scenario analysis, we can make a recommendation as to the staffing levels that should be used. Method 6, based on satisfying the service level constraints only in a worst case scenario, with no regard for eigenvalues, achieves the grade of service in all of the scenarios considered, but does far too well in some cases. Method 3 is based on satisfying the service level constraints for the forecast arrival rate function, but also requires the eigenvalues to be smaller than -1. This satisfies the service level criteria for the forecast arrival rate, but also builds in some robustness against the varying arrival rate. Method 4 aims specifically to counter a long queue at the beginning of the day, but we have established that method 4 adds no value compared to method 3. Hence, a recommendation might be to have a fixed number

	Time Interval								
Agent Type	7-9	9-11	11-13	13-15	15-17	17-19	19-21	21-23	Cost
Fixed	19	24	24	21	20	13	11	9	141
Flexible	0	2	2	2	2	1	1	1	11

Table 7.3: Suggested staffing levels

of agents corresponding to method 3, with a flexible number corresponding to the difference between the method 3 and method 6, which could be called upon if required, depending on real-time operating conditions. This would correspond to the staffing levels shown in Table 7.3.

Finally, we note that in practice, if a call centre manager knows of a *best case scenario* for the arrival rate and initial distribution, then they may wish to take the number of fixed agents as calculated using these best case scenarios, thereby also profiting from the occasions when the arrival rate is below the forecast or when there are fewer customers in the system at the beginning of the day.

Chapter 8

Discussion

8.1 Summary

In Chapters 1 and 2, we examine how a typical call centre operates and the methods normally employed to calculate the number of agents required. We look at the flaws in these methods and some of the work that has already been done in this area. Based on this, we develop the aims described in Section 2.5.

In Chapter 3, we develop a more general framework in order to describe the $M_t/M/s/k + M$ queueing model which includes a time-varying arrival rate. We discuss when stationary distributions exist and derive them when they do and we discuss how abandonments and call blocking are modelled.

In Chapter 4, we begin by using our framework to demonstrate the inadequacies of the current method. We examine how the system converges to stationarity (if a stationary situation exists) and find that this is strongly dependent on the second eigenvalue of a certain matrix. We show that a cutoff phenomenon can occur, which is intuitively obvious if we imagine a queue which is much longer than expected. We also briefly touch on the effect of abandonments.

Simulation methods are examined in Chapter 5. We show the importance of examining the spread of the grade of service as well as its average and find that abandonments can significantly reduce this spread. We also show the value of simulation methods when the queueing model becomes analytically intractable.

In Chapter 6, we consider the optimisation problem of minimising the cost of employing agents over a day, in order to meet a service level. We set up the optimisation problem in full, but also explain a method of finding an approximate solution very quickly, as long as the agent cost is constant over the day.

Chapter 7 essentially demonstrates how we imagine this work being used in practice. We set up a situation where we have a forecast for the time-varying arrival rate

and a point estimate for the average handling time. We then use various different constraints in the optimisation problem based on the arrival rate and the second eigenvalue of A, to choose the number of agents. Interestingly, we assume constant agent cost and the approximate number of agents given by the method described in Section 6.3 is actually optimal in every case. We suppose that we know a worst case scenario for both the arrival rate and the state of the queue at the start of the day and perform scenario analysis to examine how each method performs. Based on this, we make recommendations about the number of agents required.

8.2 Extensions

There are several obvious extensions to this thesis. Due to time constraints, we only briefly touched on abandonments and call blocking, despite both being discovered to be important in Section 2.4, where it was noted that the number of lines should be restricted, otherwise waiting times become to long. Based on this, work which examines the effects of deliberately blocking calls could be productive. Whilst the $M_t/M/s/k+M(n)$ queueing model seemed to be the best to study, we did not manage to incorporate state dependent abandonment rates into the model and this would certainly be worthwhile. Also, we did not consider the effects of retrials. Customers who abandon may well try again later and this could also be incorporated into the model.

In Chapter 7, we only examined a few methods of choosing the agent numbers. We believe that it could be beneficial to extend this work to study if there is a better method of controlling the second eigenvalue of A, rather than simply restricting it to be below a certain threshold, or even carrying out work to find the optimum threshold, if one exists.

8.3 Conclusion

This thesis has allowed an insight into the complex planning processes which occur in most modern day call centres and the problems that can arise when employing these methods. Hopefully this research, particularly the methods described in Chapter 7, will eventually prove useful and improve the effectiveness of planning processes in call centres.

References

[1] Ian Angus. An Introduction to Erlang B and Erlang C. *Telemanagement*, **187**:6–8, 2001.

[2] Athanassios N. Avramidis, Alexandre Deslauriers, and Pierre L'Ecuyer. Modelling Daily Arrivals to a Telephone Call Center. *Management Science*, **50**:896–908, 2004.

[3] Lawrence Brown, Noah Gans, Avishai Mandelbaum, Anat Sakov, Haipeng Shen, Sergey Zeltyn, and Linda Zhao. Statistical Analysis of a Telephone Call Center: A Queueing-Science Perspective. *J. Amer. Statist. Assoc.*, **100**:36–50, 2005.

[4] Richard L. Burden, J. Douglas Faires, and Albert C. Reynolds. *Numerical Analysis*. Prindle, Weber and Schmidt, Boston, 1981.

[5] D. R. Cox and Walter L. Smith. *Queues*. Chapman and Hall, London, 1971.

[6] P. Diaconis. The Cutoff Phenomenon in Finite Markov Chains. *Proc. Natl. Acad. Sci. USA*, **93**:1659–1664, 1996.

[7] J. C. Duder and M. B. Rosenwein. Towards "Zero Abandonments" in Call Center Performance. *European Journal of Operational Research*, **135**(1):50–56, 2001.

[8] M.A. Feinberg. Performance Characteristics of Automated Call Distribution Systems. *Global Telecommunications Conference*, **1**:415–419, 1990.

[9] Noah Gans, Ger Koole, and Avishai Mandelbaum. Telephone Call Centers: Tutorial, Review, and Research Prospects. *Manufacturing and Service Operations Management*, **5**:79–141, 2003.

[10] Donald Gross and Carl M. Harris. *Fundamentals of Queueing Theory.* Wiley, New York; Chichester, 1998.

[11] Gawain Heckley. Offshoring and the Labour Market: IT and Call Centres Considered. Available from: http://www.statistics.gov.uk.

[12] Geurt Jongbloed and Ger Koole. Managing Uncertainty in Call Centers using Poisson Mixtures. *Applied Stochastic Models in Business and Industry*, **17**:307–318, 2001.

[13] G. J. Jonsson and Lloyd N. Trefethen. A Numerical Analyst looks at the "Cutoff Phenomenon" in Card Shuffling and other Markov Chains. *Numerical Analysis 1997*, pages 150–178, 1997.

[14] Ger Koole. Optimization of Business Processes: An Introduction to Applied Stochastic Modelling. Available from: http://obp.math.vu.nl/callcenters/.

[15] Ger Koole. Performance Analysis and Optimization in Customer Contact Centers. *QEST*, pages 2–5, 2004.

[16] Avishai Mandelbaum. Call Centres: Research Bibliography with Abstracts. Available from: http://ie.technion.ac.il/serveng, 2004.

[17] Avishai Mandelbaum, William A. Massey, Martin I. Reiman, and Brian Rider. Time Varying Multiserver Queues with Abandonments and Retrials. *Proceedings of the 16th International Teletraffic Conference*, pages 355–364, 1999.

[18] J.R. Norris. *Markov Chains.* CUP, Cambridge, 1997.

[19] Alessandro Panconesi. The Stationary Distribution of a Markov Chain. Available from: http://www.dis.uniroma1.it/ leon/didattica/webir/ pagerank.pdf, 2005.

[20] Thomas R. Robbins, D. J. Medeiros, and Paul Dum. Evaluating Arrival Rate Uncertainty in Call Centers. *Proc. 38th Winter Simulation Conf.*, pages 2180–2187, 2006.

[21] Sheldon M. Ross. *Introduction to Probability Models.* Academic Press, Amsterdam; London, 2003.

[22] Oryal Tanir and Richard J. Booth. Call Center Simulation in Bell Canada. *Proc. 1999 Winter Simulation Conf.*, pages 1640–1647, 1999.

[23] Lloyd N. Trefethen and David Bau III. *Numerical Linear Algebra*. SIAM, Philadelphia, 1997.

[24] Leonardo Volpi. Similarity Transform for Unsymmetrical Tridiagonal Matrix. Available from: http://digilander.libero.it/foxes/documents.htm, 2003.

[25] Ward Whitt. Engineering Solution of a Basic Call Center Model. *Management Sci.*, **51(2)**:221–235, 2005.

[26] Ward Whitt. Staffing a Call Center with Uncertain Arrival Rate and Absenteeism. *Production And Operations Management*, **15(1)**:88–102, 2006.

Appendix A

Truncation Error of Crank-Nicolson Method

Recall that we approximated the matrix differential equation (3.10) by

$$\frac{\mathbf{p}_{n+1}-\mathbf{p}_n}{\Delta t}=\frac{1}{2}\left(A(t_n)\mathbf{p}_n+A(t_{n+1})\mathbf{p}_{n+1}\right).$$

The truncation error of this method is

$$\mathbf{T}^n=\frac{\mathbf{p}(t_{n+1})-\mathbf{p}(t_n)}{\Delta t}-\frac{1}{2}\left(A(t_n)\mathbf{p}(t_n)+A(t_{n+1})\mathbf{p}(t_{n+1})\right). \tag{A.1}$$

We now evaluate the Taylor expansions of $\mathbf{p}(t_{n+1})$ and $\mathbf{p}(t_n)$ about $\mathbf{p}(t_{n+\frac{1}{2}})$:

$$\begin{aligned}\mathbf{p}(t_n) &= \mathbf{p}(t_{n+\frac{1}{2}})-\frac{1}{2}\Delta t\mathbf{p}'(t_{n+\frac{1}{2}})+\frac{1}{8}(\Delta t)^2\mathbf{p}''(t_{n+\frac{1}{2}})+O(\Delta t^3),\\ \mathbf{p}(t_{n+1}) &= \mathbf{p}(t_{n+\frac{1}{2}})+\frac{1}{2}\Delta t\mathbf{p}'(t_{n+\frac{1}{2}})+\frac{1}{8}(\Delta t)^2\mathbf{p}''(t_{n+\frac{1}{2}})+O(\Delta t^3),\end{aligned}$$

and so

$$\frac{\mathbf{p}(t_{n+1})-\mathbf{p}(t_n)}{\Delta t}=\mathbf{p}'(t_{n+\frac{1}{2}})+O(\Delta t^2). \tag{A.2}$$

We now do the same for $A(t_n)\mathbf{p}(t_n)$ and $A(t_{n+1})\mathbf{p}(t_{n+1})$ about $A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})$:

$$\begin{aligned}A(t_n)\mathbf{p}(t_n) &= A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})-\frac{1}{2}\Delta t\left(A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})\right)'+O(\Delta t^2),\\ A(t_{n+1})\mathbf{p}(t_{n+1}) &= A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})+\frac{1}{2}\Delta t\left(A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})\right)'+O(\Delta t^2),\end{aligned}$$

and so

$$\frac{1}{2}\left(A(t_n)\mathbf{p}(t_n)+A(t_{n+1})\mathbf{p}(t_{n+1})\right)=A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})+O(\Delta t^2). \tag{A.3}$$

Now, putting (A.2) and (A.3) into (A.1), we get

$$\mathbf{T}^n=\mathbf{p}'(t_{n+\frac{1}{2}})-A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})+O(\Delta t^2).$$

But (3.10) implies that we must have

$$\mathbf{p}'(t_{n+\frac{1}{2}}) = A(t_{n+\frac{1}{2}})\mathbf{p}(t_{n+\frac{1}{2}})$$

and so

$$\mathbf{T}^n = O(\Delta t^2).$$

Appendix B

Derivation of Equations (2.3) – (2.6)

We begin by deriving (2.3), using (2.1), (2.2) and (3.4). Indeed, assuming a constant arrival rate and putting (2.1) into (3.4) gives:

$$\mathbb{P}(W_Q \leq T) = \sum_{i=0}^{s-1} \frac{a^i}{i!}\pi(0) + \sum_{i=s}^{\infty} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) \frac{a^i}{s!s^{i-s}}\pi(0).$$

Then, re-arranging the above and using (2.2):

$$\begin{aligned}
\mathbb{P}(W_Q > T) &= 1 - \sum_{i=0}^{s-1} \frac{a^i}{i!}\pi(0) - \sum_{i=s}^{\infty} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) \frac{a^i}{s!s^{i-s}}\pi(0) \\
&= \left(\pi(0)^{-1} - \sum_{i=0}^{s-1} \frac{a^i}{i!} - \sum_{i=s}^{\infty} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) \frac{a^i}{s!s^{i-s}} \right) \pi(0) \\
&= \left(\frac{a^s}{(s-1)!(s-a)} - \sum_{i=s}^{\infty} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) \frac{a^i}{s!s^{i-s}} \right) \pi(0) \\
&= \frac{a^s}{(s-1)!(s-a)} \left(1 - \sum_{i=s}^{\infty} \left(e^{-T\mu s} \sum_{k=i-s+1}^{\infty} \frac{(T\mu s)^k}{k!} \right) \frac{a^{i-s}(s-a)}{ss^{i-s}} \right) \pi(0).
\end{aligned}$$

Now, defining

$$C(s,a) = \frac{a^s}{(s-1)!(s-a)}\pi(0)$$

gives (2.4). Noting that

$$\begin{aligned}
1 &= (1-\rho)\sum_{j=0}^{\infty} \rho^j \\
&= (1-\rho)e^{-T\mu s} \sum_{j=0}^{\infty} \rho^j \sum_{k=0}^{\infty} \frac{(T\mu s)^k}{k!}
\end{aligned}$$

and letting $j = i - s$:

$$\begin{aligned}
\mathbb{P}(W_Q > T) &= C(s,a)e^{-T\mu s}\left((1-\rho)\sum_{j=0}^{\infty}\rho^j\sum_{k=0}^{\infty}\frac{(T\mu s)^k}{k!} - (1-\rho)\sum_{j=0}^{\infty}\rho^j\sum_{k=j+1}^{\infty}\frac{(T\mu s)^k}{k!}\right) \\
&= C(s,a)e^{-T\mu s}\left((1-\rho)\sum_{j=0}^{\infty}\rho^j\sum_{k=0}^{j}\frac{(T\mu s)^k}{k!}\right).
\end{aligned}$$

Changing the order of summation:

$$\begin{aligned}
\mathbb{P}(W_Q > T) &= C(s,a)e^{-T\mu s}\left((1-\rho)\sum_{k=0}^{\infty}\frac{(T\mu s)^k}{k!}\sum_{j=k}^{\infty}\rho^j\right) \\
&= C(s,a)e^{-T\mu s}\left((1-\rho)\sum_{k=0}^{\infty}\frac{(T\mu s)^k}{k!}\frac{\rho^k}{1-\rho}\right) \\
&= C(s,a)e^{-T\mu s}\sum_{k=0}^{\infty}\frac{(\rho T\mu s)^k}{k!} \\
&= C(s,a)e^{-T\mu s}e^{\rho T\mu s}.
\end{aligned}$$

However, $\rho T\mu s = \lambda T$ and so:

$$\begin{aligned}
\mathbb{P}(W_Q > T) &= C(s,a)e^{-T\mu s}e^{\lambda T} \\
&= C(s,a)e^{-(\mu s-\lambda)T}
\end{aligned}$$

which is precisely (2.3). To derive (2.5), we begin by denoting the probability density of the waiting time distribution by

$$\begin{aligned}
f(T) &= \frac{d}{dT}\mathbb{P}(W_Q \le T) \\
&= (\mu s - \lambda)C(s,a)e^{-(\mu s-\lambda)T}.
\end{aligned}$$

Then

$$\begin{aligned}
\mathbb{E}W_Q &= \int_0^{\infty} Tf(T)dT \\
&= (\mu s-\lambda)C(s,a)\int_0^{\infty} Te^{-(\mu s-\lambda)T}dT \\
&= (\mu s-\lambda)C(s,a)\left(\left[T\frac{e^{-(\mu s-\lambda)T}}{-(\mu s-\lambda)}\right]_0^{\infty} - \int_0^{\infty}\frac{e^{-(\mu s-\lambda)T}}{-(\mu s-\lambda)}dT\right),
\end{aligned}$$

but the first term is zero and so

$$\begin{aligned}
\mathbb{E}W_Q &= -(\mu s-\lambda)C(s,a)\left[\frac{e^{-(\mu s-\lambda)T}}{(\mu s-\lambda)^2}\right]_0^{\infty} \\
&= \frac{C(s,a)}{\mu s-\lambda}.
\end{aligned}$$

Finally, (2.6) follows from Little's Law (see [14]). This tells us that

$$\begin{aligned} \mathbb{E}L_Q &= \lambda \mathbb{E}W_Q \\ &= \frac{\rho C(s,a)}{1-\rho}. \end{aligned}$$

Appendix C

Derivation of Abandonment Equations

Firstly, we calculate an equation to describe the time evolution of the queue, following the same method as in Section 3.2 and secondly, we calculate the stationary distribution of the system when exponential abandonment times with constant patience, γ, are assumed (recall from Section 3.3 that existence of the stationary distribution is guaranteed if the arrival rate is constant). Note that this assumption about abandonment times means that the number of customers to abandon the queue in the next time Δt follows a Poisson distribution with rate $\delta \max(i-s,0)\Delta t$, where i is the number of customers currently in the system and $\delta = \gamma^{-1}$.

Suppose that $\mathbf{p}(t)$ is known and consider what will happen in the next time Δt, where Δt is small. As before, we can write:

$$p_i(t+\Delta t) = \sum_{j=0}^{N} q_{j,i}(t, t+\Delta t) p_j(t)$$

where $q_{j,i}(t, t+\Delta t)$ is the probability that we move from state j at time t to state i at time $t+\Delta t$. In the interval $[t, t+\Delta t]$, we know that the number of customers arriving into the system follows a Poisson distribution with rate parameter $\lambda(t)\Delta t$. However, the number of customers leaving the system is now the sum of a Poisson distribution with rate $\mu_i \Delta t$ and a Poisson distribution with rate $\delta_i \Delta t$, where i is the number of customers currently in the system, $\mu_i = \mu \min(i, s)$ and $\delta_i = \delta \max(i-s, 0)$. We therefore know that the probability of m customers arriving into the system and n customers leaving in the interval $[t, t+\Delta t]$ is

$$\left(e^{-\lambda(t)\Delta t} \frac{(\lambda(t)\Delta t)^m}{m!} \right) \left(\sum_{k=0}^{n} e^{-\mu_i \Delta t} \frac{(\mu_i \Delta t)^k}{k!} e^{-\delta_i \Delta t} \frac{(\delta_i \Delta t)^{(n-k)}}{(n-k)!} \right). \tag{C.1}$$

Now, noting that

$$e^{-(\lambda(t)+\mu_i+\delta_i)\Delta t} = 1 - (\lambda(t) + \mu_i + \delta_i)\Delta t + O(\Delta t^2)$$

we see that (C.1) becomes

$$(1 - (\lambda(t) + \mu_i + \delta_i)\Delta t) \left(\sum_{k=0}^{n} \frac{\lambda(t)^m \mu_i^k \delta_i^{(n-k)}}{m!k!(n-k)!} \right) \Delta t^{(n+m)} + O(\Delta t^{(n+m+2)}).$$

In particular, if $n + m > 1$, then this is $O(\Delta t^2)$. We now proceed to calculate $q_{j,i}(t, t + \Delta t)$. Firstly:

$$\begin{aligned} q_{i,i}(t, t + \Delta t) &= \mathbb{P}(m = n) \\ &= \mathbb{P}(m = n = 0) + O(\Delta t^2) \\ &= 1 - (\lambda(t) + \mu_i + \delta_i)\Delta t + O(\Delta t^2). \end{aligned}$$

Secondly:

$$\begin{aligned} q_{i,i+1}(t, t + \Delta t) &= \mathbb{P}(m = n + 1) \\ &= \mathbb{P}(m = 1, n = 0) + O(\Delta t^2) \\ &= \lambda(t)\Delta t + O(\Delta t^2). \end{aligned}$$

Thirdly:

$$\begin{aligned} q_{i+1,i}(t, t + \Delta t) &= \mathbb{P}(m + 1 = n) \\ &= \mathbb{P}(m = 0, n = 1) + O(\Delta t^2) \\ &= (\mu_{i+1} + \delta_{i+1})\,\Delta t + O(\Delta t^2). \end{aligned}$$

Finally, for $|i - j| > 1$, $q_{j,i}(t, t + \Delta t) = O(\Delta t^2)$. The argument now proceeds in an identical way to Section 3.2 to give equation (3.10), where $A(t)$ is given by (3.14).

We now proceed to calculate the stationary distribution (in the case of a constant arrival rate). This is done in the same way as in Section 3.3, but we now obtain our recurrence relation from A as given by (3.14). Recall that we want to find the vector $\mathbf{p}^\infty$ which satisfies

$$A\mathbf{p}^\infty = \mathbf{0}.$$

Then, denoting $\alpha_i = \mu_i + \delta_i$, we have

$$\begin{aligned} \lambda p_0^\infty &= \alpha_1 p_1^\infty, \\ \lambda p_{i-1}^\infty - (\lambda + \alpha_i) p_i^\infty + \alpha_{i+1} p_{i+1}^\infty &= 0 \qquad \text{if } 1 < i < N, \\ \lambda p_{N-1}^\infty &= \alpha_N p_N^\infty. \end{aligned}$$

Noting that $\alpha_i > 0$ for all i, a straightforward induction shows that

$$p_i^\infty = \frac{\lambda^i}{\prod_{j=1}^i \alpha_j} p_0^\infty.$$

A similarly simple induction then shows that

$$\prod_{j=1}^i \alpha_j = \begin{cases} i!\mu^i & \text{if } i \leq s, \\ s!\mu^s \prod_{k=1}^{i-s}(k\delta + \mu s) & \text{if } i > s. \end{cases}$$

Putting this together, we obtain (3.15).

Appendix D

Eigenvalues of A are real

Here, we show that the eigenvalues of $A(t)$ are real by applying a method described in [24]. We consider the general case where abandonments can occur, so that the matrix A is of the form

$$A(t) = \begin{pmatrix} -\lambda(t) & \alpha_1 & 0 & \dots & 0 \\ \lambda(t) & -(\lambda(t)+\alpha_1) & \alpha_2 & \dots & 0 \\ \vdots & & & & \vdots \\ 0 & \lambda(t) & -(\lambda(t)+\alpha_i) & \alpha_{i+1} & 0 \\ \vdots & & & & \vdots \\ 0 & \dots & \lambda(t) & -(\lambda(t)+\alpha_{N-1}) & \alpha_N \\ 0 & \dots & 0 & \lambda(t) & -\alpha_N \end{pmatrix}$$

where

$$\begin{aligned} \alpha_i &= \mu_i + \delta_i \\ &= \min(i,s)\mu + \max(i-s,0)\delta. \end{aligned}$$

Now $\delta \geq 0$ and $\mu > 0$, so that $\alpha_i > 0$. We now define

$$d_0 = 1$$

and

$$d_i = \sqrt{\frac{\lambda(t)}{\alpha_i}} d_{i-1}$$

if $1 \leq i \leq N$. Now define

$$D(t) = \begin{pmatrix} d_0 & 0 & & \dots & 0 \\ 0 & d_1 & 0 & \dots & 0 \\ \vdots & & & & \vdots \\ 0 & \dots & & 0 & d_N \end{pmatrix}.$$

Note that $D(t)$ is diagonal and every entry on the diagonal is non-zero, therefore $\det D(t) \neq 0$. Then

$$S(t) = D^{-1}(t)A(t)D(t)$$

is a similarity transform, since $\det D(t) \neq 0$. It then follows that the eigenvalues of $S(t)$ and $D(t)$ are the same [23]. Moreover, $S(t)$ is a real symmetric tridiagonal matrix and so its eigenvalues must be real.

Appendix E

Results

Scenario 1	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.1877	0.8108	0.9993	0.6859	0.7963	0.8693	0.7970
2	0.4283	0.8448	0.9999	0.8027	0.8353	0.8625	0.8378
3	0.6666	0.8542	1.0000	0.8027	0.8473	0.9196	0.8491
4	0.6682	0.8503	1.0000	0.8027	0.8448	0.9664	0.8420
5	0.6882	0.8763	1.0000	0.8027	0.8718	0.9664	0.8597
6	0.5536	0.9248	1.0000	0.8819	0.9184	0.9496	0.9243
7	0.7611	0.9313	1.0000	0.9055	0.9270	0.9496	0.9321
8	0.8431	0.9309	1.0000	0.9059	0.9282	0.9876	0.9300
9	0.8431	0.9420	1.0000	0.9059	0.9396	0.9876	0.9359

Scenario 2	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.0706	0.6620	0.9987	0.4703	0.6433	0.7257	0.6299
2	0.2369	0.7106	0.9998	0.6166	0.6956	0.7413	0.6865
3	0.4305	0.7248	1.0000	0.6166	0.7130	0.8564	0.7038
4	0.4305	0.7183	1.0000	0.6131	0.7086	0.9303	0.6918
5	0.4337	0.7582	1.0000	0.6131	0.7495	0.9303	0.7192
6	0.3518	0.8473	0.9999	0.8040	0.8369	0.8577	0.8377
7	0.5896	0.8576	1.0000	0.8141	0.8501	0.8983	0.8502
8	0.6794	0.8566	1.0000	0.8011	0.8513	0.9701	0.8466
9	0.6794	0.8743	1.0000	0.8011	0.8693	0.9701	0.8561

Scenario 3	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.1251	0.8068	0.9996	0.6916	0.7912	0.8573	0.7903
2	0.3267	0.8396	1.0000	0.7909	0.8282	0.8561	0.8292
3	0.5669	0.8490	1.0000	0.7909	0.8405	0.9053	0.8409
4	0.5738	0.8455	1.0000	0.7875	0.8389	0.9593	0.8345
5	0.5738	0.8714	1.0000	0.7875	0.8655	0.9593	0.8519
6	0.4478	0.9201	1.0000	0.8659	0.9123	0.9405	0.9163
7	0.6748	0.9269	1.0000	0.9039	0.9216	0.9405	0.9249
8	0.7643	0.9273	1.0000	0.9012	0.9239	0.9879	0.9245
9	0.7643	0.9384	1.0000	0.9012	0.9352	0.9879	0.9304

Scenario 4	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.0000	0.7862	0.9793	0.5873	0.7824	0.8693	0.7846
2	0.0000	0.8273	0.9958	0.7599	0.8262	0.8625	0.8297
3	0.0000	0.8409	0.9988	0.8027	0.8408	0.8678	0.8433
4	0.0000	0.8399	0.9999	0.8027	0.8399	0.9275	0.8377
5	0.0000	0.8659	0.9999	0.8027	0.8669	0.9275	0.8554
6	0.0000	0.9094	0.9992	0.8210	0.9109	0.9449	0.9167
7	0.0000	0.9193	0.9996	0.9024	0.9215	0.9449	0.9263
8	0.0000	0.9226	1.0000	0.9059	0.9245	0.9579	0.9267
9	0.0000	0.9337	1.0000	0.9059	0.9359	0.9579	0.9326

Scenario 5	Instantaneous			Interval			Whole
Method	Min	Ave	Max	Min	Ave	Max	Day
1	0.1877	0.7829	0.9993	0.6787	0.7688	0.8335	0.7587
2	0.4283	0.8153	0.9999	0.6402	0.8059	0.8547	0.7963
3	0.6114	0.8246	1.0000	0.6402	0.8179	0.9196	0.8074
4	0.6113	0.8207	1.0000	0.6402	0.8154	0.9664	0.8003
5	0.6113	0.8469	1.0000	0.6402	0.8426	0.9664	0.8178
6	0.5536	0.9098	1.0000	0.8331	0.9039	0.9403	0.9029
7	0.7611	0.9163	1.0000	0.8331	0.9124	0.9472	0.9106
8	0.8086	0.9161	1.0000	0.8331	0.9136	0.9876	0.9093
9	0.8086	0.9272	1.0000	0.8331	0.9250	0.9876	0.9151